|  |  |
| --- | --- |
| **Use Case ID:** | UC 01 |
| **Use Case Name:** | Authenticate |
| **Created By:** | Mon David Olarte |
| **Date Created:** | 9/11/24 |
| **Description:** | This use case |
| **Primary Actor:** | User (Patient, Guardian, Therapist, Admin) |
| **Secondary Actor:** | None |
| **Include use cases:** | User Registration (UC 01A) and User Login (UC 01B) |
| **Preconditions:** | * The system is online and accessible * User has access to an internet browser * User is logged in |
| **Postconditions:** | **Success**: The user is successfully authenticated, either by creating a new account or logging into an existing account.  **Failure**: The user receives an appropriate error message (e.g., invalid credentials, duplicate email) and is not granted access to the system. |
| **Triggers:** | * The user is accessing the system and either needs to create an account (register) or log in to an existing account. |
| **Main Flow:** | 1. User accesses the authentication page. 2. System presents the user with two options: "Register" for new users or "Log in" for returning users. 3. User selects either the "Register" or "Log in" option. |
| **Alternate Flow:** |  |
| **Special Requirements** | * The system should support secure authentication (e.g., password encryption, multi-factor authentication if needed). * The system must handle forgotten passwords and email verification efficiently to avoid authentication delays. |
| **Assumptions** | * Users have access to a valid email address and internet access to complete the registration process and or login. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 01A |
| **Use Case Name:** | User Registration |
| **Created By:** | Jan Michael Villeza |
| **Date Created:** | 8/21/24 |
| **Description:** | This use case allows the therapist to register a new patient for an account in the clinic portal to have more access to its facilities. |
| **Primary Actor:** | Patient, Guardian, Therapist, Admin |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The user does not have an existing account in the system. * The system is online and accessible. |
| **Postconditions:** | **Success:** The user is successfully registered, and a confirmation email is sent to verify their email address.  **Failure:** The user is notified of the failure reason (e.g., duplicate email, missing information) and prompted to correct and resubmit. |
| **Triggers:** | * The therapist needs to create a new account for a patient to access therapy services. |
| **Main Flow:** | 1. User navigates to the register new patient page on their dashboard. 2. System prompts the user to enter required registration details, including:    1. Name    2. Email address    3. Password (with security criteria)    4. Role selection (Patient, Guardian, Therapist, Admin) 3. User submits the registration form. 4. System validates the provided information, ensuring all required fields are filled, the email is not already registered, and the password meets security standards. 5. System creates the user account and sends a confirmation email to the provided address. 6. User receives the confirmation email and clicks the link to verify their email. 7. System verifies the email address and activates the user account. 8. User registration is completed, and the user is notified of the successful registration. |
| **Alternate Flow:** | **4a. Duplicate Email:**   * + - 1. If the user enters an email that is already registered in the system, the system notifies the user.       2. The user is prompted to log in instead or use a different email.   **5a. Email Not Received:**  If the user does not receive the confirmation email, the system provides an option to resend the verification link.  User clicks the link to request the confirmation email again and proceeds with email verification. |
| **Special Requirements** | * The system must enforce password security requirements (e.g., minimum length, special characters). * Email delivery should be reliable, and verification links must have an expiration period (e.g., 24 hours). |
| **Assumptions** | * Users have access to a valid email address and internet access to complete the registration process. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 02 |
| **Use Case Name:** | User Login |
| **Created By:** | Jan Michael VIlleza |
| **Date Created:** | 8/21/24 |
| **Description:** | This use case allows the user to log in to their registered account to access more of the portal’s features |
| **Primary Actor:** | Guardian, Therapist, Clinic Admin |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The user has a registered account in the system. * The system is online and accessible. |
| **Postconditions:** | **Success**: The user is authenticated and logged into the system, gaining access to the appropriate dashboard.  **Failure**: The user is notified of the failure reason (e.g., incorrect credentials, locked account) and prompted to retry or recover their account. |
| **Triggers:** | * A user wishes to access their account by logging in. |
| **Main Flow:** | 1. **User navigates to the login page** from the homepage. 2. **System prompts the user** for their email address and password. 3. **User enters their credentials** and submits the form. 4. **System validates the credentials**, checking for a matching account and correct password. 5. **System authenticates the user** and grants access to the appropriate dashboard (Patient, Therapist, Guardian, or Admin). 6. **User is redirected to their dashboard**, where they can access relevant services and information. |
| **Alternate Flow:** | **4a. Invalid Credentials**:   1. If the user enters incorrect email or password, the system notifies the user of the invalid credentials. 2. The user is given the option to retry or reset their password.   **4b. Forgot Password**:   1. User selects the "Forgot Password" option on the login page. 2. System prompts the user to enter their registered email address. 3. System sends a password reset link to the provided email address. 4. User clicks the link, resets their password, and returns to the login page. 5. User logs in with the new password.   **4c. Account Locked**:   1. If the user fails to log in after several attempts, the system may lock the account. 2. System notifies the user that the account is locked and provides instructions to unlock the account (e.g., via email verification or contacting support). |
| **Special Requirements** | * Passwords must be securely hashed and encrypted. * The system should allow for multiple unsuccessful login attempts before locking the account. * The password reset process should include secure validation (e.g., tokenized reset links). |
| **Assumptions** | * Users have registered accounts and valid login credentials. * Users can access their registered email to receive password reset links if needed. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 03 |
| **Use Case Name:** | Book Appointment |
| **Created By:** | Liam Miguel Supremo |
| **Date Created:** | 8/29/24 |
| **Description:** | This use case allows patients and guardians to book an appointment with a therapist through the online portal. |
| **Primary Actor:** | User (Guardian, Patient) |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The system is online and accessible * The therapist’s schedule must be available in the system. * Guardian must be on the Therapro Online Portal. * User has access to an internet browser |
| **Postconditions:** | **Success:** The appointment is successfully booked, and a confirmation notification is sent to the User.  **Failure:** The user is notified of the failure reason (e.g., selected time slot no longer available) and prompted to correct and resubmit. |
| **Triggers:** | * User wants to book an appointment with a therapist. |
| **Main Flow:** | 1. User accesses the Therapro Online Portal. 2. User navigates to the “Book Appointment” section. 3. System displays the list of available schedule of the therapist. 4. User selects an available time slot. 5. User enters their contact information (e.g., name, email, phone number) and confirms the booking details. 6. System validates the provided information and ensures the selected time slot is still available. 7. System saves the appointment and updates the therapist’s schedule. 8. System sends a confirmation notification to the user via email or SMS. 9. User can view the appointment details via the confirmation notification. |
| **Alternate Flow:** | **5a. Incomplete Contact Information:**   1. If the user does not provide all required contact information, the system prompts them to fill in the missing details. 2. The user provides the necessary information.   **6a. System Error:**   1. If the system encounters an error while saving the appointment, it displays an error message to the user. 2. The user is asked to try booking the appointment again later or contact support. |
| **Special Requirements** | * The system must prevent double booking of the same time slot. * The appointment booking process should be simple and user-friendly. * The system should be accessible on both desktop and mobile devices. |
| **Assumptions** | * User has access to a valid email address or phone number to receive appointment confirmation. * User has internet access to complete the appointment booking process. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 04 |
| **Use Case Name:** | Appointment Management |
| **Created By:** | Jan Michael Villeza |
| **Date Created:** | 8/24/24 |
| **Description:** | This use case allows the user to manage their appointments, checking if it is available or has conflicts with other patients. |
| **Primary Actor:** | Therapist, Guardian |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The user has basic information required for booking (e.g., patient name, contact information). * System has available therapy slots. |
| **Postconditions:** | **Success**: The appointment is scheduled, rescheduled, or canceled, and both the user and the therapist receive confirmation.  **Failure**: The user is notified of the failure reason (e.g., no available slots, conflicting appointment) and prompted to try again. |
| **Triggers:** | * A user wants to modify a therapy appointment. |
| **Main Flow:** | 1. User navigates to their appointment calendar on the portal. 2. System displays user’s existing appointments. 3. User selects an available slot for recurring appointments and clicks "Schedule Appointment." 4. System confirms the selected slot and prompts the user for final confirmation. 5. Therapist confirms the appointment. 6. System schedules the appointment, updates the calendar, and sends notifications to both the user and the therapist. 7. User receives appointment confirmation, which includes the date, time, and therapist details. |
| **Alternate Flow:** | **6a. Reschedule Appointment**:   1. User navigates to the reschedule option on the appointment page. 2. System prompts the user for the existing appointment details (e.g., confirmation number, contact info). 3. System displays available slots for rescheduling. 4. User selects a new slot and confirms the rescheduled appointment. 5. System updates the appointment and sends notifications to both the user and the therapist.   **6b. Cancel Appointment**:   1. User navigates to the cancellation option on the appointment page. 2. System prompts the user for the existing appointment details (e.g., confirmation number, contact info). 3. User confirms cancellation. 4. System removes the appointment from both the user and therapist’s calendars and sends cancellation notifications. |
| **Special Requirements** | * The system must ensure data security and prevent unauthorized access to appointment information. * Notifications (email/SMS) must be timely and accurate. |
| **Assumptions** | * Therapists have entered their availability into the system. * Users can access the appointment scheduling feature and provide valid contact information. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 04 |
| **Use Case Name:** | Recurring Appointments |
| **Created By:** | Jan Michael Villeza |
| **Date Created:** | 8/24/24 |
| **Description:** | This use case allows the therapist to schedule recurring appointments for the patient as part of their treatment plan. |
| **Primary Actor:** | Therapist, Assistant |
| **Secondary Actor:** | Guardian |
| **Include use cases:** | None |
| **Preconditions:** | * The patient has completed their **initial appointment** with the therapist. * The therapist has access to the system to schedule appointments. * The system is online and capable of scheduling recurring appointments. |
| **Postconditions:** | **Success:** Recurring appointments are successfully scheduled for the patient over the next 6 months, and reminders are sent to both the guardian and therapist before each session.  **Failure:** If scheduling fails, an appropriate error message is displayed, and the appointments are not set. |
| **Triggers:** | * After the initial appointment, the therapist decides that the patient requires recurring therapy sessions. * The assistant sends reminders before each session based on the configured schedule. |
| **Main Flow:** | 1. Therapist logs into the system after the initial appointment with the patient. 2. Therapist coordinates with the guardian of the patient to determine the most suitable times for recurring appointments. 3. Therapist selects the option to schedule recurring appointments for the patient in the system. 4. Therapist enters the recurrence details, specifying the duration (e.g., 6 months) and the frequency of appointments (e.g., weekly). 5. Therapist reviews available time slots based on their schedule and the parent's/guardian's preferences. 6. Therapist confirms the recurring appointments in the system, which are scheduled automatically. 7. System saves the appointments and updates the therapist's schedule with the recurring appointments for the patient. 8. Assistant manages and adjusts reminders as needed after the appointment is scheduled. 9. System sends a notification to both the therapist and the guardian, confirming the scheduled recurring appointments. |
| **Alternate Flow:** |  |
| **Special Requirements** | * The system must handle recurring appointments efficiently and prevent scheduling conflicts over a long-term period (e.g., 6 months). * The system must send reminders automatically at predefined intervals (e.g., 24 hours or 48 hours before each appointment). |
| **Assumptions** | * Therapists have entered their availability into the system. * Assistant has access to valid contact information for the guardian and therapist to send reminders. * Users can access the appointment scheduling feature and provide valid contact information. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 05 |
| **Use Case Name:** | View Appointments |
| **Created By:** | Jan Michael Villeza |
| **Date Created:** | 8/24/24 |
| **Description:** | This allows the user to view their scheduled appointments or the appointments of all patients, depending on their role. |
| **Primary Actor:** | Patient, Guardian, Therapist, Clinic Admin |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The user has an account and is logged into the system. * The user has scheduled appointments in the system. |
| **Postconditions:** | * **Success**: The user views their upcoming and past appointments with relevant details. * **Failure**: The user is notified if there are no appointments found. |
| **Triggers:** | * A user wants to check their scheduled or past appointments. |
| **Main Flow:** | 1. User navigates to the appointment calendar 2. System retrieves all upcoming and past appointments associated with the user. 3. System displays the appointments list and details including:    * Date and time of each appointment    * Therapist's name    * Appointment type (e.g., virtual or in-person)    * Status (e.g., upcoming, completed, canceled) 4. User reviews the appointments and can click on individual appointments for more details if needed. |
| **Alternate Flow:** | **3a. No Appointments Found**:   1. If the user has no upcoming or past appointments, the system notifies the user that no appointments are found. 2. The user is given the option to schedule a new appointment. |
| **Special Requirements** | * The system should display appointment information in a user-friendly format, with clear distinctions between upcoming, past, and canceled appointments. * The user should be able to easily navigate to other options, such as rescheduling or canceling an appointment. |
| **Assumptions** | * The user has previously scheduled appointments, and they are recorded in the system. * The system can retrieve and display appointment data in real-time. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 06 |
| **Use Case Name:** | Access Game Library |
| **Created By:** | Liam Miguel Supremo |
| **Date Created:** | 8/24/24 |
| **Description:** | This use case allows logged in users to access the game library. |
| **Primary Actor:** | Player (Patient, Guardian, Therapist, Assistant) |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The player has access to the game library. * The player has an existing account. |
| **Postconditions:** | **Success:**   * The game has successfully launched. * The progress and activity of the player are saved.   **Failure:**   * The game was not added or removed from the game library due to an error and receives an appropriate error message. * The game could not be launched due to missing files or updates, the player is informed. |
| **Triggers:** | * The player wants to browse or play a game from their library. |
| **Main Flow:** | 1. The player selects the “Game Library” section from the main menu. 2. Browse Games:    1. 3a. The player views the list of games available in the game library.    2. 3b. The player can search for specific games using the search bar. 3. Play a Game:    1. 4a. The player selects a game from the game library.    2. 4b. The player clicks “Play” to start the game. 4. The system records the player’s actions, such as tracking the number of times each game is played. |
| **Alternate Flow:** | **2a. Player Not Logged In:**   1. If the player is not logged in, the system redirects them to the login page. 2. The player enters their credentials and is granted access to the game library.   **4a. Network Failure:**   1. If there is a network failure during gameplay, the system attempts to reconnect and notify the player. |
| **Special Requirements** | * The game library must support multiple platforms (PC and mobile) and sync across them. * The system must handle large libraries efficiently, with quick search and filtering capabilities. |
| **Assumptions** | * The player has a stable internet connection for browsing and playing games online. * The player has access to a PC or mobile phone. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 07 |
| **Use Case Name:** | Patient Information |
| **Created By:** | Mon David Olarte |
| **Date Created:** | 9/12/24 |
| **Description:** | This use case allows the user to navigate and access patients’ information, including records and progress notes. The content accessible will be dependent on the user’s role. |
| **Primary Actor:** | Therapist, Assistant, Member |
| **Secondary Actor:** | None |
| **Include use cases:** | Patient Records and Patient Progress Notes. |
| **Preconditions:** | * The system is online and accessible. * User is logged in |
| **Postconditions:** | **Success:** Correct scope of information is accessed and editable  **Failure:** An error will be displayed, notifying the user that the system failed to retrieve patient information. |
| **Triggers:** | * + 1. Members wants to view or edit their information.     2. Therapist and Assistant wants to view the information of a specific patient.     3. Therapist wants to edit the information of a specific patient. |
| **Main Flow:** | * + - 1. User accesses patient information          1. Patient records          2. Patient progress notes |
| **Alternate Flow:** | None |
| **Special Requirements** | * System must comply with data privacy regulations to ensure that no sensitive information will be exposed to unauthorized personnel. * System must have the capability to assign roles to users and grant them the correct permissions. |
| **Assumptions** | * User has a stable internet connection. * User can navigate simple UI elements such as buttons and links. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 07A |
| **Use Case Name:** | Access Patient Records |
| **Created By:** | Mon David R. Olarte |
| **Date Created:** | 9/13/2024 |
| **Description:** | This use case specifically tackles the option of accessing patient records. Available information will be based on the role of the user. |
| **Primary Actor:** | Member, Therapist, Assistant |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The system is online and accessible. * User is logged in |
| **Postconditions:** | **Success**: The user successfully accesses patient records.  **Failure**: The system fails to access or save the changes made to the patient records due to a system error or invalid data, and an error message is displayed to the user. |
| **Triggers:** | * + 1. Member wants to view or edit his/her records.     2. Therapist of Assistant wants to view a specific patient’s records.     3. Therapist wants to view or edit the records of a specific patient. |
| **Main Flow:** | 1. System displays the information and grants permissions based on the user’s role    1. **Member:** The system displays only the member’s personal records.    2. **Therapist/Assistant:** The system displays a list of all patients for them to choose from. 2. User views the information 3. User does his/her desired changes (if any)    * + - 1. **Member:** Can only modify his/her own data          2. **Therapist:** Can view and modify any patient record          3. **Assistant:** Can view any patient record but cannot modify them. 4. System validates changes. 5. System saves changes and notifies the user. |
| **Alternate Flow:** | **1a. User is a Member**   * + 1. System displays the member’s personal records, which they can edit.     2. Member can only modify his/her own data   **1b. User is a Therapist**   * + 1. System displays a list of patients     2. Therapist selects a specific patient to access their records     3. Can view and modify ant patient record listed   **1c. User is an Assistant**   1. System displays a list of patients 2. Assistant selects a specific patient to view their records 3. Can only view any patient records listed   **1d. Patient Record Not Found:**   * + 1. If the selected patient’s record does not exist or cannot be found, the system displays an error message, notifying the user about the issue   **3d. Edit Patient Record:**   1. User updates their chosen information. 2. User saves changes to the patient record. 3. System validates the changes. 4. System saves the changes and alerts the user about the changes.   **4a. Invalid Data Entry:**   1. If the therapist enters invalid or incomplete data, the system prompts them to correct the errors before saving.   **5a. System Error During Save:**   1. If a system error occurs while saving, the system displays an error message, and the therapist’s changes are not saved. The therapist is prompted to try again or contact support. |
| **Special Requirements** | * System must comply with data privacy regulations to ensure that no sensitive information will be exposed to unauthorized personnel. * System must have the capability to assign roles to users and grant them the correct permissions. * The system must have validations on input data |
| **Assumptions** | * User has a stable internet connection. * User can navigate simple UI elements such as buttons and links. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 07B |
| **Use Case Name:** | Access Patient Progress Notes |
| **Created By:** | Mon David R. Olarte |
| **Date Created:** | 9/13/2024 |
| **Description:** | This use case specifically tackles the option of accessing patient progress notes. Available information and permissions will be based on the role of the user. |
| **Primary Actor:** | Member, Therapist, Assistant |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The system is online and accessible. * User is logged in |
| **Postconditions:** | **Success**: The user successfully accesses patient progress notes  **Failure**: The system fails to access or save the changes made to the progress notes due to a system error or invalid data, and an error message is displayed to the user. |
| **Triggers:** | 1. Member wants to access his/her progress notes 2. Therapist or Assistant wants to view a specific patient’s progress notes 3. Therapist wants to modify the progress notes of a specific patient |
| **Main Flow:** | 1. System displays patient progress notes and grants permission based on the user’s role    1. **Member:** The system displays only the member’s personal progress notes.    2. **Therapist/Assistant:** The system displays a list of all patients for them to choose from. 2. User views the information 3. User does his/her desired changes (if any).    * + - 1. **Member:** Can only modify his/her own data          2. **Therapist:** Can view and modify any patient record          3. **Assistant:** Can view any patient record but cannot modify them. 4. System validates changes. 5. System saves the changes and notifies the user. |
| **Alternate Flow:** | **1a. User is a member**   1. System displays only their personal progress notes, which they can modify.   **1b. User is a therapist**   1. System displays a list of patients 2. Therapist selects a specific patient to access their progress notes 3. Therapist can view and modify any patient progress note listed   **1c. User is an assistant**   1. System displays a list of patients 2. Therapist selects a specific patient to view their progress notes 3. Assistant can view any patient record listed   **1d. No Existing Notes:**   1. If there are no progress notes available for the user, informs them about the issue.   **3a. Add Progress Note**   1. User clicks the “Add Progress Note” button. 2. User inputs the note details, such as session date, note title, and main content. 3. User confirms the content by saving the changes made.   **3b. Edit Progress Note**   1. Therapist selects specific progress note. 2. Therapist selects the “Edit Progress Note” button. 3. System makes the information editable. 4. Therapist does his/her desired changes.   **3c. Delete Progress Note**   1. Therapist selects a specific progress note 2. Therapist selects “Delete Progress Note” 3. System displays a warning that the action cannot be undone 4. Therapist confirms the action 5. The specific progress note is deleted from the system.   **4a. Invalid Data:**   1. If the system detects invalid data, it prompts the user to check their inputs. 2. User resubmits   **4b. Incomplete Data**   1. If the system detects that the user input is incomplete, it marks the areas that need completion. 2. User completes their input 3. User resubmits   **5a. System Error:**  If the system encounters a technical issue while saving the notes, it displays an error message and logs the error for further investigation. |
| **Special Requirements** | * System must comply with data privacy regulations to ensure that no sensitive information will be exposed to unauthorized personnel. * System must have the capability to assign roles to users and grant them the correct permissions. * The system must have validations on input data |
| **Assumptions** | * User has a stable internet connection. * User can navigate simple UI elements such as buttons and links. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 07 |
| **Use Case Name:** | Record Patient Progress |
| **Created By:** | Mon David Olarte |
| **Date Created:** | 8/25/24 |
| **Description:** | This use case describes the process by which a Therapist or Administrator records progress notes for patients within the system. This includes viewing, editing, deleting, and saving progress notes. |
| **Primary Actor:** | Therapist, Administrator |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * There are existing progress notes. * The system is online and accessible. * The therapist/administrator is logged in, granting them appropriate permissions. |
| **Postconditions:** | **Success**: The therapist or administrator can view, edit, delete, and save patients’ progress notes.  **Failure**: The system fails to save the progress notes due to a system error or invalid data, and an error message is displayed to the user. |
| **Trigger:** | * The therapist or administrator needs to update, review, or manage a patient's progress notes. |
| **Main Flow:** | 1. User selects a patient from the list of patients in the system. 2. The system displays the patient's existing progress notes 3. User selects the option to view, edit, or delete the progress notes. 4. User saves the changes. 5. The system validates the changes and saves the updated progress notes. 6. The system confirms that the progress notes have been successfully updated and displays the updated information to the actor. |
| **Alternate Flow:** | **1a. No Existing Notes:**   1. If there are no progress notes for the selected session, the system informs the primary actor and provides an option to create new progress notes.   **5a. Invalid Data:**   1. If the system detects invalid or incomplete data, it prompts the primary actor to correct the errors and resubmit.   **6b. System Error:**   1. If the system encounters a technical issue while saving the notes, it displays an error message and logs the error for further investigation. |
| **Special Requirements** | * System must comply with data privacy regulations to ensure that no sensitive information will be exposed to unauthorized personnel. |
| **Assumptions** | * User is able to navigate UI elements such as buttons and links. * A reliable internet connection is available to ensure the information displayed is up to date. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 08 |
| **Use Case Name:** | View Progress Notes |
| **Created By:** | Mon David Olarte |
| **Date Created:** | 8/25/24 |
| **Description:** | This use case describes the process by which an Assistant or Member views progress notes for a specific patient session within the system. It includes steps for both user roles to access and read the notes. |
| **Primary Actor:** | Assistant, Member |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The assistant or member is logged into the system with appropriate access permissions. * The system is online and operational. * Progress notes for the patient are available in the system. |
| **Postconditions:** | **Success:** The assistant or member can successfully view the progress notes of the selected patient session.  **Failure:** If the progress notes cannot be retrieved, an error message is displayed, and the notes are not shown. |
| **Trigger:** | * The assistant or member wants to view the progress notes for a specific patient session. |
| **Main Flow:** | 1. User navigates to the progress notes window. 2. The system checks the role of the user:    1. 2a. If the user is an assistant, the system displays a list of all patients.    2. 2b. If the user is a member, the system displays his/her profile containing the notes 3. The system retrieves and displays the progress notes for the selected patient. 4. The user reads through the progress notes. 5. User returns to the menu. |
| **Alternate Flow:** | **2a. User is a member:**   1. The system displays his/her notes   **2b. User is an assistant:**   1. The system displays the list of patients 2. The assistant selects a specific patient from the list to view his/her notes   **3a. No Progress Notes Available:**   1. If there are no progress notes available for the selected patient, the system informs the actor that no notes are available for viewing.   **4a. System Error:**   1. If the system encounters an error while retrieving the progress notes, an error message is displayed, and the actor is advised to try again later or contact support. |
| **Special Requirements** | * The progress notes should be presented in a clear and readable format. * The system must ensure that only authorized users can access patient notes, respecting privacy and confidentiality. |
| **Assumptions** | * The assistant or member can navigate simple UI elements such as buttons and links * The system is properly secured to prevent unauthorized access to sensitive patient information. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 09 |
| **Use Case Name:** | Manage Patient Records |
| **Created By:** | Mon David Olarte |
| **Date Created:** | 8/25/24 |
| **Description:** | This use case outlines the process by which a Therapist manages patient records within the system. This includes viewing, adding, editing, and deleting patient information, ensuring that the records are up to date and accurately maintained. |
| **Primary Actor:** | Therapist |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * Therapist is logged in to the portal. * System is online and operational. * The therapist has appropriate permissions to manage records (view, add, edit, delete). |
| **Postconditions:** | **Success:** The therapist can view, update, edit, and delete patient records, and all changes are saved to the system.  **Failure:** An error will be displayed to the therapist containing the reason why the patient records cannot be accessed. |
| **Trigger:** | * The Therapist needs to access to a patient's record. |
| **Main Flow:** | 1. Therapist navigates to the list of patients. 2. System displays a list of all patients. 3. Therapist selects a specific patient to manage their records. 4. System retrieves and displays the patient’s records. 5. Therapist updates chosen information. 6. Therapist saves changes to the patient record 7. System validates the changes and confirms that the changes were made successfully. |
| **Alternate Flow:** | **4a. Patient Record Not Found:**   * If the selected patient’s record does not exist or cannot be found, the system displays an error message, and the therapist may need to verify the patient’s details or contact system support.   **5a. Invalid Data Entry:**   1. If the therapist enters invalid or incomplete data, the system prompts them to correct the errors before saving.   **6a. System Error During Save:**   1. If a system error occurs while saving, the system displays an error message, and the therapist’s changes are not saved. The therapist is prompted to try again or contact support. |
| **Special Requirements** | * The system should have audit logging to track changes made to patient records, including who made the changes and when. * The interface should be user-friendly, allowing therapists to quickly access and update patient records. |
| **Assumptions** | * User is able to navigate UI elements such as buttons and links. * A reliable internet connection is available to ensure the information displayed is up to date. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 11 |
| **Use Case Name:** | Report Issue |
| **Created By:** | Mon David Olarte |
| **Date Created:** | 8/25/24 |
| **Description:** | This use case describes the process by which a Therapist, Assistant, or Member reports an issue related to the system. It involves the steps to fill out, submit, and confirm the submission of the issue report. |
| **Primary Actor:** | Therapist, Assistant, Member |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * System is available and online. * User is logged in to the portal. * User has an issue related to the system. |
| **Postconditions:** | **Success:** The issue reported will be submitted to the system and a confirmation will be displayed.  **Failure:** An error will be displayed to the user containing the reason why the report was not submitted. |
| **Trigger:** | * A user wants to report an issue that they know/encountered. |
| **Main Flow:** | 1. User selects the option to report an issue from the main menu or dashboard. 2. The system prompts the user to fill out an issue report form. 3. User completes the form with the necessary details. 4. User submits the issue report. 5. The system validates the submitted information. 6. The system saves report. 7. The system displays a confirmation message to the user. |
| **Alternate Flow:** | **2a. Incomplete Form:**   1. If the user submits the form without filling in all required fields, the system highlights the missing fields and prompts the user to complete them before resubmitting.   **5a. Invalid Data:**   1. If the system detects invalid data, it notifies the actor of the specific error and requests corrections.   **6a. System Error:**   1. If the system encounters an error while saving the issue report, an error message is displayed, and the report is not submitted. The system logs the error for further investigation. |
| **Special Requirements** | * The system must comply with data privacy. * User has the capability to explain or describe the issue. |
| **Assumptions** | * User is logged in. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 13 |
| **Use Case Name:** | Update User Accounts |
| **Created By:** | Kai Butalid |
| **Date Created:** | 8/24/24 |
| **Description:** | This use case describes the process by which the Administrator manages user accounts within the system, including editing or deleting and setting permissions for user accounts. |
| **Primary Actor:** | Administrator |
| **Secondary Actor:** | None |
| **Include use cases:** | Edit User Account (UC 13A), Delete User Account (UC 13B) and Set User Permissions (UC 13C) |
| **Preconditions:** | * The Administrator is logged into the system with the necessary privileges to manage user accounts. * Users (Therapist, Patient, Assistant, Guardian) are registered in the system. |
| **Postconditions:** | **Success:** The user accounts are successfully managed (edited, deleted, or permissions set).  **Failure:**   * The user account was not updated or deleted due to a system error or invalid input. * No changes are made to the user accounts list. |
| **Trigger:** | * The administrator needs to manage user accounts to ensure correct information, deactivate or delete accounts, or adjust user permissions in the portal system. |
| **Main Flow:** | 1. Administrator navigates to the "User Management" section. 2. Administrator selects the "Manage User Account" option. 3. System displays a list of all users (Therapist, Patient, Assistant, Guardian) with options to edit, delete, or set permissions for each user. |
| **Alternate Flow:** | **2a. System Error:**   1. If the system encounters an error and fails to load the "Manage User Account" page, it displays an error message to the Administrator. 2. The Administrator is asked to refresh the page or try again later. |
| **Special Requirements** | * The system must ensure that user data is handled securely during account management. |
| **Assumptions** | * The Administrator has the necessary privileges to manage all aspects of user accounts. * Users are aware that their accounts may be managed by an administrator and understand the implications. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 13A |
| **Use Case Name:** | Edit User Account |
| **Created By:** | Kai Butalid |
| **Date Created:** | 9/11/24 |
| **Description:** | This use case describes the process by which the Administrator edits user accounts within the system. |
| **Primary Actor:** | Administrator |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The Administrator is logged into the system with the necessary privileges to manage user accounts. * Users (Therapist, Patient, Assistant, Guardian) are registered in the system. * The Administrator is in the “Manage User Account” page. |
| **Postconditions:** | **Success:** The user accounts are successfully edited.  **Failure:**   * The user account was not updated due to a system error or invalid input. * No changes are made to the user accounts list. |
| **Trigger:** | * The administrator needs to manage user accounts to ensure correct information. |
| **Main Flow:** | 1. System displays a list of all users (Therapist, Patient, Assistant, Guardian). 2. Administrator selects a user and chooses the "Edit" option. 3. System displays the editable fields for the user, such as name, email, contact information, and role. 4. Administrator updates the user details as needed. 5. Administrator submits the changes. 6. System validates the new information and updates the user account in the database. 7. System confirms the update and notifies the administrator. |
| **Alternate Flow:** | **4a. Invalid User Information:**   1. If the administrator enters invalid information (e.g., incorrect email format, missing required fields), the system notifies the administrator. 2. Administrator is prompted to correct the information and resubmit. |
| **Special Requirements** | * The system must ensure that user data is handled securely during account management. |
| **Assumptions** | * The Administrator has the necessary privileges to manage all aspects of user accounts. * Users are aware that their accounts may be managed by an administrator and understand the implications. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 13B |
| **Use Case Name:** | Delete User Account |
| **Created By:** | Kai Butalid |
| **Date Created:** | 9/11/24 |
| **Description:** | This use case describes the process by which the Administrator deletes user accounts within the system. |
| **Primary Actor:** | Administrator |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The Administrator is logged into the system with the necessary privileges to manage user accounts. * Users (Therapist, Patient, Assistant, Guardian) are registered in the system. * The Administrator is in the “Manage User Account” page. |
| **Postconditions:** | **Success:** The user accounts are successfully edited.  **Failure:**   * The user account was not updated due to a system error or invalid input. * No changes are made to the user accounts list. |
| **Trigger:** | * The administrator needs to delete user accounts to deactivate or delete accounts. |
| **Main Flow:** | 1. System displays a list of all users (Therapist, Patient, Assistant, Guardian). 2. Administrator selects a user and chooses the "Delete" option. 3. System prompts the administrator to confirm the deletion. 4. Administrator confirms the deletion. 5. System checks for any dependencies or data related to the user (e.g., scheduled sessions, assigned tasks). 6. System deletes the user account and all associated data or reassigns dependencies as necessary. 7. System confirms the deletion and notifies the administrator. |
| **Alternate Flow:** | **5a. Deletion Conflict:**   1. If the user being deleted has active dependencies (e.g., scheduled sessions or tasks), the system notifies the administrator of the conflict. 2. Administrator is prompted to reassign or cancel dependencies before proceeding with deletion. |
| **Special Requirements** | * The system must ensure that user data is handled securely during account management. |
| **Assumptions** | * The Administrator has the necessary privileges to manage all aspects of user accounts. * Users are aware that their accounts may be managed by an administrator and understand the implications. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC 13C |
| **Use Case Name:** | Set User Permissions |
| **Created By:** | Kai Butalid |
| **Date Created:** | 9/11/24 |
| **Description:** | This use case describes the process by which the Administrator deletes user accounts within the system. |
| **Primary Actor:** | Administrator |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The Administrator is logged into the system with the necessary privileges to manage user accounts. * Users (Therapist, Patient, Assistant, Guardian) are registered in the system. * The Administrator is in the “Manage User Account” page. |
| **Postconditions:** | **Success:** The permissions of the user accounts are successfully set.  **Failure:**   * The permissions were not updated due to a system error or invalid input. * No changes are made to the user accounts list. |
| **Trigger:** | * The administrator needs to set permissions to adjust user permissions. |
| **Main Flow:** | 1. System displays a list of all users (Therapist, Patient, Assistant, Guardian). 2. Administrator selects a user and chooses the "Set Permissions" option. 3. System displays the current permissions for the user and options to adjust them. 4. Administrator selects the appropriate permissions for the user based on their role. 5. Administrator submits the new permissions. 6. System validates the permissions for any conflicts or inconsistencies. 7. System updates the user’s permissions in the database. 8. System confirms the update and notifies the administrator. |
| **Alternate Flow:** | **6a. Permission Conflict:**   1. If the new permissions conflict with system rules or roles, the system notifies the administrator. 2. Administrator is prompted to resolve the conflict by adjusting the permissions. |
| **Special Requirements** | * The system must ensure that user data is handled securely during account management. |
| **Assumptions** | * The Administrator has the necessary privileges to manage all aspects of user accounts. * The administrator understands the roles, permissions, and data management policies within the system. * Users are aware that their accounts may be managed by an administrator and understand the implications. |

|  |  |
| --- | --- |
| **Use Case ID:** | UC14 |
| **Use Case Name:** | Monitor Site Traffic |
| **Created By:** | Kai Butalid |
| **Date Created:** | 8/26/24 |
| **Description:** | This use case describes the process by which an Administrator monitors and analyzes site traffic data, including user visits, activity, and engagement metrics within the system. |
| **Primary Actor:** | Administrator |
| **Secondary Actor:** | None |
| **Include use cases:** | None |
| **Preconditions:** | * The administrator is logged into the portal system with appropriate administrative privileges. * The system is online and accessible. * The system tracks and logs site traffic data, including user visits and interactions. |
| **Postconditions:** | **Success:** The administrator successfully views and analyzes site traffic data, including user activity and engagement metrics.  **Failure:** The administrator is notified of any issues with accessing or displaying traffic data and is prompted to resolve the issue. |
| **Trigger:** | * The administrator wants to review and analyze site traffic to understand user engagement and system usage patterns. |
| **Main Flow:** | 1. Administrator navigates to the "Analytics" section. 2. Administrator selects the "Monitor Site Traffic" option. 3. System displays options to view traffic data, including:    * Date range selection    * Traffic metrics (e.g., page views, unique visitors, session duration)    * User activity details (e.g., login frequency, session times) 4. Administrator selects the desired date range and metrics to view. 5. System retrieves and displays the site traffic data according to the selected parameters. 6. Administrator analyzes the traffic data. 7. Administrator generates reports based on the traffic data if needed. |
| **Alternate Flow:** | **4a. No Data Available:**   1. If there is no traffic data available for the selected date range, the system notifies the administrator. 2. Administrator is prompted to adjust the date range or check data collection settings. |
| **Special Requirements** | * The system must ensure the accuracy and timeliness of traffic data. |
| **Assumptions** | * The administrator is familiar with interpreting site traffic metrics and reports. * The system is properly configured to track and log all relevant site traffic data. |