THOMAS MORE

MEMENTO

TEAM: LST1

Analysis and design report

Niels Van Nerum,
Malak Malkaoui,
Zubair Hussain,
João Lagos Nunes,
Rob Verbeek,
Matej Bursik

2023 - 2024

Table of contents

1. Introduction

As a group, we have got the chance to work on a meaningful project commissioned by a client who envisions an application to support young individuals struggling with memory problems. We began with an instructive meeting with our client where we delved into the specifics of this application. Followed by a long discussion within our team.

In this report, we will bring together what we have learned from our client and our early research. This combined information will be the foundation of our project. Additionally, we will explain how this application could really make a big difference for our specific target audience.

Using what we have learned, we have created a clear use case diagram. This will show how the application works and how users will interact with it. We have also written detailed descriptions for each use case, explaining every step in a way that makes it easy to grasp what the application can do.

2. Assignment description

In this section, we delve into a comprehensive overview of the assignment at hand. We will begin by providing essential background information. We will discuss our client and why they need a new system. We will also talk about what is not working with the current one, and why a custom solution is crucial.

Further on we will discuss the objectives and target groups of our application. We will take a close look at how this new tool will offer crucial support to our target group. By breaking down what each possible user can do on this application, we will build a clear picture of how this application will make a real difference in their daily routines.

2.1. Background information

This project was commissioned by our customer. The reason they want this application is to help younger people with memory problems get through their daily life while being independent of others. Currently they are using a calendar to keep track of their day-to-day life. This calendar includes things such as appointments but also a daily routine in the morning. Chores, breakfast, medication are all things that are planned into their schedule so they do not forget to do them. The problem with this, however is that even with a schedule many of them forget things and would like to be notified in case they miss them.

On top of a calendar to help them plan, they also keep a diary to take notes of what has happened each day or to log things that happened during an appointment or event. They want to be able to remember what happened last time they met with friends. However, having to find the correct notes each time is also a problem that can occur and would be resolved by having a digital diary.

Feedback from the people surrounding the patients in the form of comments is another thing asked for to be added in the application. An example provided to us by our customer is that sometimes, when they are having a bad day, their friends and family should be able to tell them about what happened, so they are able to remember it later on. Currently, there is no system in place to help with this.

At the moment, doctors usually have to tell their patients which medication to take and how much. They would also like to know any side-effects that occurred from taking the medication and an estimate of how long it took before the side-effects started showing up. At the moment the doctors give patients a list of questions to answer however, because it is hard for patients to remember things they usually cannot accurately tell their doctors about the side-effects.

2.2. Objectives and target groups

Main user: The main users of the system are patients between 18 to 50. They will see various benefits in their everyday life as a result of using the automated system. These improvements include improved memory assistance, effectively organizing and keeping track of her daily routines, and an easy access to beneficial information. The patient will have a single, user-friendly platform in order to manage their schedule, track tasks, and maintain a digital diary. This will help the patient's overall well-being.

Caretakers: Caretakers will gain more visibility into the patient's everyday life and well-being. They will be able to interact with the system by leaving comments on calendar entries, reading the patient's calendar, and even filling out questionnaires which consist of standard questions and some specific questions that need to be shared by the doctor. Improved communication and information sharing will enable caretakers to provide more effective assistance and care, ensuring the patient's requirements are fulfilled quickly and effectively.

Doctor: The doctor's primary responsibility includes medical management, treatment planning, monitoring patient progress, and making educated choices based on the patient's feedback and the assessments on the filled questionnaire. This information can lead to more tailored and effective medication which in turn can improve the patient's overall health.

Acquaintances: Acquaintances are able to login, and interact with the system by making comments, viewing comments and reading calendar entries. The actor, Main user and Caretakers inherit all the tasks the acquaintances are able to do.

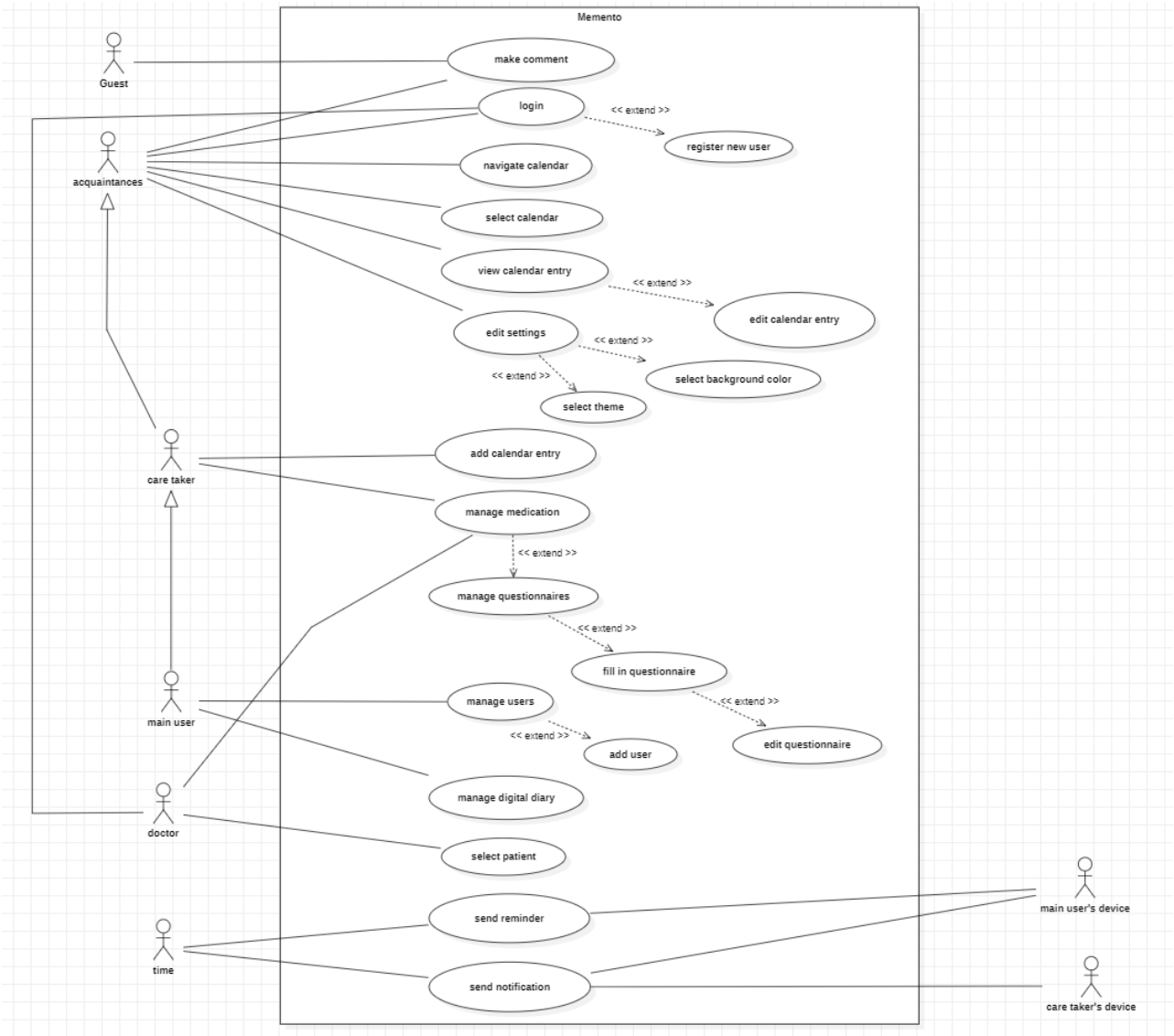
Guests: Guests do not require a login. They can only access the entry of which they are given a QR code. With access to this calendar entry they can then leave a comment.

3. Requirements analysis

In this section, we focus on understanding what the application needs to do. We start by introducing a carefully designed use case diagram. This diagram was created after discussions with both our client and within our team, and it lays out the main functions of the app. Following this, we give detailed descriptions of each function shown in the diagram.

3.1. Functional requirements

3.1.1. Use Case Diagram



3.1.2. Use Case Descriptions with prototypes

In this section, we focus on understanding what the application needs to do. We start by introducing a carefully designed use case diagram. This diagram was created after discussions with both our client and within our team, and it lays out the main functions of the app. Following this, we give detailed descriptions of each function shown in the diagram.

3.1.2.1. Make comment

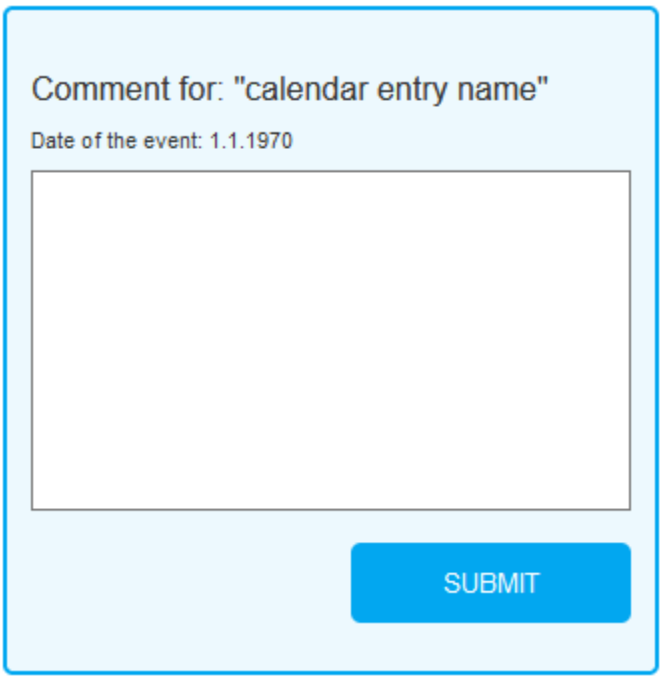
Functionality:

As an acquaintance or Guest, I can comment on calendar entries.

Normal flow:
After the actor clicks on a link, the system shows a comment field for the specified entry (by the link). The actor writes a comment. The system adds the comment underneath the opened calendar entry.

Alternatives:
Comment under an entry: Based on user permissions the system allows the user to comment directly under an entry without the link.

Additional remarks:
The make comment use case I for mostly guests who do not have access to see and the calendar entry. They are provided by a link to the use case by a user we adequate permission.



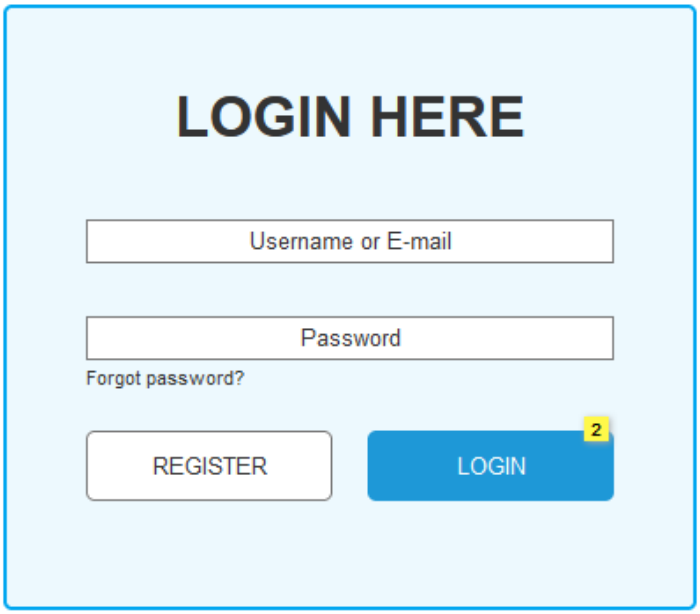
3.1.2.2. Login

Functionality:

As an acquaintance, I can login in.

Normal flow:
The system shows a login page. The system asks for the login information (username, password). The actor completes and confirms the information. The system redirects to the main page.

Alternatives:
Login not successful: The system displays an error message and asks again for the login information.
Forgot password: The system displays an option if the user does not remember the password of his/her account so that the user can reset it.
Register new user: The system runs the Register new user use case after the REGISTER button was used.



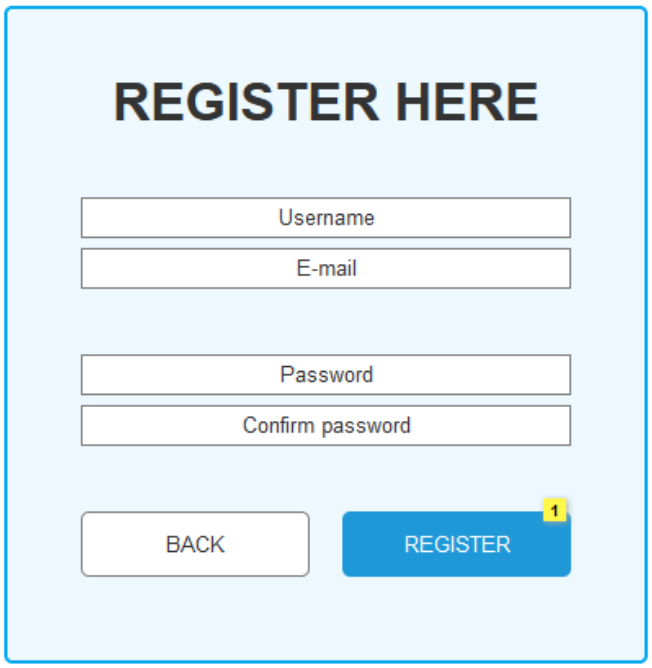
3.1.2.3. Register new user

Functionality:

As an acquaintance, I can register as a new user of the application.

Normal flow:
The system shows a registration form. The system asks for the user information (username, email, password, confirm password). The actor completes and confirms the information. The system redirects to the main page.

Alternatives:
Registration not successful: The actor has incorrectly filled out the registration form, so they are returned back to the Register new user.

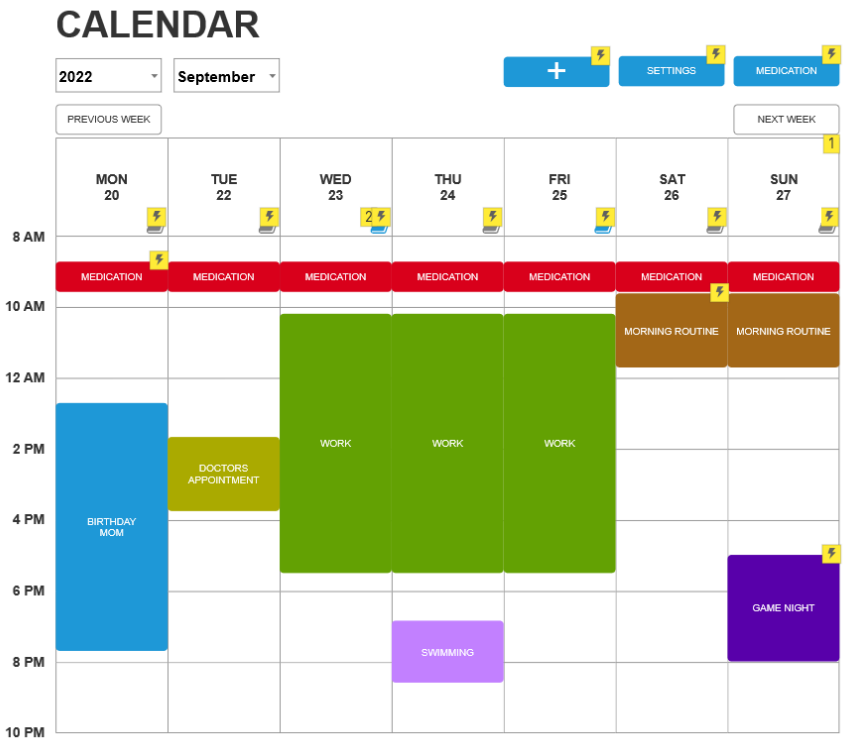


3.1.2.4. Navigate Calander

Functionality:
As an acquaintance, I can navigate through the calendar.

Precondition:
The acquaintance must be logged in.

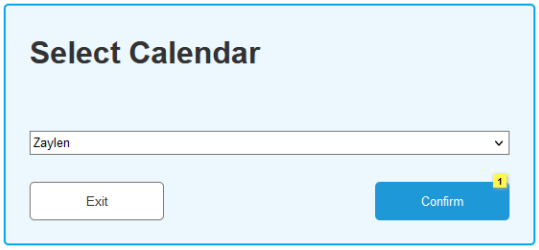
Normal flow:
The System displays a calendar overview. The actor selects a date, month, week. The system changes the calendar overview to the respective data.



3.1.2.5. Select Calendar

Functionality:
As an acquaintance, I can select which available calendar do I want to view.

Normal flow:
The system displays a dropdown menu with all available calendar for the user. User selects that calendar they want to view. The System redirects the user to the selected calendar.



3.1.2.6. View Calendar Entry

Functionality:
As a main user, I can see calendar entry details.

Normal Flow:
The System displays a calendar overview. The actor selects an active entry. The system opens a window with the entry details.

Alternatives:
Share: The actor can choose to create a link for sharing the entry with guests.
Edit: The actor can choose to edit the selected entry.
Comment: The actor can leave comments on the selected entry.

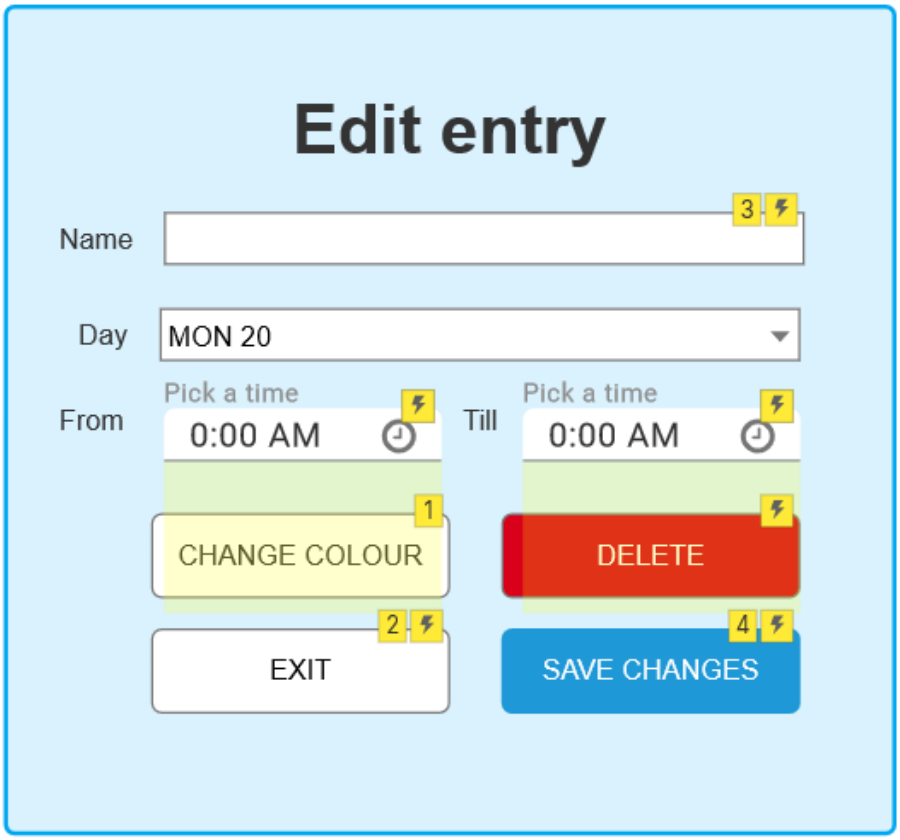
3.1.2.7. Edit Calendar Entry

Functionality:
As the main user, I can edit the details of a calendar entry.

Precondition:
The user has permission to edit calendar entries.

Normal Flow:
The System displays the calendar entry with its current name, date, time. The actor changes one or multiple of the entry details and saves changes. The system asks for confirmation. The actor confirms the changes.

Alternatives:
Delete: The Actor can delete the selected calendar entry.
Please note: Entries older than a day are made inactive instead of deleted entirely.
Change Colour: The actor can change the colour of the calendar entry.



3.1.2.8. Edit Checklist Entry

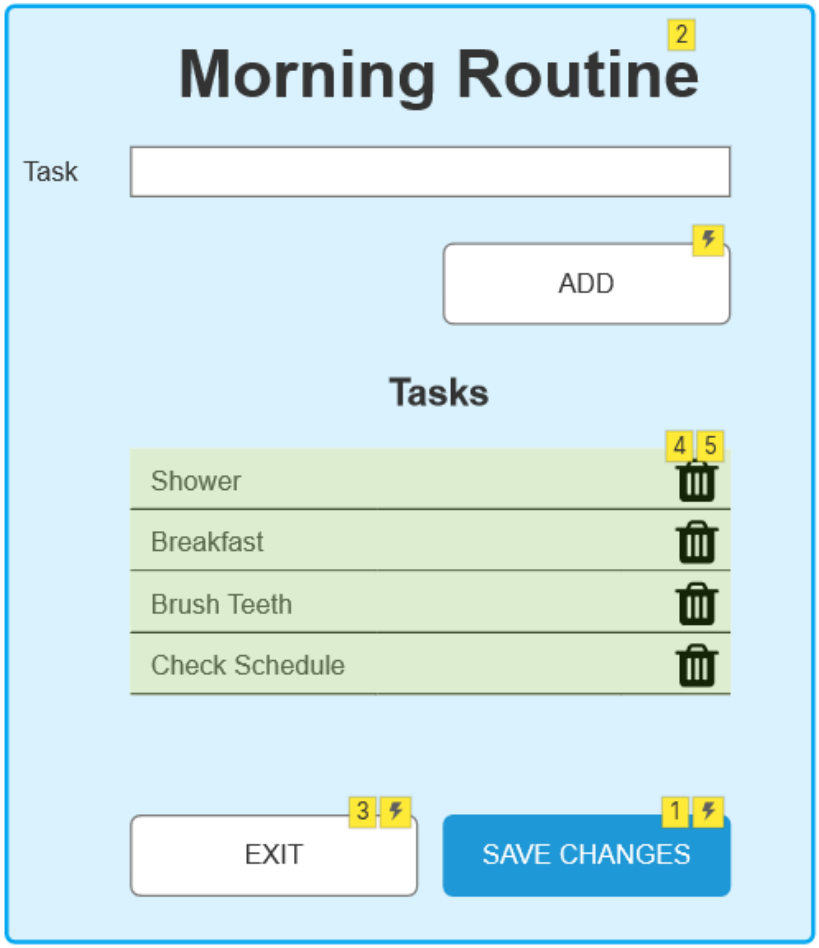
Functionality:
As the main user, I can edit the details of a checklist entry.

Precondition:
The user has permission to edit calendar entries.

Normal Flow:
The system displays the checklist with its current tasks. The actor adds a task and saves the changes. The system asks for confirmation. The actor confirms the changes.

Alternatives:
Remove Task: The actor can delete any of the tasks in the checklist.
Please note: Removing a task deletes it entirely, making it inactive serves no purpose as the tasks are not reused.
Cancel: The actor can choose to cancel editing the checklist entry. The system will not save any changes in this case.

Additional remarks: The actor should be able to move the order of tasks in the checklist around by dragging them around.



3.1.2.9. Edit Settings

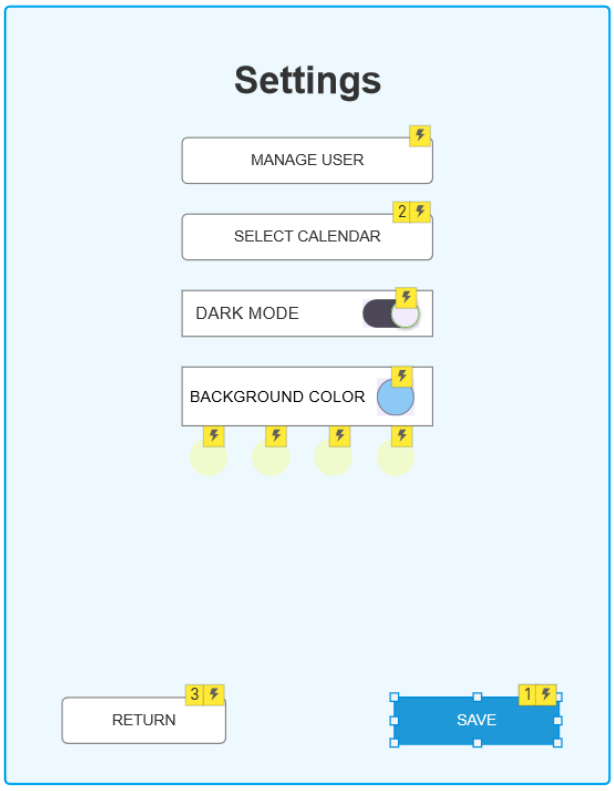
Functionality:
As an acquaintance, I can change the app settings.

Precondition:
The acquaintance has access to the app settings.

Normal Flow:
The system displays the settings page. The actor changes the settings and saves. The system asks for confirmation. The actor confirms. The system redirects the actor to the calendar view.

Alternatives:

The actor exits the settings without saving: The actor exits the settings without saving. The system displays a message saying all changes will be lost. The actor confirms. The system redirects the actor to the calendar view.
Change Theme: The system displays an option for the actor to change the app theme from dark mode to light mode.
Change Background Colour: The system displays an option for the actor to change the background colour.



3.1.2.10. Select Background Color

Functionality:
As an acquaintance, I can change the app's background color.

Precondition:
The acquaintance has access to the app settings.

Normal Flow:
The actor navigates to settings within the application. The system displays the background color setting. The actor changes the background color setting. The system displays the settings with the new background color. The actor saves the settings. The system displays a confirmation message. The actor confirms the changes. The system redirects the actor to the view calendar.

Alternative:

The actor exits without saving: The actor exits the settings without saving. The system displays a message saying all changes will be lost. The actor confirms. The system redirects the actor to the calendar view.

3.1.2.11. Select Theme

Functionality:
As an acquaintance, I can change the app theme.

Precondition:
The acquaintance has access to the app settings.

Normal Flow:
The actor navigates to settings within the application. The system displays the theme setting. The actor changes the theme setting. The system displays the settings with the new theme setting. The actor saves the settings. The system displays a confirmation message. The actor confirms the changes. The system redirects the actor to the view calendar.

Alternative:

The actor exits without saving: The actor exits the settings without saving. The system displays a message saying all changes will be lost. The actor confirms. The system redirects the actor to the calendar view.

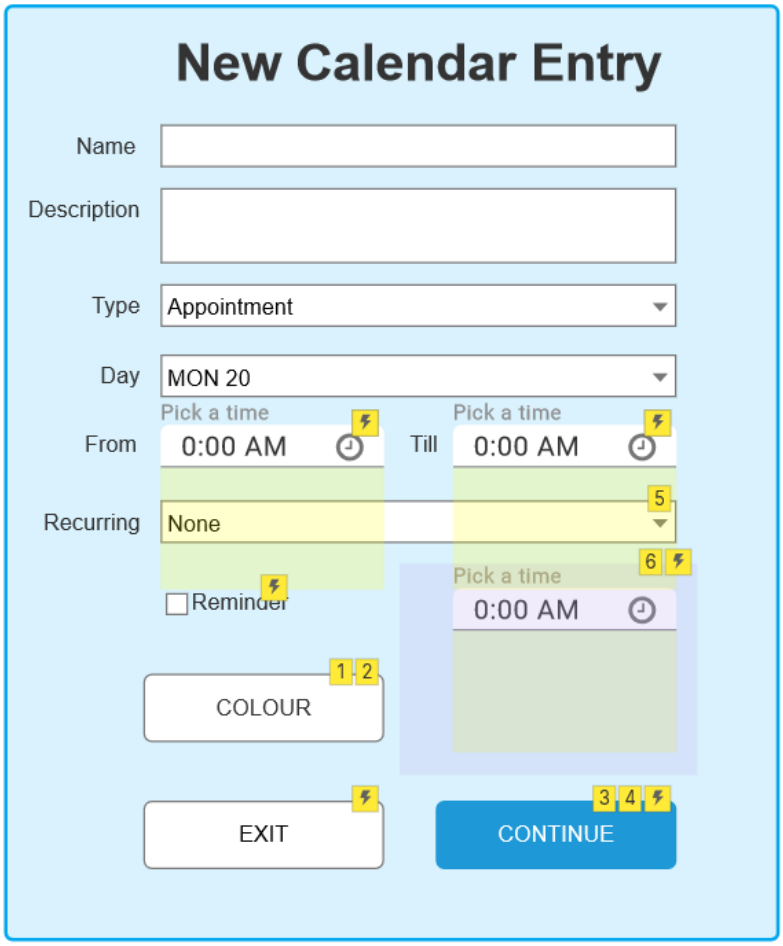
3.1.2.12. Add Calendar Entry

Functionality:
As the main user, I can add new entries to the calendar.

Precondition:
The user has permission to add a calendar entry.

Normal Flow:
The system shows the calendar view, The actor selects to add a new calendar entry. The system asks the user to fill in the name, description, choose a calendar type, day, time, whether it is recurring or not and if it needs a reminder. The actor fills them in and saves the entry. The system asks for confirmation. The actor confirms. The system adds the new calendar entry to the calendar view.

Alternatives:
Checklist Type: System shows additional options to create a checklist.
Choose Colour: The system opens a colour wheel for the actor to select a colour.
Cancel: The actor can cancel adding a new calendar entry.



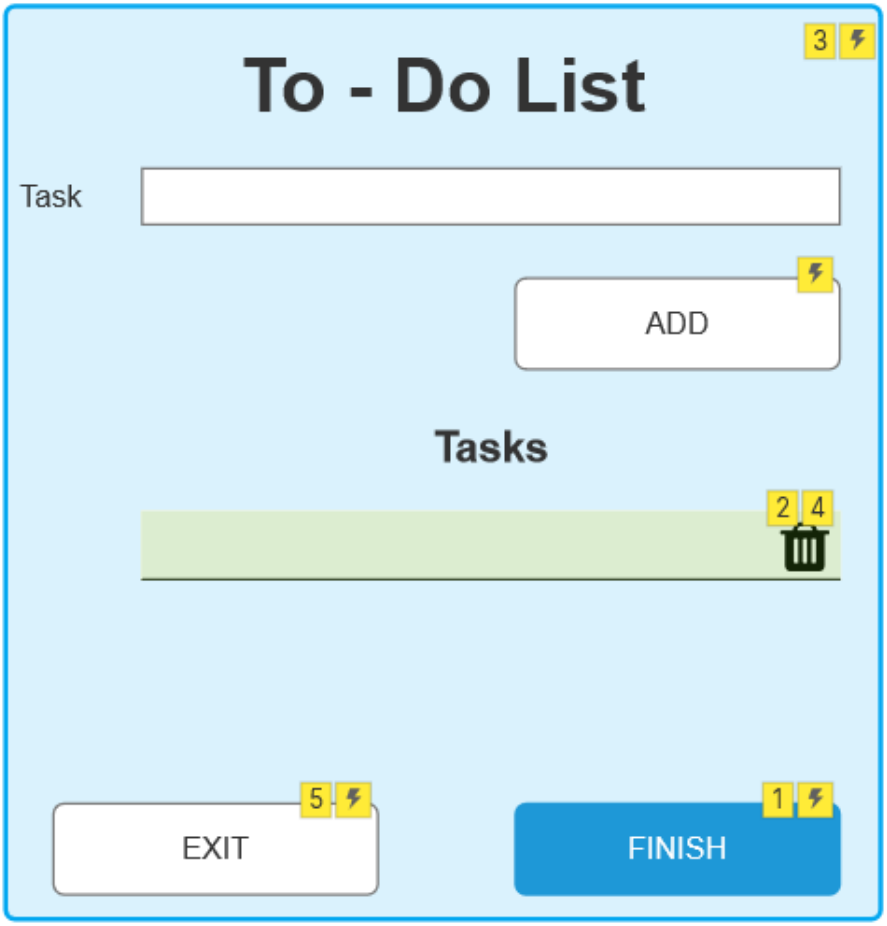
3.1.2.13. Add Checklist Entry

Functionality:
As the main user, I can add a checklist calendar entry.

Normal Flow:
The system displays an empty list of tasks. The actor adds a new task to the checklist. The system adds the new task to the checklist. The actor saves the checklist. The system displays the checklist on the calendar view.

Alternatives:
Remove task: The actor can remove tasks from the checklist.
Cancel: The actor cancels the creation of a checklist. The system will return to the add calendar entry view.

Additional remarks: The actor should be able to move the order of tasks in the checklist around by dragging them around.



3.1.2.14. Manage Medication

Functionality:
As a main user, caretaker or doctor, I am able to manage the medication list, including viewing, adding, editing and deleting medications.

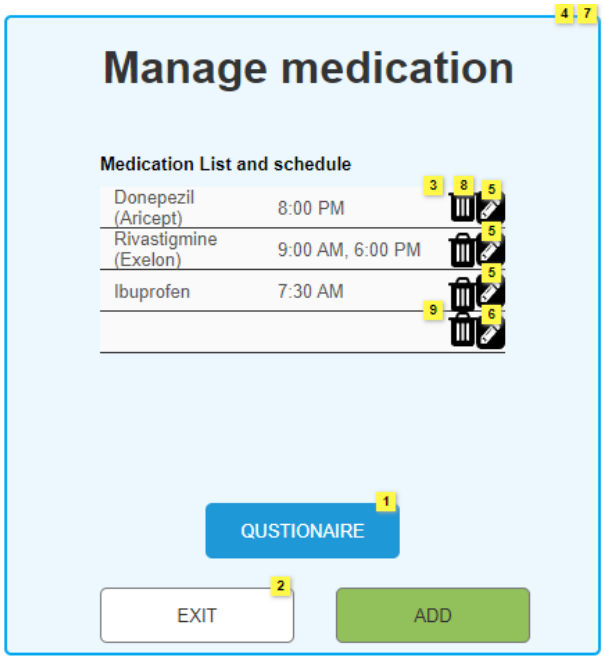
Precondition:
The user is logged into the system.

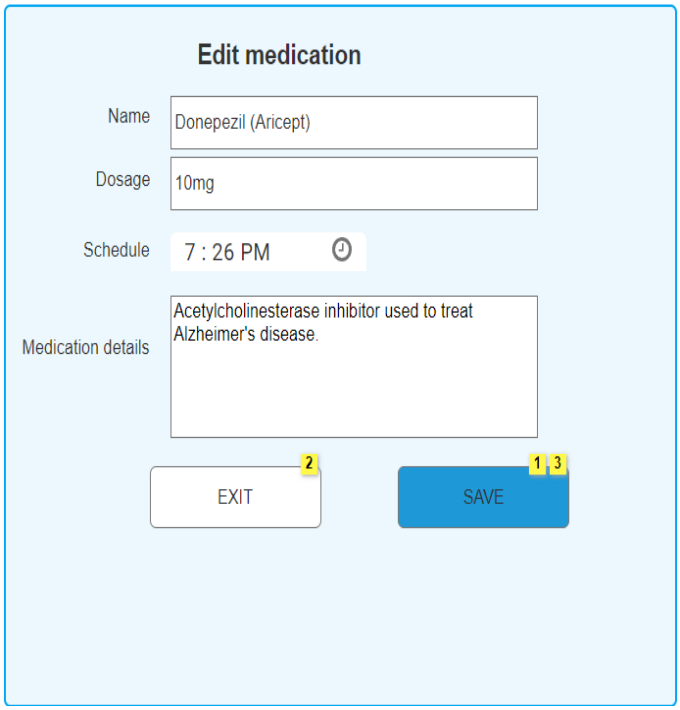
Normal flow:
The system displays a list of details regarding the medication along with their mandatory scheduled dosages. The actor can click on the medication field to see the details of a specific medication without the capability to edit or delete.

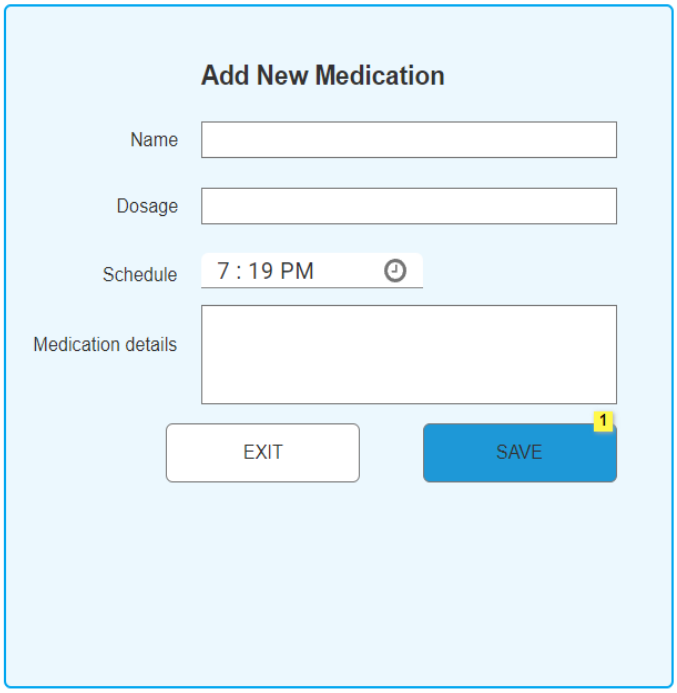
Alternatives:
Edit Medication (with permission): The actor with the necessary permissions, can click on the edit icon and modify the name, dosage, schedule, medication details of a specific medication. Upon confirmation the actor receives a confirmation prompt and upon confirmation they are sent back to the manage medication page and the list is updated.

Add New Medication: The actor is able to add a new medication upon clicking the add button which will display the field's name, dosage, schedule, and medication details. Upon confirmation the new medication is added to the medication details list.

Delete Medication: The actor with the necessary permissions is able to click the delete icon to remove the specific medication from the list.
Please note: Medication that gets deleted is instead made inactive. This will retain the dosage and schedule so it can be used to learn from in the future.




Edit medication
Name
Donepezil (Aricept)
Dosage
10mg
Schedule
7 : 26 PM
Medication details
Acetylcholinesterase inhibitor used to treat Alzheimer's disease.
2
EXIT
1 3
SAVE


Add New Medication
Name
Dosage
Schedule
7 : 19 PM
Medication details
EXIT
1
SAVE

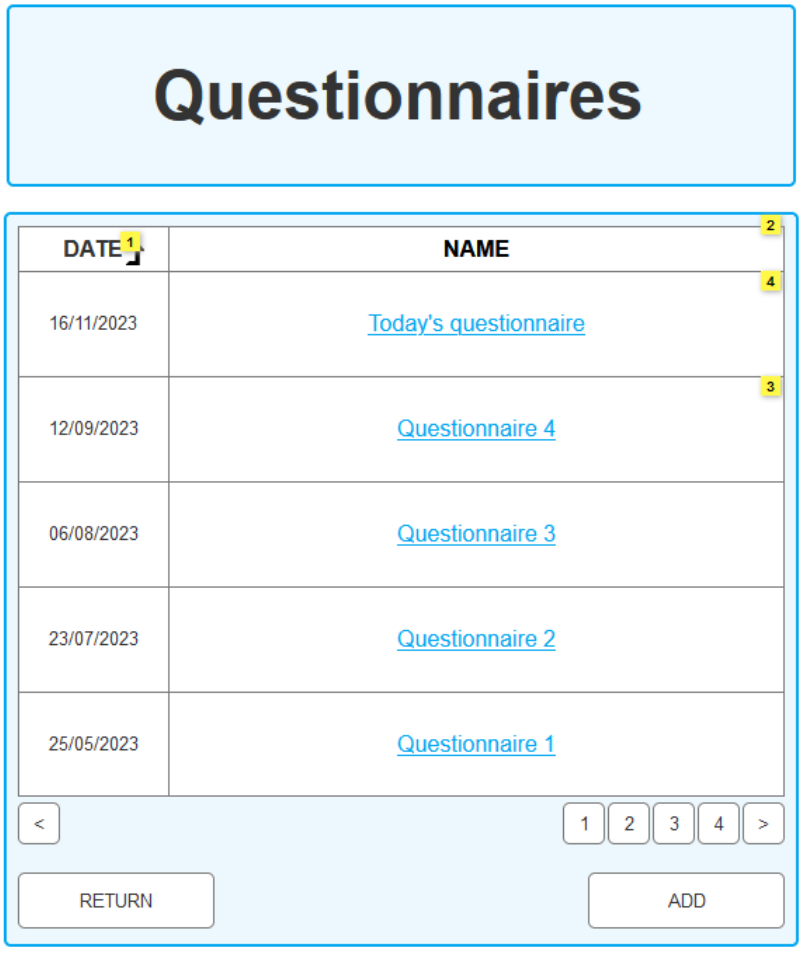
3.1.2.15. Manage Questionnaires

Functionality:
As a caretaker or doctor, I can manage the medical questionnaires.

Precondition:
The user is logged in as a caretaker or doctor.

Normal flow:
The system displays the list of active questionnaires. The actor selects a questionnaire. The system shows the fill in questionnaire use case.

Alternatives:
Add: The actor can choose to add a new questionnaire.

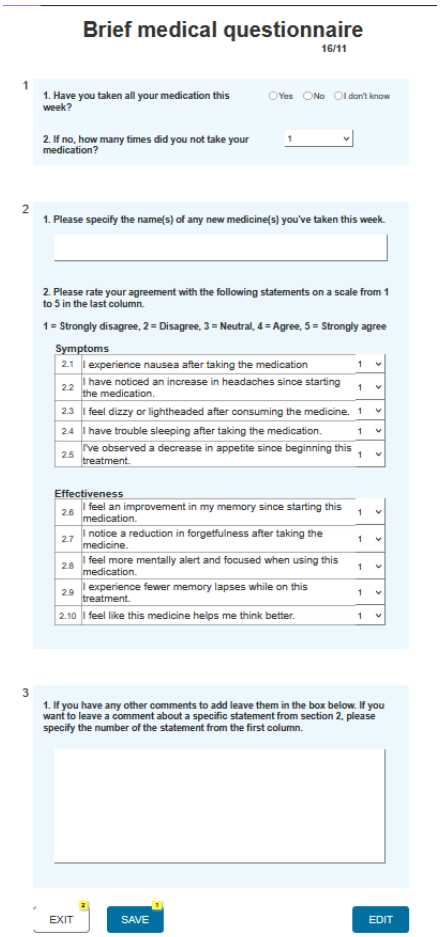


3.1.2.16. Fill in Questionnaire

Functionality:
As the main user, I can fill in questions that are in the questionnaires.

Precondition:
The user is logged in as the main user.

Normal Flow:
The system displays a list of active questions. The actor fills in the questions and saves the answers. The system saves the questionnaire and returns the user back to the manage questionnaire page.
Alternatives:
Edit: The actor can edit the questionnaire. Adding, removing or changing questions.
Please note: Removing questions will make them inactive, so they can be reviewed in the future if necessary.
Exit: The actor can back out of the questionnaire. Current answers will still be saved.



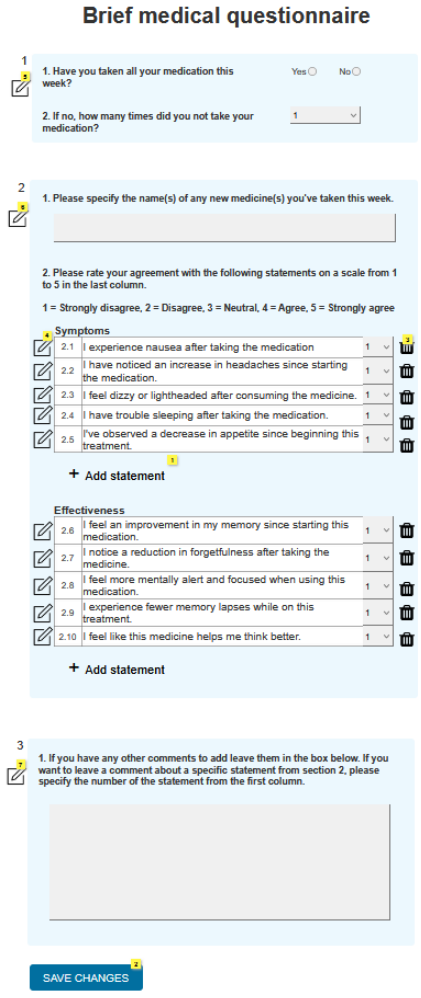
3.1.2.17. Edit Questionnaire

Functionality:
As a doctor, I can edit questionnaires.

Precondition:
The user is logged in as a doctor or main user.

Normal Flow:
The system displays a list of active questions. The actor adds a new statement. The system shows the new statement in the list of active questions. The actor saves the changes. The system updates the questionnaire and returns the actor to the fill in questionnaire page.

Alternatives:
Edit Question: The actor can choose to edit an already existing question.
Delete Question: The actor can choose to delete one of the active questions.
Please note: Deleting a question after the questionnaire is initial creation will make it inactive instead of deleting it completely.



3.1.2.18. Manage Users

Functionality:
As an administrator, I can set specific user rights in order to control their access and permissions within the system.

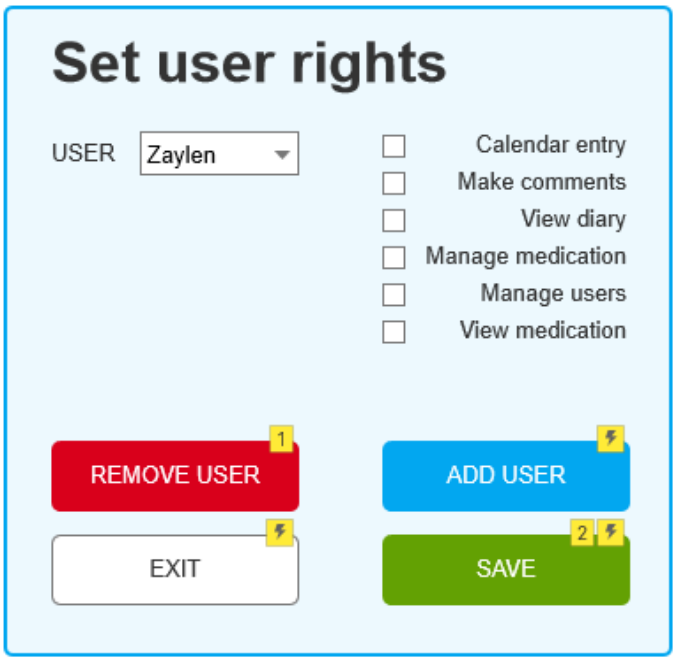
Precondition:
The Main user is logged in as the administrator.

Normal flow:
The Actor accesses the “set user Rights” functionality. The system presents a list of all the users. The actor selects a user from the list. The system displays all the functionalities associated with the selected user. The actor has the option to deselect or select different functionalities for the user.

Alternatives:
Remove User: The actor can remove user when the “REMOVE USER” prompt is given by the system. If confirmed, the selected user is removed from the system.

Add User: The actor can add a user in the system which would navigate to the user creation interface.

Save Changes: The actor clicks the “Save” the system saves the modified user rights. If any changes are made, the system is prompted for confirmation before saving.

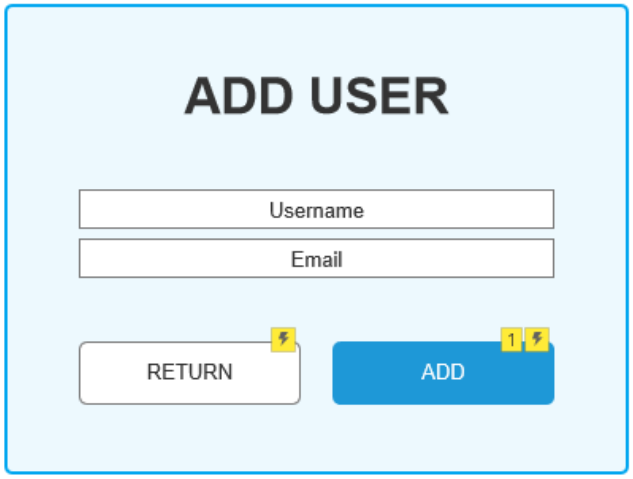


3.1.2.19. Add user

Functionality:
As the main user, I can add a user to the system.

Normal flow:
The system asks for the username and email address of the user you want to add to the system. The actor fills in the information and adds it to the system. The user that was added gets an email notifying them that they were added together with a confirmation link.

Alternatives:
Cancel: The actor can at any point cancel the process of adding a user.



3.1.2.20. Manage digital Diary

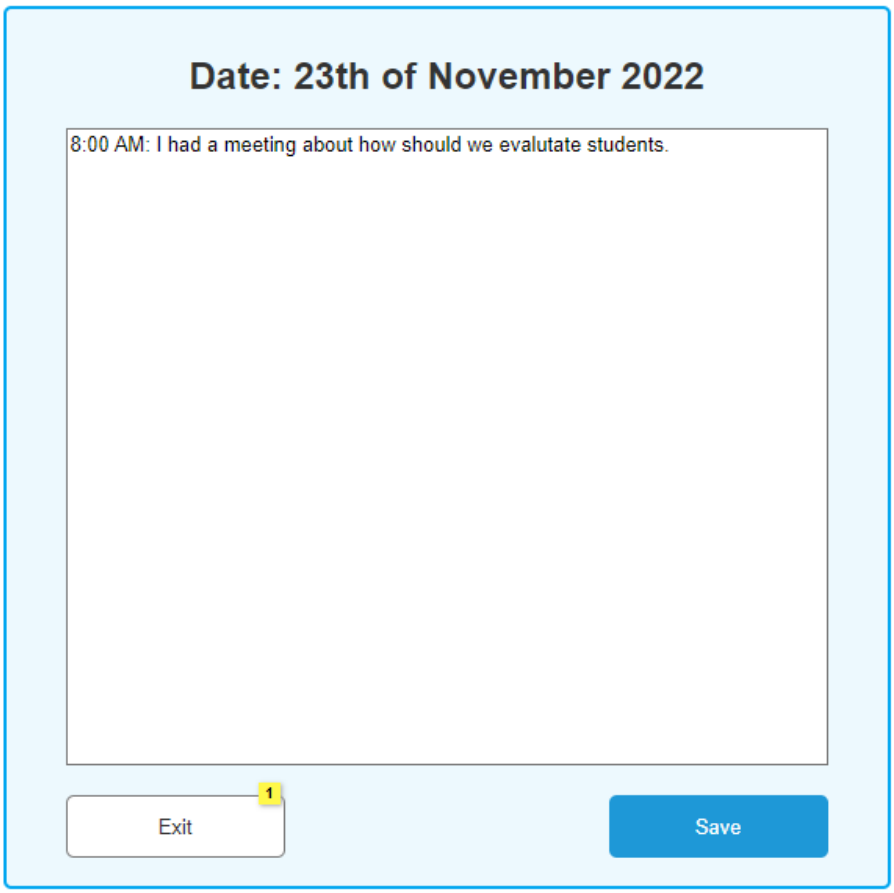
Functionality:
As the main user, I am able to manage my digital diary by adding, reviewing and also saving diary entries.

Precondition:
The main user is logged into the system.

Normal Flow:
The system displays the actor with a digital diary entry form, displaying fields for date, time, location, text and existing journal entries in chronological order, with the latest entry at the top. The actor adds a new entry by typing it in the textbox, such as the date, time, location, and their thoughts or experiences for the day. Upon confirmation the system updates the entry form saving the diary entry, timestamp in it with date, time, and the moment of the day.

Alternatives:
Modify: The actor has the ability to choose to modify and edit a diary entry.

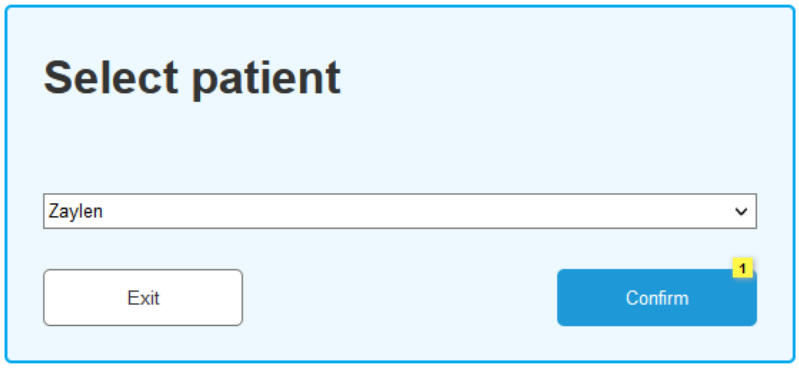
View digital diary: The actor can view a diary entry by selecting the date, time or moment they want to see the digital diary from.



3.1.2.21. Select Patient

Functionality:
As a doctor, I can select which patients' medication I will view.

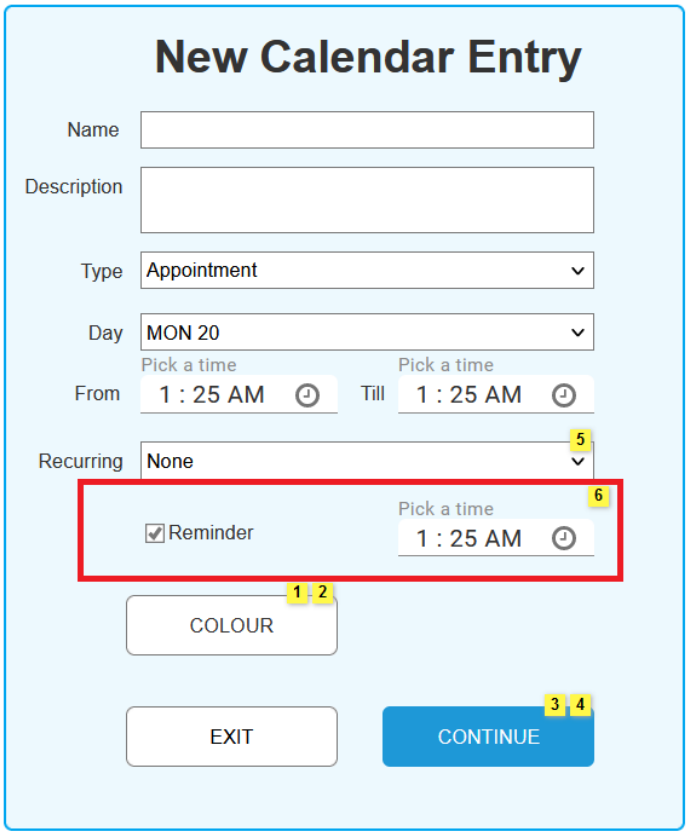
Normal flow:
The system displays a dropdown menu with all available patients for the user. User selects that patients' medication they want to view. The System redirects the user to the selected patient medication.



3.1.2.22. *Send reminder*

Functionality:
As time, I can send reminders.

Normal flow:
Actor initiates the “Send Reminders” process when it reaches a predefined scheduled time for a reminder. The system delivers the reminder to the main user at the scheduled time.



3.1.2.23. *Send notification*

Functionality:
As time, I can send notifications.

Normal flow:
Actor initiates the “Send Notifications” process when the main user has not taken their medication yet. The system delivers the notifications to the main user and the caretaker after a certain amount of time and continues to do so as long as the main user or caretaker has not responded to the notification.

3.2. Non-Functional Requirements

3.2.1 Implementation

To guarantee a robust and user-friendly application, the system's implementation must comply with the following non-functional criteria.

3.2.1.1 Hardware and System Software

The program is compatible with typical web servers such as Apache or Nginx and for server-side processing, PHP 7.4 or later is necessary. Also, the program is meant to operate on Windows, Linux-based operating systems and IOS for best performance.

3.2.1.1. Programming Language

The creation of the application will be carried out with PHP (Laravel framework) for server-side scripting. Furthermore Laravel 8.x will be utilized for its rich features.

3.2.1.2. Database

MySQL will be used as the relational database management system for the application and for sensitive user information saved in the database, data encryption will be applied.

3.2.1.3. Guest User Access

The system should allow guest users to comment on entries without requiring a login. This feature increases user engagement and encourages a broader audience to participate.

3.2.1.4. Password Reset Functionality

The application must offer a password reset function to improve the user experience. Users must be able to reset their passwords using either their username or email address, ensuring accessibility in the case of lost password.

3.2.1.5. Email Notifications to Doctor

The system should manage email notifications to the doctor wisely. Instead of sending alerts for each completed questionnaire, a single email should be sent before the planned session. This method reduces email frequency while maintaining effective communication.

3.2.2. Quality Requirements

To ensure a high-quality user experience and application performance, the system must fulfill the following quality standards.

3.2.2.1. Accessibility

The application's accessibility features should be prioritized. A "Forgot Password" option that allows users to reset their password using their username or email address should be offered. This feature guarantees that users may easily restore access to their accounts.

Additionally, if the user wishes to, they can link their phone number to the account and request an access code instead of having to use a password.

3.2.2.2. Usability

The usability of the program is critical for a great user experience. It should always open on the current week, simplifying user interactions. Additionally, users should be able to drag and drop things in the morning routine list or checklist to rearrange them, improving the application's overall usability.

3.2.2.3. Security

Security is of the utmost importance. Confirmation pop-ups for crucial procedures such as altering prescription data should be created to avoid unintentional or illegal acts. This assures that users perform deliberate actions, which reduces the possibility of mistakes.

3.2.3. Database Accessibility

The database containing completed surveys must be available to authorized users, which must include both the doctor and the main user (and sometimes the caretaker). Users should be able to examine and edit previous questions, increasing openness and user control.

3.2.4. Notifications

A notification mechanism should be included in the system to notify users when a checklist is left unfinished. Timely notifications remind users to complete necessary tasks, resulting in a more organized and efficient user experience.

3.2.5. User Interface Consistency

Maintaining consistency in the user interface across several user roles, including the main user, doctor, and caretaker, is critical. A consistent and intuitive user interface contributes to a smooth user experience.

3.2.6. Performance

The application's performance should be satisfactory, with quick responses for user interactions. Optimal performance adds to a responsive and effective user interface.

3.2.7. Historical Data

Entries that were created in the calendar should be automatically archived. The user should be able to look back to entries from a year ago. Entries older than a year should be automatically archived.

4. Priority by functionality

In this section, we listed all the functionalities / use cases which were ordered by MoSCoW-priority. The use cases are split up into groups of which are absolute must haves to nice to have or optional features.

4.1. Must have this

As a user, I need to login, so that I can access the calendar. As a user, I need to register new user, so that I can use the application. As an acquaintance, I need to navigate calendar, so that I can access content from different days. As an acquaintance, I need to view calendar entry, so that I can view that task related to the entry. As an acquaintance with a role, I need to edit calendar entry so that I can edit the task related to the entry. As a caretaker, I need to add calendar entry, so that I can add content to new timeslots. As a main user and doctor, I need to manage medication, so that I can change medications. As a main user, I need to manage digital diary, so that I can write down my thoughts on day-to-day basis.

4.2. Should have this if at all possible

As an acquaintance, I need to edit settings, so that I can manage my applications environment. As a main user and doctor, I need to manage questionnaires, so that I can review older questioners. As a main user, I need to fill in questionnaire, so that I keep track of answers to medical related questions by the doctor. As a main user and doctor, I need to edit questionnaire, so that I can change the questions in the questionnaire. As the system, I need to send reminder, so that I can remind the users about future or current entries that need to be fulfilled. As the system, I need to send notification, so that the users are notified about missed fulfillments of entries.

4.3. Could have this if it does not affect anything else

As an acquaintance, I need to select calendar, so that I can view other available calendars. As a main user, I need to manage users, so that I can control who has access to my calendar. As a main user, I need to add user, so that I can allow other users to view my calendar. As a doctor, I need to select patient, so that I can view and manage their medication of the correct patient.

4.4. Would like to have but will not have this time around

As a guest or an acquaintance, I need to make comment, so that I can add a comment onto a calendar entry to inform other permitted users about something. As an acquaintance, I need to select background color, so that I can personalize the application. As an acquaintance, I need to select theme, so that I can personalize the application.