User Stories, Acceptance Criteria and Test Cases

The following is the link for our jira board for reference:

https://almost5.atlassian.net/jira/software/projects/A5/boards/1/roadmap

Epic: Sign-Up

Task: Doctor's Sign-Up

User story:

As a doctor,

I want to be able to sign-up on the app,

So that I am able to help and connect with new and existing patients.

Acceptance criteria:

Given that I have successfully downloaded the app and have an active internet connection,

When I choose to sign-up as a doctor,

And I enter my email address, provide my doctor's ID and set a valid password for my account,

Then my doctor's account is created on the app.

Test Cases:

- Test 1:
  - Input - The email address used for signup already is used in an existing account
  - Output - The field would be highlighted in red and the system would prompt "The email being used already has an existing account".
- Test 2:
  - Input - The password set does not match the criteria - minimum length of 8 characters (a-Z, 0-9, special symbols, etc.)
  - Output - The password field will be highlighted with the system prompt "Invalid password".
- Test 3:
  - Input - The doctor's ID provided is fake.
  - Output - The system will inform the user that the ID is not verified hence the account cannot be created.

Sub Tasks

- Create Front-end
    - Create UI for the page
    - Create information input fields
    - Create check-boxes and add button
- Save information in a database
    - Link the information in the fields to a database
    - Correctly store and update information in that database

Task: Patient’s Sign-Up

User story:

As a patient,

I want to be able to sign-up on the app,

So that I am able to access the app’s functionality and be able to connect with the preferred doctor for consultation if needed.

Acceptance criteria:

Given that I have successfully downloaded the app and have an active internet connection

When I choose to sign-up as a patient

And I enter my email address and set a valid password for my account

Then my patient account is created on the app.

Test Cases:

- Test 1:
    - Input - The email address used for signup already is used in an existing account
    - Output - The field would be highlighted in red and the system would prompt “The email being used already has an existing account”.
- Test 2:
    - Input - The password set does not match the criteria - minimum length of 8 characters (a-Z, 0-9, special symbols, etc.)
    - Output - The password field will be highlighted with the system prompt “Invalid password”.

Sub Tasks

- Creating the Front End
    - Having a signup form with the UI to enter the name email and password
- Backend

    - Like the doctor, signup retrieve information from the form
    - Save this info on the database
    - Save the info onto the database

Epic: Login

Task: Doctor’s Login

User Story:

As a Doctor

I want to be able to login into my doctor account online

So that I can communicate with my patients.

Acceptance Criteria:

Criterion 1:

Given that the Doctor has an account and the credentials are correct

When the Doctor requests to login

Then the Doctor is logged into their account.

Criterion 2:

Given that the Doctor doesn't have a valid account

When the Doctor requests to login

Then the Doctor is redirected to the “Sign-up”.

Test Cases:

- Test 1:
    - Input- The credentials used are incorrect.
    - Output - The fields would be highlighted in red and an error message would be prompted “Incorrect email or password, try again.”
- Test 2:
    - Input - The incorrect credentials have been typed for more than 3 times.
    - Output- the account will be locked and the user has to reset the password.

Sub Tasks:

- Create front-end UI
    - Creating a form for the doctor to enter their email address and password to login

    - Creating a link to reset the password for the account if the user forgets their password
    - Login button
- Backend Connection
    - Writing queries to store the information on the database if the user changes their password.
    - Writing queries to retrieve information that is being entered into the form, from the database.
- Authentication
    - Using authentication API, getting the data filled in the form to login, and validating with the details already saved on the database.

Task: Patient’s Login

User Story:

As a Patient

I want to be able to login into my Patient account

So that I can communicate with my Doctors.

Acceptance Criteria:

Criterion 1:

Given that the Patient has an account and the credentials are correct

When the Patient requests to login

Then the Patient is logged into their account.

Criterion 2:

Given that the Patient doesn't have a valid account

When the Patient requests to login

Then the Patient is redirected to the “Sign-up”.

Test Cases:

- Test 1:
    - Input- The credentials used are incorrect.
    - Output - the fields would be highlighted in red and an error message would be prompted “ Incorrect email or password, try again”.
- Test 2:
    - Input - The incorrect credentials have been typed for more than 3 times.
    - Output- The account will be locked and the user has to reset the password.

Sub Tasks:

- Creating front-end UI
  - Similar to the doctor’s login, creating a form for the patient to enter their email address and password to login
  - Creating a link to reset the password for the account if the user forgets their password
  - Login button
- Backend
  - Writing queries to store the information on the database if the patient changes their password.
  - Writing queries to retrieve information that is being entered into the form, from the database.
- Authentication
  - Using authentication API, getting the data filled in the form to login, and validating with the details already saved on the database.

Task: Admin Login

User story:

As an Admin

I want to be able to login into my Admin account

So that I can manage the enterprise.

Acceptance Criteria:

Criterion 1:

Given that the Admin has an account and the credentials are correct

When the Admin requests to login

Then the Admin is logged into their account.

Test Cases:

- Test 1:
  - Input- The credentials used are incorrect
  - Output - The fields would be highlighted in red and an error message would be prompted “Incorrect email or password, try again”.

Sub Tasks:

- Front-end UI
  - Creating a form for the admin to enter their email address and password to login
  - Login button
- Backend and Authentication
  - Writing queries to retrieve information that is being entered into the form, from the database that's specific to the admin information and rights.
  - Using authentication API, getting the data filled in the form to login, and validating with the details already saved on the admin information database to check the admin rights of an account.

Epic: Booking

Task: Calendar View

User-Story:

As a Patient

I want to be able to view all the available times for a particular doctor

So that I can book a new appointment with them accordingly

Acceptance Criteria:

Criterion 1:

Given I have logged in on the app

When I click on ‘Booking’ on my dashboard

And then select the the 'New Booking option'

Then I will be able to select a doctor based on specialization and then view their availability in a calendar format.

Test Cases:

- Input - The patient is not able to see the availability that the doctor has added, on the calendar.
- Output - The test fails.

Sub Tasks:

- Create calendar UI
  - Create a calendar with dates, months and year as shown in the wireframes
  - Display that calendar on the 'New Booking' page

- Link calendar to availabilities database
  - Link the created calendar to the Doctor Availabilities database
  - Highlight the days based on the availability of the particular doctor being searched for

Task: Booking Menu

User Story:

As a Patient

I want to be to select the ‘Booking’ option on my dashboard

So that I can further choose to either create a new booking or view the bookings I already have.

Acceptance Criteria:

Given that I am a patient and have logged in on the app

When I am on the dashboard

And click on the 'Booking' option

Then I should be taken to the booking page where I can either create a new appointment or view my current bookings.

Test-Cases:

- Test 1:
  - Input - When the patient clicks on “Booking” on the dashboard and is not redirected to the booking page
  - Output - The test fails.
- Test 2:
  - Input - When the patient clicks on “Booking” and is taken to the booking page but the menu option there aren’t working
  - Output - The test fails.

Sub-Tasks:

- Create UI for the 'Booking page'

    - Create a button for 'New Booking' and link it to the respective page
    - Create a button for 'View Booking' and link it to the respective page

<u>Task:</u> Book New Appointment

User Story:

As a Patient

I want to be able to select a doctor and book a new appointment with them

So that I can book a new appointment with them accordingly and consult them

Acceptance Criteria

Given I have logged into my account successfully

When I click on ‘Booking’ on my dashboard

And then select the the 'New Booking option'

Then I will be able to select a doctor based on specialization, select a time and book the appointment

Test Cases:

- Test 1:
    - Input - When the patient clicks on “Book Now” and the appointment is not confirmed.
    - Output - The test fails. The user is asked to select the preferred date again to confirm the selection.

- Test 2:
    - Input - The patient tries to book an appointment at a time slot in which they already have some other existing appointment.
    - Output - The booking is not proceeded and the user is informed that they have an existing appointment at the same time and they are redirected to choose another time.

Sub-Tasks:

- Create Front-end
    - Create a list for doctor specializations

    - Create a dropdown menu for displaying the list of doctors

- Create Back-end
    - link each doctor mentioned in specific specializations to ‘New Booking’ page which displays their availability
    - Display their availability in a calendar view such that the patient can book using the CalendarFX API

- Link the new added information to the database
    - Store all the booked appointments information to the appointments database

- Link doctor availabilities database to the page
    - Link the doctor availabilities database to the calendar to display relevant information

Task: View Appointments

User Story:

As a Patient

I want to be able to view my upcoming and past appointments

So that I can schedule, plan and prepare for the consultation accordingly

Acceptance Criteria

Given I have logged into my account successfully

When I click on ‘Booking’ on my dashboard

And then select the the ‘View Booking’ option

Then I will be able to view my upcoming appointments and also the past appointments in a chronological order

Test Cases:

- Test 1:
    - Input - When the patient clicks on “View Booking” and is able to see the bookings but in a shuffled manner

- Output - The test fails.

- Test 2:
  - Input - When the patient clicks on “View Booking” but hasn’t made any bookings yet
  - Output - The test fails. The screen with say, ‘You haven’t booked any appointments yet'

Sub-Tasks:

- Create UI and link to database
  - Create a list for displaying upcoming appointments in chronological order using the sort method in the database
  - Create a list for displaying past appointments in chronological order using the sort function in the database
  - Create a search icon
  - Link the page to booked appointments database to update and retrieve information

Epic: Chat / Communication

Task: Chat UI

User story 1:

As a patient,

I want to be able to text my preferred doctor during my appointment,

So that my health concerns and doubts are resolved through direct communication.

Acceptance Criteria:

Given that I have a confirmed appointment with my preferred doctor,

When I try to open the chat and message the doctor,

Then the doctor is able to receive the texts that I have sent.

User Story 2:

As a doctor,

I want to be able to reply my patients during appointments,

So that the check-up is complete addressing their concerns.

Acceptance Criteria:

Given I have a confirmed appointment with a patient at a particular time,

When I open the chat with the patient,

Then I'm able to see any texts that they have sent and am able to reply back to them.

Test Cases:

- To test the "Send button" -
  - Input - The patient/doctor types in the text and clicks the send button.
  - Output -
    - if the message doesn't go through – test failed
    - if the message goes through from the senders end – test passed
- To test if the message is received -
  - Input - The patient/doctor types in the text and clicks the send button.
  - Output -
    - The receiver is able to see the same message that was sent – test passed
    - the message was not received – test failed
    - the message was not the same as to what was sent – test failed.

Sub Tasks:

- front-end UI
  - The profile picture of the receiver needs to be displayed
  - The chat bubbles need to be colour coded according to the sender and the receiver of the messages
  - The message type box with a placeholder text needs to be shown.
  - The icon to attach files needs to be placed
  - The send button.
- Back-end
  - Using twilio for Chat REST API , the messages will be sent through queries using the resources in this REST API.

__Task:__ Upload file option

User story:

As a doctor,

I want to be able to upload any prescription files,

So that the patient can refer to it during or after consultation.

Acceptance Criteria:

Given that I have had a consultation with a patient,

When I try to upload their prescription file from my system,

Then the patient receives that file and can access it from their end.

Tests:

- To test the file to be uploaded -
    - Input - The file is not in a format which is permitted.
    - Output - The user is informed saying, “format not supported” and is given the option to choose again.

- To test the “upload” button -
    - Input - The file that is selected from the system is not being sent/ shared when the user clicks “upload”.
    - Output - The test fails.

- To test if the uploaded file is received -
    - Input - The file that is being sent from the doctor's side is not reflected on the patient's end or the file that the patient receives is not the same as what the doctor sent.
    - Output - The test fails.

Sub Tasks

- Front End
    - displaying an attach document icon from where the user can attach any file from their system.
- Backend
    - Using twilio - Flex webchat will be used to allow users to send file attachments like pdfs.
    - chat attachments will use Twilio chat media resources for file storage. for the user to use their own storage, we’ll need to replace the sendMediaMessage action which is triggered when the file is sent.

Task: Encryption

User story:

As a patient.

I want that only the appointed doctor is able to read the chats

So that the privacy of my health status and other information is maintained.

Acceptance Criteria:

Given that I have an appointment with a doctor

When I share my health status, updates and concerns with the doctor and the doctor uploads prescription or any other files,

Then I expect that our chat is end-to-end encrypted and no one outside the chat is able to read them except me or the doctor I'm conversing with.

Test Cases:

- To test the security of the message -
  - Input- The patient sent a message to the appointed doctor which is also visible to the admin since the admin has super access to the patient's details/profile.
  - Output- The test fails as the messages from the patient's chat are not end-to-end encrypted.

Sub Tasks.

- Backend
  - Because we are using an API for the chat, Twilio itself supports encryption to protect communications. Twilio supports the TLS cryptographic protocol..

Epic: Patient's Dashboard

Task: Patient's Profile

User Story:

As a Patient

I want to add information about myself

So that the doctor can diagnose me

Acceptance Criteria 1:

Given The patient is signing up on the app

When the patient clicks the "Sign-Up"

Then the patient can add information like name, email address, age, date of birth, Mobile number

Acceptance Criteria 2:

Given the patient is already logged on to the app

When the patient clicks the “Edit Profile”

Then the patient can edit the phone number, email address, etc.

Test Cases:

- Input- The user has not entered compulsory information on the form.
- Output- he isn't allowed to proceed.

Sub Tasks.

- Creating UI
    - Create entry fields for the user to enter
        - name
        - phone no
        - address
        - date of birth
        - blood group
        - allergies
        - emergency contact
- Storing information on the database
    - Writing code to retrieve and store the information into a database

Task: Health Record

User Story:

As a Patient

I want to add information about my health like lab tests and symptoms

So that the doctor can treat me according to my conditions

Acceptance Criteria 1

Given The patient is logged on the app

When The patient selects “Health Record“ from the dashboard

Then Show the patient options to add “Symptoms“, “Reports/tests“, and update “Health status“

Acceptance Criteria 2

Given The patient is on “Health Record“

When The patient goes on to add “Symptoms“

Then Show a text box to write down their symptoms.

Acceptance Criteria 3

Given The patient is on “Health Records“

When The patient selects to add in “Report/tests“

Then Show an option to attach files in pdf format only and “upload“ after done.

Acceptance Criteria 4

Given The patient is on “Health Record“

When The patient selects “Health status“

Then the patient can enter his current health situation (not more than 50 words) eg ”I am feeling better my fever has gone.”

Test Cases:

Test 1:

- Input - The patient tries to upload a file of a format other than pdf
- Output - Error message of “Enter file in pdf format only“.

Test 2:

- Input - The patient tries to enter more than 50 words in “update health status“
- Output - Show error message “Enter 50 words or less”.

Sub Tasks.

- Creating The UI
  - Creating tying fields for entering symptoms
  - Creating add button to upload file
  - Creating textbox field to enter the health status
- Backend
  - Storing the information and entering it into the database
  - Integrating add file option to the health record section

Task: Medicine

User story:

As a Patient,

I want to add the latest medicine schedule to my dashboard as prescribed by my doctor

So that I can view the medicines, its type and its time and duration

Acceptance criteria:

Given I have logged in and am on my dashboard

When I click on ‘Medicines’ icon

And add all the required medicine details

And click on ‘Add medicines’ icon

Then I should be able to add medicine name, its type and time to take and duration

Test cases:

- Test 1:
  - Input: The medicine name entered is invalid
  - Output: A prompt will pop up saying that, ‘The medicine name is invalid’
- Test 2:
  - Input: The time entered is invalid
  - Output: A prompt will pop up saying that, ‘The time entered is invalid’

Sub Tasks

- Creating The UI
  - Having the medicine section in place
  - Fields to enter the medicine name type duration and usage
  - Add new button works
- Backend
  - Storing the entered data using queries in the backend
- Scheduling the medicine times by adding clock/time api
  - Allow scheduling the medicine using the CalnderFX API

Task: Prescription

User Stories:

As a Patient

I want to keep an archive of my previous prescriptions

So that I can view my prescriptions later (even after the consults)

Acceptance Criteria 1:

Given The patient has done a consult

And the doctor has uploaded a prescription

And the patient downloaded the prescription

When the patient clicks the “Prescriptions” from the dashboard

Then Show the patient an option to upload a file

And also enter the date of the consult, name of the doctor

Acceptance Criteria 2

Given The patient already has one prescription uploaded

When the patient clicks the “Prescriptions” from the dashboard

Then Show the patient option “Add prescription“

Test Cases:

- Input - The patient tries to upload a file of a format other than pdf
- Output - Error message of “Enter file in pdf format only“

Sub Tasks

- Create Front end
  - Enable a section to add files (Add button)
  - Enable option to add date
- Create Backend
  - Add the Twilio API to integrate to upload file option
  - Save the files on the database (Same as chat upload file option)

Epic: Doctor’s Dashboard

Task: Doctor Profile

User Story:

As a Doctor,

I want to create a profile for myself on the telemedicine app,

So that I can my patients can view relevant information about me

Acceptance criteria:

Given that I have installed the telemedicine app and signed up for that

When I am on my dashboard and click on 'Profile'

Then I can fill-in or edit relevant information about myself in that section

Test cases:

- Test 1 -
  - Input: The date of birth entered is invalid
  - Output: A prompt will be shown on the screen saying that 'The date entered is invalid'.
- Test 2 -
  - Input: The location/address entered is invalid
  - Output: A prompt will be shown on the screen saying that 'The location/address entered is invalid'.
- Test 3 -
  - Input: The specialization entered is invalid
  - Output: A prompt will be shown on the screen saying that 'The specialization entered is invalid'.
- Test 4 -
  - Input: The Description entered exceeds the word limit of 50 words
  - Output: A prompt will be shown on the screen saying that 'The description has exceeded the word limit. Please Do not write more than 50 words'.

Sub Tasks

- Create Front end
  - Design the UI of the page
  - Add fields for adding information
  - Add icon to upload and set a profile picture
- Save information in a Database
  - Link the information input field to the database
  - Save the information in correctly in the database

Task: View Appointments

User story:

As a Doctor,

I want to view the list of all appointments sorted in a chronological order

So that I can view the patient history and plan my schedule accordingly

Acceptance criteria:

Given that I have logged in to the telemedicine app and am currently at the dashboard

When I click on the 'Appointments' tab

Then a list of all the appointments is shown along with the name of patient and time.

Test cases:

- Test -
  - Input: Clicking on the 'Appointments' tab but there are no appointments scheduled
  - Output: Output message is shown up on the screen saying that 'There are no upcoming appointments'.

Sub Tasks

- Create Front end
  - Create the UI of the page
  - Create a table to display appointment information
  - Create a search bar
- Link database to the front-end table
  - Link the database with appointments information with the table on the page
  - Display the sorted information in the table
- Implement Search
  - Add code to search from the database and display on the screen
  - Write database queries for the function

Task: View Patient List

User story:

As a Doctor,

I want to view a list of patients consulting me with links to their health update section/profile

So that I can view their latest health update as well as health history

Acceptance criteria:

Given I have logged in and am on the dashboard of the telemedicine app

When I click on ‘My Patients’ tab

Then it should display a list of patients with links to their health update section/profile.

Test cases:

- Input: Clicking on the ‘Patients’ tab but the list doesn’t show up.
- Output: Output message is shown up on the screen saying that ‘The ‘Patients’ page is not responding, please try reloading the window.

Sub Tasks

- Create Front end
  - Create the UI of the page
  - Create boxes to display patient information
- Link database to the front-end table
  - Link the patient information database to the page
  - Display each patient’s information from the database in separate boxes on the page
- Link the Health Record page for each patient
  - Link the each patient information box to the patient’s health record page

Task: Add Availabilities

User story:

As a Doctor,

I want to add my availability for consultations

So that the patients book consultations according to those timings

Acceptance criteria:

Given that the doctor is logged into the app

When the doctor selects “Availabilities“

And selects the edit option in “Set Your Availability“

Then the doctor is given an option to “tick“ for the days he is available and also set time for those days. Any day that is not “Ticked“ is set as unavailable.

Test cases:

- Input: The doctor enters the time out of bound eg 25:00
- Output: Output message is shown up on the screen saying ‘Please enter valid time”

Sub Tasks

- Create Front end
  - Create UI for the page
  - Create information input fields
  - Create check-boxes and add button
- Save information in a database
  - Link the information in the fields to a database
  - Correctly store and update information in that database

Epic: Admin’s Dashboard

Task: Requests

User story:

As a Doctor,

I want to view a list of patients consulting me with links to their health update section/profile

So that I can view their latest health update as well as health history

Acceptance criteria:

Given I have logged in and am on the dashboard of the telemedicine app

When I click on ‘My Patients’ tab

Then it should display a list of patients with links to their health update section/profile.

Test cases:

- Input: Clicking on the ‘Patients’ tab but the list doesn’t show up.
- Output: Output message is shown up on the screen saying that ‘The ‘Patients’ page is not responding, please try reloading the window.

Sub Tasks

- Create the UI
  - Have a list structure for showing the fields

    - Have a search button on top
- Backend Connection
    - Query to fetch data and print them in a list structure
    - To enable the search button to look up specific words

<u>Task:</u> Doctors

User Story:

As an Admin

I want to view a list of current doctors on the app

So that I can keep a track of their details

Acceptance Criteria:

Given the admin is logged into the app

When the admin clicks “Doctors“ on the dashboard

Then the admin is shown a list of doctors on the app with their name, id, specialization

And shows a details button which shows details like the number of appointments conducted, ratings

And there is a search option to search for a specific doctor

Test Case:

- Input - The admin searches for a doctor's name that is not in the database
- Output - An error message is shown “Doctor not found, Try Again!“.

Sub Tasks

- Create the UI
    - Have a list structure for showing the fields
    - Have a search button on top
- Backend Connection
    - Query to fetch data and print them in a list structure
    - To enable the search button to look up specific words

Task: Patients

User Story:

As an Admin

I want to view a list of patients on the app

So that I can keep a track of their details

Acceptance Criteria:

Given the admin is logged into the app

When the admin clicks “Patients“ on the dashboard

Then the admin is shown a list of patients on the app with their names, id, no of appointments so far

And shows a “View“ button which shows the patient’s profile

And there is also a search option to look for specific patient

Test Case:

- Input - The admin searches for a patients name that is not in the database
- Output - An error message is shown “Patient not found, Try Again!”

Sub Tasks

- Create the UI
  - Have a list structure for showing the fields
  - Have a search button on top
- Backend Connection
  - Query to fetch data and print them in a list structure
  - To enable the search button to look up specific words

Task: Appointments

User Story:

As an Admin

I want to view a list of appointments on the app

So that I can keep a track of details for the same

Acceptance Criteria:

Given the admin is logged into the app

When the admin clicks “Appointments“ on the dashboard

Then the admin is shown a list of appointments on the application organized in a chronological order

And it has details of the patient-doctor, date of appointment, and time of appointment

And also a search bar to search for the specific appointment

Test Case

- Input - The admin searches for an appointment that is not in the database
- Output - An error message is shown “Appointment not found, Try Again!”.

Sub Tasks

- Create the UI
  - Have a list structure for showing the fields
  - Have a search button on top
- Backend Connection
  - Query to fetch data and print them in a list structure
  - To enable the search button to look up specific words

Task: Profile

User Story:

As a Admin

I want to add information about myself

So that I can be verified as an admin who can manage the app users.

Acceptance criteria:

Given that I have installed the telemedicine app and login with the admin account

When I am on my dashboard and click on 'Profile'

Then I can fill-in or edit relevant information about myself in that section.

Test case:

- Input- The admin has not entered compulsory information on the form.
- Output- he isn't allowed to proceed.

Sub Tasks

- Creating UI and Backend Store Information
    - Have a form set up to fill in relevant details
    - Have a query to store info entered into the database

Epic: Notifications

Task: Chat Notification

User Story 1:

As a patient,

I want to get notified whenever I receive a message on the app

So that I can read important messages on time

User Story 2:

As a Doctor,

I want to get notified whenever I receive a message on the app

So that I can read urgent messages on time and act accordingly

Acceptance Criteria 1:

Given I am a patient and I have logged in to my account

When I click on ‘Notifications' and then select 'Chat’

Then I will be able to toggle the notification settings (on or off) for the chat

Acceptance Criteria 2:

Given I am a Doctor and I have logged in to my account

When I click on ‘Notifications' and then select 'Chat’

Then I will be able to toggle the notification settings (on or off) for the chat

Test Cases:

- Input: No notification received for the recent chat message.
- Output: The test fails. The user should cross check if they have turned on the chat notifications from the “Notifications” tab on the dashboard.

Sub Tasks:

- Front-end
  - adding the toggle buttons to turn the notifications on and off.
  - the message notification preview
- Back-end
  - Twilio as Notification API - Programmable Chat integrates APN (iOS) and GCM/FCM (Android and browsers) Push Notifications.

Task: Medicines Notifications

User Story 1:

As a patient,

I want to get notified for taking medicines on the scheduled time

So that I can be reminded to take medicines and don't miss out on any dose.

Acceptance Criteria:

Given I am a patient and I have the ND Telemedicine on my phone

And I have successfully logged into my account

When it's the time for me to take a medicine

Then I will be able to receive a notification (reminder) for that medicine showing its name, time and dosage

Test Cases:

- Test 1:
  - Input: No notification received for the medicine on time
  - Output: The test fails. The user should cross check if they have turned on the medicine reminder from the "Notifications" tab on the dashboard.
- Test 2:
  - Input: Notification received with wrong medicine name/dosage
  - Output: A prompt will be displayed on the screen asking to 'Check Medicines information on the patient's dashboard'.

Sub Tasks:

- Front-end and backend
  - the toggle button
  - Notification API twilio - Programmable Chat integrates APN (iOS) and GCM/FCM (Android and browsers) Push Notifications.

Task: Appointment Notification

User Story 1:

As a patient,

I want to get notified 1 hour before the scheduled appointment time with the doctor

So that I can prepare for the appointment meeting and be on time.

User Story 2:

As a Doctor,

I want to get notified 1 hour before the scheduled appointment with the patient

So that I can prepare for the appointment meeting and be on time.

Acceptance Criteria:

Given I am a patient and I have the ND Telemedicine on my phone

And I have successfully logged into my account

When it's 1 hour left to my scheduled appointment with the doctor

Then I will be able to receive a notification (reminder) for that appointment showing the name of the doctor and time

Given I am a Doctor and I have the ND Telemedicine on my phone

And I have successfully logged into my account

When it's 1 hour left to my scheduled appointment with the patient

Then I will be able to receive a notification (reminder) for that appointment showing the name of the patient and time

Test Cases:

- Input: No notification received for the chat message received
- Output: The test fails. The user should cross check if they have turned on the appointment reminder from the “Notifications” tab on the dashboard.

Sub Tasks:

- Front-end
  - the toggle button
  - the medicine notification preview that the user will get saying “Reminder: You have an upcoming appointment in an hour!”
- Back-end
  - Retrieving the appointment details from the database.
  - Twilio as Notification API - Programmable Chat integrates APN (iOS) and GCM/FCM (Android and browsers) Push Notifications.

Task: Patient Health Update

User Story:

As a Doctor,

I want to get notified whenever my patient updates their health status

So that I can be updated with the current situation and advise accordingly

Acceptance criteria:

Given I am a Doctor and I have the ND Telemedicine on my phone

And I have successfully logged into my account

When my patient updates their health status

Then I will be able to receive a notification for that change

Test Cases:

- Input: No notification received for the update in the patient’s health status
- Output: Please check your Health Status Notification settings.

Sub Tasks:

- Frontend - Health update page
  - create a form for the patient to add their health information.
  - uploading file icon for “reports”.

- Toggle
    - adding the toggle button and linking it to the backend and api
- Backend
    - Retrieving the patient’s health details from the database.
    - Twilio as Notification API - Programmable Chat integrates APN (iOS) and GCM/FCM (Android and browsers) Push Notifications for the doctor for when the patient updates their status.