

DigiMed SRS

1.1

Scenario: A patient user needs to see their own profile and their medical data, so they open the app and navigate through the different sections in their profile.

Function: Access patient profile

Description: allow patients to access and navigate through their profile data, including their medical records, medical history and prescription history.

Inputs: none

Source: N/A

Outputs: none

Destination: N/A

Action: display the appropriate profile sections when prompted

Requirements: none

Pre-condition: The patient must have signed up for this system and must be logged in.

Post-condition: none

Side effects: none

1.31 (ADAM)

Scenario: Following an appointment, Rana receives notification through the app to pay her doctor's charge. She picks the "Pay Now" option, enters her credit card information, and verifies the transaction. The system conducts the transaction, sends Rana a receipt, and adjusts her account to reflect that the bill was paid.

Function: Online payments.

Description: enables patients to use integrated payment systems to pay online for their services and doctor's visits.

Inputs: Payment information and appointment or service details.

Source: Patient input through the app.

Outputs: Payment confirmation.

Destination: Payment processing module.

Action: Process the transaction through a secure payment gateway.

Requirements: Secure integration with a payment service provider.

Pre-condition: Valid patient account and secure authentication.

Post-condition: Payment is processed, and receipt is issued.

Side effects: None.

1.34 (ADAM)

Scenario: Ahmed accesses the "My Prescriptions" area of the Digimed app after signing in. A list of his current medicines, along with dose guidelines and due dates for refills, is visible to him. To see further information, including any doctor's notes pertaining to the drug, he clicks on the prescription.

Function: Access prescriptions.

Description: Patients can view and manage their prescriptions online.

Inputs: Patient ID, prescription data.

Source: Doctor's prescription records.

Outputs: List of active prescriptions.

Destination: Patient's app interface.

Action: Fetch and display all active prescriptions for the patient.

Requirements: Data security and compliance with healthcare regulations.

Pre-condition: Valid patient account and authentication.

Post-condition: None.

Side effects: None.

1.41 (ADAM)

Scenario: Sarah's visit with Dr. Green is set for next week. Through the Digimed app, she is notified that the Doctor had to postpone the appointment. The message offers choices for rescheduling and makes it clear that doctors have the right to cancel appointments. Sarah decides to reschedule the appointment.

Function: Appointment management notifications.

Description: The system must notify patients that appointments can be canceled by doctors and inform them promptly if a cancellation occurs.

Inputs: Appointment status.

Source: Doctor's input (appointment management).

Outputs: Cancellation notification.

Destination: Patient's app interface.

Action: Notify patients immediately if an appointment is canceled by the doctor.

Requirements: Real-time notification system.

Pre-condition: Valid patient and doctor accounts.

Post-condition: Patients are aware of appointment cancellations.

Side effects: None.

1.5

Scenario: A patient is looking for a doctor and wants to access a doctor's profile to decide if they will book with them or not

Function: Access Doctor Profile

Description: A patient can access a doctor profile and view any information that the doctor published on his profile

Source: input by patient

Outputs: doctor's profile

Destination: Displayed on the patient's device.

Action: redirect the patient to the link of the doctor's profile

Requirements:input by patient

Pre-condition:The user is logged in as a patient

Post-condition: N/A

Side effects: N/A

1.6

Scenario: A patient finds a doctor they want to schedule an appointment with but they want the appointment to be online so they request an online appointment.

Function: Request an online appointment

Description: A patient can request an online appointment but he will have to write a brief description of his symptoms and wait for confirmation from the doctor or the administrator .

Source: input by patient

Outputs: pending confirmation note, a request for an appointment

Destination: Displayed on the patient's device, sent to the doctor or administrator as a notification

Action: send a notification to the administrator or the doctor of the request and asking for their response

Requirements:input by patient

Pre-condition:The user is logged in as a patient

Post-condition: the patient is notified with the request's response and if confirmed store it in the database.

Side effects: No response for the request and the time slot is booked and confirmed by another patient, so the request needs to be dropped and the patient needs to be notified .

1.7

Scenario: A patient is in a situation where they need urgent medical care, so they use the app to promptly contact an emergency responder.

Function: Contact emergency services

Description: An emergency contact service is made available for the patients to request emergency assistance at any given time.

Inputs: distress signal

Source: patient input from a dedicated input point

Outputs: emergency alert, along with patient location and information

Destination: Any emergency responders within a 10 km radius with active devices

Action: The distress signal is processed and the patient profile is retrieved from the database and forwarded with an alert to any nearby emergency responders with active logins

Requirements: The patient must be in an area with service reachability.

Pre-condition: Patient must have signed up for the system and consented to sharing their data in the case of emergencies.

Post-condition: Emergency responders within the specified distance receive the alert.

Side effects: None

1.8

Scenario: A patient scheduled an appointment or a follow-up with a doctor and they wish to be notified when the appointment date approaches, so they set custom reminders through DigiMed.

Function: Add custom reminders for upcoming appointments/ follow-ups.

Description: A user may choose to set up to 3 reminders for any of their future appointments, at any given dates and times before their appointment time. They may also be given default suggestions.

Inputs: Appointment ID, reminder 1 date and time, (optional) reminder 2 date and time, (optional) reminder 3 date and time.

Source: Appointment ID automatically received through patient selection, the rest of the data is received through patient's direct input.

Outputs: none

Destination: appointment database.

Action: The appointment is flagged in the system and a notification is sent to the user at the specified date and time.

Requirements: The date/time input must be prior to the actual date/time of the appointment.

Pre-condition: The appointment for which the reminder is being created must exist

Post-condition: Appointment entry modified to flag the reminders and add their dates and times.

Side effects: none

1.9 [Mohamed Khalil]

Scenario: Patients often ask for clinic details after booking appointments. The DigiMed platform should automatically include the clinic's location in the appointment confirmation.

Function: Include the clinic's location in appointment confirmations.

Description: The system automatically adds the clinic's address to the appointment confirmation sent to patients.

Inputs:

- Clinic location
- Appointment details

Source: Scheduling system.

Outputs:

- Appointment confirmation with clinic location

Destination: Notification system.

Action: Attach clinic location to the appointment confirmation message.

Requirements:

- Ensure location is correct and included in all confirmations

- Automate the inclusion of the clinic's address

Pre-condition: Appointment must be booked.

Post-condition: Patients receive confirmation with clinic location details.

Side effects: NA

1.10 [Mohamed Khalil]

Scenario: To improve service quality and patient satisfaction, patients should be able to rate and review doctors after their appointments. This feedback will help other patients choose doctors and provide valuable insights to the clinic for improvement.

Function: Allow patients to rate and review doctors.

Description: The system enables patients to provide a rating and written review of their experience with a doctor.

Inputs:

- Patient rating (1-5 stars)
- Text review

Source: Patients' post-appointment feedback.

Outputs:

- Rating and review stored in the system
- Displayed average rating and reviews for each doctor

Destination: Doctor's profile in the system, visible to other patients and clinic administrators.

Action: Collect, store, and display ratings and reviews.

Requirements:

- Ensure reviews are linked to completed appointments
- Ability for doctors to respond to reviews
- Anonymous or verified feedback options

Pre-condition: The patient must have completed an appointment with the doctor.

Post-condition: NA

Side effects: Possible biased, unfair reviews or inappropriate feedback.

1.11

Scenario: A patient that is registered in DigiMed could get an appointment with a doctor that does not use the DigiMed system , therefore they won't be able to upload the data for the patient for them and the patient has to upload it for themselves. Also when a patient registers for the first time in DigiMed they will need to upload their medical history.

Function: a patient can add information to their medical record.

Description: The patient's medical history service will receive a request from the patient UI and then after authenticating that this person modifying is the patient themselves , a request is sent to the database management system.

Inputs: The patient's ID.

Source: A request sent from the Patient UI.

Outputs: Either a message stating that the medical records have been updated or one that says the update was not allowed.

Destination: The patient medical history table in the database.

Requirements: The person signing in has to be the patient themselves.

Pre-condition: Checking if the patient can add to their medical records.

Post-condition: The patient will have an updated medical record.

Side effects: The patient could upload unnecessary or incorrect information due to their lack of experience.

1.12

Scenario: If a patient uploads their tests and one of those tests show alarming results that should be checked by a doctor.

Function: viewing the suggestions of which doctor to visit.

Description: The profile management service receives a request from the patients UI and then after the user logging in is identified to be a specific patient the profile management service sends a request to the database management system

Inputs: Patients ID.

Source: The request sent from the Patient UI.

Outputs: Either a message denying access to the users profile is sent or the user profile is displayed.

Destination: The patient table in the database.

Requirements: A user should be authorized to access this profile.

Pre-condition: The patient should have tests or scans uploaded.

Post-condition: The patient could take the suggestion or ignore it.

Side effects: The suggestion might not always be accurate.

2.1

Scenario: A patient has had an appointment with a doctor, and was given a prescription. This prescription needs to be added to the patient's profile.

Function: Write a prescription for a patient

Description: Allows a doctor to add a prescription to a patient's profile.

Inputs: Doctor's name, date of prescription, medicine name, form of intake, frequency of consumption, duration of consumption, subscription heading.

Source: The doctor's name is taken from the doctor's profile, the date is automatically added based on the current date, and the rest is manually inputted by the doctor.

Outputs: A prescription entry

Destination: The prescription table in the database

Action:

Requirements: Input from the doctor.

Pre-condition: The doctor must be logged in and the patient must have had an appointment with the doctor creating the prescription.

Post-condition: The prescription will be available on the patient's profile

Side effects: None

2.2

Scenario: A doctor has signed up for this system and wants to add their specialization and other relevant information, including education, years of experience, to their profile to be viewable for those wanting to search for a doctor or schedule an appointment

Function: Add data to doctor profile

Description: The doctor adds their specialization, education and years of experience to their profile in the designated slots, along with any additional information they wish to add to the notes section.

Inputs: Specialization, education, years of experience, and any optional notes
Source: doctor's input.
Outputs: updated doctor profile
Destination: doctor table in database.
Requirements: Input from the doctor.
Pre-condition: The doctor must have signed up for this system and created a profile and be logged in.
Post-condition: The new data is displayed on the doctor's profile.
Side effects: Doctor becomes searchable by specialization.

2.3

Scenario: After an appointment or before an appointment with a patient, a doctor would like for them to run some tests so they look up the patient and send a request for the desired tests.

Function: A doctor can request a test for a patient.

Description: Through the **prescription management system** a doctor can request a test for a current patient which is then reflected in the patient's prescriptions.

Inputs: Doctor ID; Patient ID; the test requested.

Source: The client request source is from the **API Gateway**; the doctor requests the test through the Doctor UI.

Outputs: The requested test for the patient with reference to the doctor who issued it.

Destination: The patient's prescriptions are updated through the **Database management system**. The patient is notified of the test request through the **Notifications API**.

Action: If a patient is scheduled to have a visit to a doctor or has had a visit the doctor can request a test be added to the patient prescriptions.

Requirements and Pre-condition: The system must have at least one organization with a doctor profile, at least one patient who has a scheduled appointment or has had one.

Post-condition and Side effects: The test is added to the patient's prescriptions, but the patient can choose to reject the request and remove it directly using the **prescription management system**. A notification is issued to the patient indicating the test request through the **Notifications API**.

2.4

Scenario: A doctor needs to ask another doctor for elaborations about something they wrote in a patient's medical record

Function: Contact patients' previous doctors for elaborations

Description: The doctor can view contact info of their patient's previous doctors

Inputs: Patient's national ID

Source: input by doctor

Outputs: doctors' contact info from the patient's medical record

Destination:Displayed on the doctor's device.

Action:

Requirements:input by doctor

Pre-condition:The doctor can view their patients medical record including their previous doctors and which doctor added which part.

Post-condition: N/A

Side effects: N/A

2.5

Scenario: A doctor has an emergency and needs to access or edit appointments by themselves without contacting the administrator

Function: Cancel/ filter/ postpone an appointment

Description: The doctor can view all appointments or filter them based on date/time/location and for each appointment, they can choose to cancel or postpone it.

Inputs: doctor's national ID

Source: input by doctor

Outputs: list of appointments

Destination:Displayed on the doctor's device.

Action: The doctor can cancel or postpone the appointment

Requirements:input by doctor

Pre-condition:The doctor can view their appointments

Post-condition: Schedule updates are reflected, and notifications are sent out.

Side effects: N/A

2.6

Scenario: In an appointment with a doctor a patient is either given a prescription entailing the medicines the patient needs to take or receives a blood test or scan done in the appointment.

Function: Updating the medical history of patients and adding notes to their profiles.

Description: After checking that this doctor is allowed to update this patient's profile , will send a request to the Profile medical history management service to update that patient's medical history.

Inputs: The prescriptions written to the patient or the tests and scans the patient has done and Patients ID and doctors ID.

Source: The documents are sent through the doctors UI.

Outputs: a message stating whether the update was successful or not.

Destination: The patient's medical history table in the database.

Action: Allowing the content coming from the doctor UI to be accepted in the database.

Requirements: Input from the doctor UI and permission for the doctor to access these records.

Pre-condition: The patient has to actually be registered in the organization that the doctor works in and to be connected to that specific doctor.

Post-condition: For the records to be updated.

Side effects: None

2.7

Scenario: If a person is a patient to a specific doctor They can access their profile of their patient including the details about the patient and their medical history and history of previous visits.

Function: Assessing a patient's profile.

Description: After receiving an approval that the doctor can access this patient's profile and medical history, the profile medical records management service sends a request to the database management system.

Inputs: Patients ID and doctors ID.

Source: Request from the doctors UI.

Outputs: the patient's profile and medical history or a message disapproving the request.

Destination: The patient's medical history table in the database.

Requirements: Input from the doctor UI and permission for the doctor to access these records.

Pre-condition: The patient has to actually be registered in the organization that the doctor works in and to be connected to that specific doctor.

Post-condition: For the patient's profile to appear on the doctors screen.

Side effects: None

2.8 (ADAM)

Scenario: Dr. Darine uses the Digimed app to check her daily schedule. She clicks "Today's Appointments" to view a list of her booked patients, complete with time slots, patient names, and brief remarks. This allows her to prepare for each appointment.

Function: View scheduled appointments.

Description: Doctors can access their schedules to see upcoming appointments.

Inputs: Doctor's ID, date/time range.

Source: Internal database.

Outputs: List of appointments.

Destination: User interface for doctors.

Action: The system retrieves and displays the list of scheduled appointments for the selected time frame.

Requirements: Secure login for doctors.

Pre-condition: Valid doctor account.

Post-condition: None.

Side effects: None.

2.13 (ADAM)

Scenario: Dr. Adams consults with a patient in need of a specialist. During the visit, Dr. Adams picks the "Refer Patient" option in the app, enters Dr. Williams' (a specialist) information, and offers his reasons for the referral. The system sends notification to Dr. Williams with the referral information, and the patient's profile is changed to reflect it.

Function: Refer patients.

Description: Enables doctors to refer patients to other specialists or healthcare providers.

Inputs: Patient information, referred doctor's details, reason for referral.

Source: Doctor's input during consultation.

Outputs: Referral details shared with the referred doctor.

Destination: Referred doctor's account.

Action: Referral details are saved and shared with the referred doctor via the system.

Requirements: Secure authentication and appropriate patient consent.

Pre-condition: Valid doctor account and patient consent.

Post-condition: Referral is logged in the system.

Side effects: None.

2.14 [Mohamed Khalil]

Scenario: Dr. Ahmed is a registered doctor in DigiMed. His clinic has long patient wait-times. DigiMed will optimize Ahmed's schedule to improve his clinic efficiency and the satisfaction of his patients.

Function: Optimize the doctor's schedule.

Description: The system will optimize schedules by adjusting appointments based on wait times and average appointment lengths.

Inputs:

- Patient wait times
- Average appointment length

Source: Scheduling Management System

Outputs:

- Optimized doctor's schedule
- Alerts for adjusted appointments

Destination:

- Scheduling system
- Patient notification system

Action: Rearrange and adjust the appointment slots to increase efficiency.

Requirements: N/A

Pre-condition: Appointment and wait time data must be available.

Post-condition: NA.

Side effects: Few appointments may need to be rescheduled.

3.1

Scenario: An existing administrator would like to provide another administrator in their office with only read access privileges or read and write access privileges to a certain doctor's schedule. Also a doctor would like to provide their administrator with only read access privileges or read and write access privileges to their own schedule(s).

Function: Provides the administrator with read and/or write access to certain appointments and schedules.

Description: Using the **Profile Management Service** read and/or write access is provided to schedules and appointments associated with the profile of the administrator linked to the doctor profile or organization that owns the schedules.

Inputs: Organization ID; Doctor ID; Administrator ID; Schedule and appointment ID; readAccess (boolean); writeAccess (boolean); IssuerID (the ID of the person or system assigning the access).

Source:

Client request source is from the "API Gateway"; the data is entered from the administrator UI or doctor UI.

Data retrieval source is from the database management system.

Outputs: Organization ID; Doctor ID; Administrator ID; Schedule and appointment ID with the updated access level; Type of access (read or write); IssuerID (who granted the access)

Destination: The database is updated through the **Database management system**. The change is logged using the Analysis and Reporting system.

Action: By default, an administrator has no permissions for a doctor's schedule in a particular organization. If a request is made by an authorized administrator or the doctor who owns the schedule for read access only, read access is granted. If the request is for write access, both read and write permissions are given. Permissions are updated accordingly in the system, any relevant information is logged, and notification(s) are sent out to the issuer, doctor, and receiver.

Requirements and Pre-condition:

The system must have at least one organization with a doctor profile, at least one schedule, and an administrator profile.

The administrator making the request must be authorized to request or assign access.

Post-condition and Side effects:

The administrator's access level to the schedule and appointment is updated in the database.

The reporting system reflects the access update.

A notification is sent to the issuer, doctor who owns the schedule, receiver of access.

3.2 (ADAM)

Scenario: An administrative staff member uses the Digimed app to update Dr. Wael's schedule. After validating a patient's appointment rescheduling request, the staff member updates the appointment time and clicks "Save." The system adjusts the timetable and notifies both the patient and the doctor.

Function: Manage and update a doctor's schedule.

Description: Allows authorized users to update and share a doctor's schedule, including setting, canceling, or rescheduling appointments.

Inputs: Appointment details, patient information, schedule updates.

Source: User inputs (administrative staff or doctors).

Outputs: Updated schedule, notifications to relevant parties.

Destination: Doctor's schedule module.

Action: New or modified appointments are saved in the system. Patients and relevant staff members are notified of any changes.

Requirements: Access control to ensure only authorized users can edit the schedule.

Pre-condition: Valid login and authentication for users updating schedules.

Post-condition: Schedule updates are reflected, and notifications are sent out.

Side effects: None

3.3

Scenario: A patient scheduled an appointment with a doctor and the doctors' clinic/ assistant wants to contact that patient for confirmations, inquires or providing them with information

Function: Contact patients scheduled with the doctor

Description: the administrative can see a list of patient registered with the doctor and their contact information

Inputs: Patient's national ID

Source: input by administrator

Outputs: Patient's contact info

Destination: Displayed on the administrator's device.

Action:

Requirements: input by administrator

Pre-condition: the patient is scheduled with a doctor that the administrator manages

Post-condition: N/A

Side effects: N/A

3.4 [Mohamed Khalil]

Scenario: When a prescribed medicine is unavailable at the pharmacy, the pharmacist should suggest alternative medications that are in stock.

Function:

Suggest alternative medicines if the prescribed one is unavailable.

Description:

The pharmacist recommends substitute medications when the original prescription is not available.

Inputs:

- Prescribed medicine
- Stock availability

Source: Input by Pharmacist

Outputs:

- List of available alternative medicines

Destination: Pharmacist system, patient notification.

Action: Check inventory, compare available alternatives, and suggest suitable substitutes.

Requirements:

- Access to real-time inventory data
- Approved list of alternative medications for each prescription

Pre-condition: Original prescribed medicine must be out of stock.

Post-condition: Alternative medicine is suggested and approved by the pharmacist.

Side effects: Potential for patient allergies or ineffectiveness with alternative medicines, requiring additional pharmacist intervention.

4.1

Scenario: A patient goes to a pharmacy in case of an emergency and the pharmacist gives him a drug that was not prescribed by any doctor, in order to handle the emergency.

Function: Record the drug given to patient in Urgent Cases (without Dr approval)

Description: A Pharmacist can add medications to a patient's medical record given that they also add a description of the urgent situation .

Source: input by pharmacist

Outputs: Add to medical profile form

Destination: Displayed on the pharmacist device

Action: N/A

Requirements: log in as a pharmacist

Pre-condition: putting the patient's medical ID

Post-condition: The patient's medical ID is updated

Side effects: The update could then be deleted if the patient reported it wasn't true or it was misleading.

5.1

Scenario: When a patient is admitted to a hospital they are assigned a nurse or a number of nurses to take care of them, therefore they should be able to access the medical records of the patients and see the reason why they are in the hospital. A nurse will need to know when to measure the patient's vitals depending on their state and what they are admitted for. They also need to know which medicines to give the patient and which ones they should avoid giving to them.

Function: Accessing the records of patients they are responsible for.

Description: The patient medical history service will receive a request from the Nurse UI and it will either reject it if that nurse is not allowed to check that patient's records or accept it otherwise, sending a request to the database management.

Inputs: Patents ID, Nurse ID

Source: Request sent from the nurse UI to access patients records.

Outputs: Either the patient's history will be opened to the nurse or a message stating that the access to this information was denied.

Destination: The patient's medical history table in the database.

Requirements: The patient has to be one the nurse is taking care of.

Pre-condition: The patient has to be admitted to the hospital that specific nurse works in.

Post-condition: None.

Side effects: Nurses might get to view other sensitive information about the user that won't help the nurse to take care of the patient better.

6.2

Scenario: A patient is in need of urgent care, and the emergency responders handling the case need to be able to access the patient's data.

Function: Access Medical profile of Patient in Urgent Cases .

Description: Using the patient's National ID and the emergency responders' status, the emergency responder is given access to the patient's profile.

Inputs: Patient's national ID.

Source: Input by emergency responder.

Outputs: Patient's medical and prescription histories.

Destination: Displayed on the emergency responder's device.

Action: If the emergency responder has clearance to view the patient's profile, their data is fetched from the local emergency database (or the main database if not found).

Requirements: input from the emergency responder.

Pre-condition: Both the emergency responder and the patient must have accounts on the system and the responder must be logged in. Additionally, the emergency responder must be given emergency access in the system, and the patient must have approved that their data be shared in urgent situations.

Post-condition: None.

Side effects: None.

6.4 (ADAM)

Scenario: An emergency responder arrives at an accident site and uses the Digimed app to locate a patient. They include crucial notes on the patient's condition, which are automatically updated on the patient's record and labeled as "Emergency Notes." This guarantees that if the patient is later admitted to the hospital, the doctor who treated them will have instant access to this information.

Function: Update patient records in emergencies.

Description: Emergency responders can add critical notes to a patient's profile during urgent situations without a doctor's approval.

Inputs: Patient information, emergency notes.

Source: Emergency responder input.

Outputs: Updated patient records.

Destination: Patient's medical record.

Action: New notes are added to the patient's profile, tagged as emergency input.

Requirements: Emergency access privileges for responders.

Pre-condition: Valid emergency responder account.

Post-condition: Notes are added to the patient's record.

Side effects: None.

7.1

Scenario: N/A

Function: N/A

Description: Emergency responders have access to a local database, which is only accessible by them, containing the data of those who approved emergency access in their local area.

Inputs: none

Source: N/A

Outputs: N/A

Destination: N/A

Action: N/A

Requirements: N/A

Pre-condition: the emergency responder must be given emergency clearance.

Post-condition: N/A

Side effects: N/A

7.2

Scenario: A user wants to log in as patient and a doctor or as a patient and emergency responder

Function: Users should be able to "acquire" titles/ modes

Description: one user can have access to the system with more than one role

Inputs: National ID or medical ID and role

Source: input by user

Outputs: specific role's interface

Destination: Displayed on the user's device.

Action: a user log in with a specific role

Requirements:input by user

Pre-condition: N/A

Post-condition:N/A

Side effects: N/A

7.2

As any user, I want to easily switch between my different users so that I can access and do what I need easily and quickly.

7.3

For each service it shall have replicas which are maintained and deployed using AKS. This is for reliability.

7.4 (ADAM)

Scenario: Every night, the Digimed system performs a scheduled task that compresses patient records older than one year. These compressed documents are sent to an archive database. This method decreases the burden on the live database, allowing for faster access to current data during business hours.

Function: Data management and optimization.

Description: Older patient and appointment data are compressed to reduce storage requirements and enhance system efficiency. Compressed data is transferred to less active databases.

Inputs: Historical data.

Source: Internal data management protocols.

Outputs: Compressed, archived data.

Destination: Archived storage database.

Action: Data older than a specified time frame is compressed and stored in an archive.

Requirements: Efficient data compression algorithms.

Pre-condition: Defined time threshold for data archiving.

Post-condition: Older data is archived, freeing space on active databases.

Side effects: None.

7.6 (ADAM)

Scenario: At the end of each day, the Digimed system automatically updates the emergency database, synchronizing crucial information from the main database. This assures that emergency responders have access to the most recent patient data, even when the main system is offline.

Function: Database synchronization.

Description: Ensures the emergency database is updated daily to reflect the most current information.

Inputs: Updated patient data, emergency records.

Source: Internal data synchronization protocols.
Outputs: Updated emergency database.
Destination: Emergency database storage.
Action: A scheduled process updates the emergency database every day.
Requirements: Regular data synchronization schedule.
Pre-condition: Stable connection between main and emergency databases.
Post-condition: Emergency database contains the latest critical information.
Side effects: None.

7.7

Scenario: Doctors , administrators, and Nurses work in the same hospital or clinic.
Function: The DigiMed system keeps track of which doctors, administrators, and nurses work in the same organization.
Description: The DigiMed system will only store the doctors, administrators, and nurses working for a specific organization.
Inputs: None
Source: N/A
Outputs: N/A
Destination: N/A
Action: N/A
Requirements: N/A
Pre-condition: N/A
Post-condition: N/A
Side effects: None

7.8

Scenario: A patient could schedule an appointment with a doctor and need to pay for it
Function: The DigiMed system will allow patients to pay for their appointments.
Description: The scheduling service will communicate with the payment API that will be used by the DigiMed system.
Inputs: none
Source: N/A
Outputs: N/A
Destination: Schedules table in database.
Requirements: N/A
Pre-condition: N/A
Post-condition: N/A

Side effects: none

7.9

Scenario: A user is trying to log in to the app and is waiting to access their account

Function:(Authentication)-> google/ national id/ phone number/ medical id

Description: A user's account can be authenticated using google/ national id/ phone number/ medical id

Inputs: google/ national id/ phone number/ medical id

Source: input by user

Outputs: specific user's profile

Destination:Displayed on the user's device.

Action: a user log in with a specific role

Requirements:input by user

Pre-condition: being registered in the system

Post-condition:N/A

Side effects: An account could be suspended due to many failed login attempts

7.10

Scenario: A doctor wants to check their past schedules to revisit some of his patients' info

Function :Doctor schedules are kept for a year

Description: Doctors' schedules are stored in the database and expire after one year. Doctors can view their past schedules.

Inputs: Doctors' schedules

Source: input by doctor or administrator and the database

Outputs: past schedules

Destination: stored in the database and displayed on the doctor's/ administrators device.

Action: a doctor request to view their past schedules

Requirements:input by user

Pre-condition: being registered in the system as doctor or administrator

Post-condition:N/A

Side effects: N/A

7.11 [Mohamed Khalil]

Scenario: Patients in Digimed can book online and in person appointments. The patient will be able to have an online appointment and he will get the feedback of the doctor whether an in person visit is needed.

Function: Approve an online appointment system or recommend in-person based appointment.

Description: The system decides whether the appointment can be conducted online or requires an in-person visit.

Inputs:

- Appointment type
- Dr decision

Source: Scheduling System

Outputs:

- Suitable Appointment type

Destination: Scheduling System, notification system.

Action: Evaluate the request and approve online or request in-person visit.

Requirements:

- Assess case complexity
- Automate approval or rejection based on criteria

Pre-condition: Patients must submit an appointment request with relevant details.

Post-condition: NA

Side effects: Possible delays.

7.12 [Mohamed Khalil]**Scenario:**

Users such as doctors, nurses, and patients require different levels of access to sensitive data. To ensure security, the system should categorize users and verify their identity based on their role, granting access only to the appropriate information.

Function: Categorize user roles and perform access verification.

Description: The system classifies users by role and verifies their identity before granting appropriate access to data.

Inputs:

- User role
- Login credentials

Source: User database, authentication system.

Outputs:

- Verified access to data

- Role-based access permissions

Destination: User interface (doctor, nurse, patient portals).

Action: Authenticate user and grant or restrict access based on role.

Requirements:

- Role-based access control
- Secure authentication process (e.g., multi-factor authentication)

Pre-condition: Users must be registered with roles assigned.

Post-condition: Users gain appropriate access based on verification.

Side effects: Delay in access

7.13 [Mohamed Khalil]

Scenario: Sensitive patient data including medical history, prescriptions, and personal details need to be protected in DigiMed. The system will encrypt all patient data, both at rest and in transit.

Function: Encrypt all patient data.

Description: The system ensures that patient data is encrypted at all times to prevent unauthorized access.

Inputs:

- Original Patient data

Source: Patient database

Outputs:

- Encrypted patient data

Destination: Secure storage, data transmission channels.

Action: Encrypt data before storing or transmitting.

Requirements:

- Use strong encryption algorithms (e.g., AES-256)
- Encrypt both stored data and data in transit

Pre-condition: Patient data must be collected and available in the system.

Post-condition: NA

Side effects: Potential performance delays due to encryption and decryption processes, particularly with large datasets.

7.14 [Mohamed Khalil]

Scenario: Doctors, nurses, and pharmacists need to collaborate on patient care by adding notes to a patient's profile. This allows all healthcare professionals involved to stay updated on the patient's condition, treatment, and medication.

Function: Allow doctors, nurses, and pharmacists to add notes to a patient's profile.

Description: The system enables healthcare professionals to record and share notes on a patient's profile for better care coordination.

Inputs:

- Notes from doctors, nurses, and pharmacists
- Patient profile

Source: Healthcare professionals, patient treatment records.

Outputs:

- Updated patient profile with new notes

Destination: Patient profile in the healthcare system.

Action: Allow healthcare professionals to enter and save notes to the patient's profile.

Requirements:

- Role-based access for adding notes
- Timestamp and identity of the note author
- Ability to edit or view previous notes

Pre-condition: The patient profile must exist in the system, and the user must be authenticated.

Post-condition: NA

Side effects: Overwriting important information or miscommunication if notes are not properly formatted or reviewed.

7.15 [Mohamed Khalil]

Scenario: DigiMed will track and display the average waiting time for each doctor based on patient feedback received after appointments. This data will help the clinic improve their services.

Function: Calculate and display the average waiting time for each doctor.

Description: The system collects patient feedback on waiting times after appointments and calculates the average waiting time for each doctor.

Inputs:

- Patient-reported waiting times

Source: Feedback forms by patients.

Outputs:

- Average waiting time for each doctor

Destination: Doctor profiles

Action: Collect patient feedback and calculate average waiting times.

Requirements: NA

Pre-condition: Patients must provide feedback after appointments.

Post-condition: NA

Side effects: NA