10Cent

LOGiT

Version <1.13>

Revision History

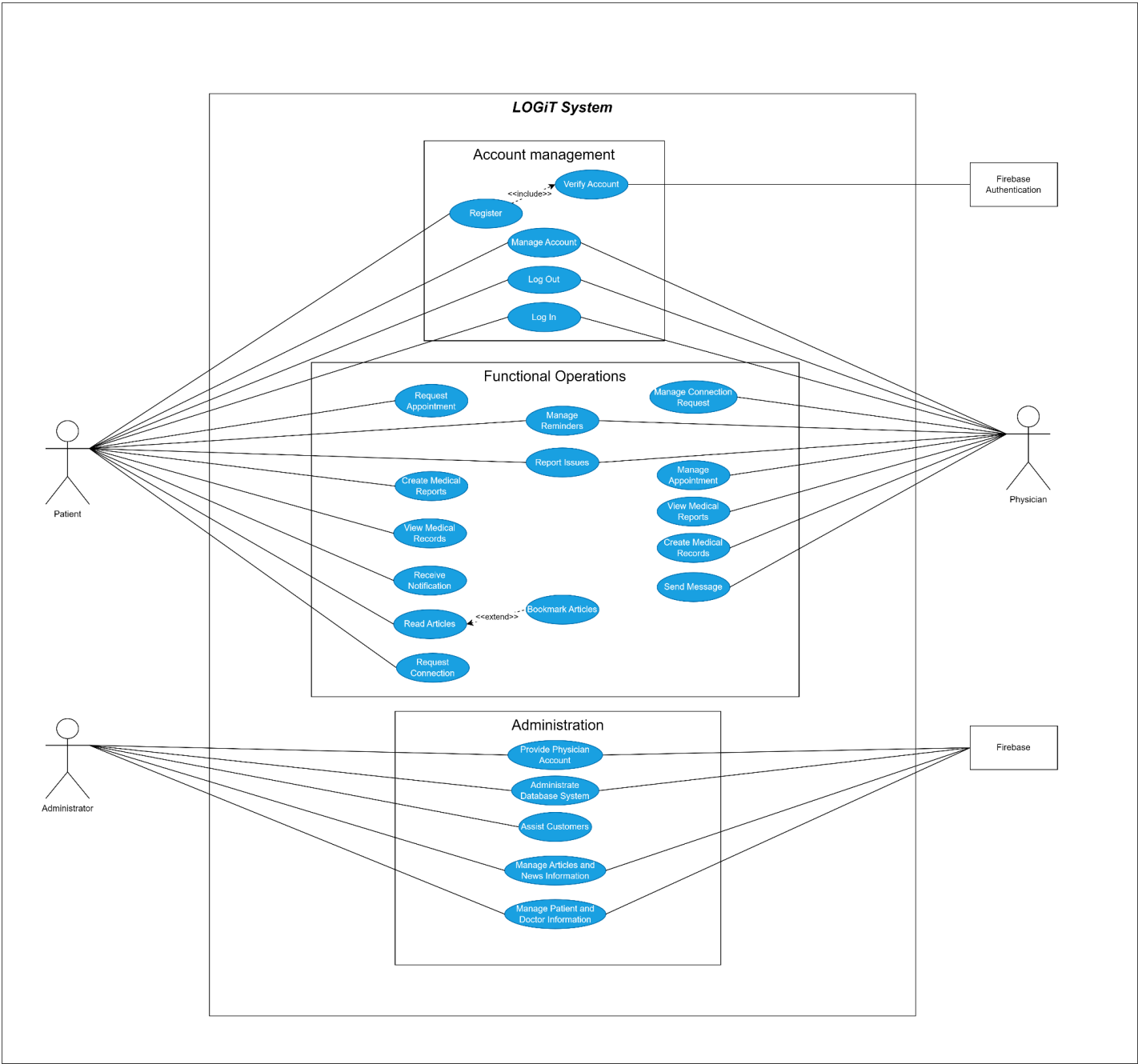| Date	Version	Description	Author
01/07/2024	1.0	Overall document structure	Lê Ngọc Thảo, Nhâm Đức Huy
02/07/2024	1.1	Insert Use Case Diagram	Võ Minh Khôi
02/07/2024	1.2	Complete table of use-cases	Võ Minh Khôi, Lê Ngọc Thảo
03/07/2024	1.3	Fill out use case for: Request Appointment (patient), Create Medical Reports, View Medical Records, Receive Notifications	Nguyễn Tấn Hoàng, Lê Nguyễn Minh Châu
04/07/2024	1.4	Fill out use case for: Read Articles, Request Connection, Bookmark Articles	Lê Ngọc Thảo, Nhâm Đức Huy
05/07/2024	1.5	Fill out use case for: Manage Connection Request, Verify Account, Manage Appointment	Lê Nguyễn Minh Châu, Nguyễn Tấn Hoàng
06/07/2024	1.6	Fill out use case for: View Medical Reports, Create medical records, Send message, Request Appointment (doctor)	Lê Ngọc Thảo, Nhâm Đức Huy
07/07/2024	1.7	Fill out use case for: Provide Physician Account, Administrate Database System, Assist Customers	Võ Minh Khôi
07/07/2024	1.8	Fill out use case for: Register, Log In, Log Out	Lê Nguyễn Minh Châu, Nguyễn Tấn Hoàng
08/07/2024	1.9	Fill out use case for: Manage Account, Manage Reminders, Report Issues	Lê Ngọc Thảo, Nhâm Đức Huy
10/07/2024	1.10	Overall content proofread	Võ Minh Khôi, Lê Ngọc Thảo, Nhâm Đức Huy
11/07/2024	1.11	Overall document format	Võ Minh Khôi
25/07/2024	1.12	Add use cases of Manage Patient, Doctor Information and Manage Articles, News Information, and format document.	Lê Ngọc Thảo
27/07/2024	1.13	Update Use-case Diagram	Võ Minh Khôi

Use case Specification Document

I. Use Case Diagram

II. Use Case Specification

No	Actors
1	Patient

2	Physician
3	Administrator

Use-case ID	Name	Actor
UC01	Request Appointment (patient)	1
UC02	Create Medical Reports	1
UC03	View Medical Records	1
UC04	Receive Notifications	1
UC05	Read Articles	1
UC06	Request Connection	1
UC07	Bookmark Articles	1
UC08	Manage Connection Request	2
UC09	Verify Account	2
UC10	Manage Appointment	2
UC11	View Medical Reports	2
UC12	Create Medical Records	2
UC13	Send Message	2
UC14	Request Appointment (doctor)	2
UC15	Provide Physician Account	3
UC16	Administrate Database System	3
UC17	Assist Customers	3
UC18	Manage Patient and Doctor Information	3
UC19	Manage Articles and News Information	3
UC20	Register	1,2
UC21	Log In	1,2
UC22	Log Out	1,2
UC23	Manage Account	1,2

UC24	Manage Reminders	1,2
UC25	Report Issues	1,2

1. Use-Case Specification: Request Appointment (patient)

Use case ID	UC01
Use case Name	Request Appointment (patient)
Brief description	Patients request an appointment with a connected healthcare provider through the LOGiT app.
Pre-conditions	1. The user must be authenticated and logged into the LOGiT application as a patient. 2. The doctor that the patient wants to request an appointment with must have an active account in the LOGiT system. 3. The doctor that the patient wants to request an appointment with must have an active connection with the patient.

Basic Flow	1. The patient accesses the LOGiT app.
	2. The patient taps on the "Health Diary" icon.
	3. The system displays a list of "Treatment" cards which contain information about the conditions of the patient and the corresponding doctor responsible for diagnosis/treating.
	4. The patient selects the "Treatment" card in which they want to request an appointment from to receive diagnosis/treatment.
	5. The system displays a "Treatment" interface.
	6. The patient selects "Request Appointment" to request an appointment from the doctor responsible for diagnosis/treating the patient.
	7. The system displays a "request appointment" form including fields like:
	    - Appointment date
	    - Appointment time
	8. The patient chooses the date and time for the appointment.
	9. The patient submits the appointment request.
	10. The system displays a confirmation message, indicating that the request has been sent.
	11. The system sends a notification to the doctor about the appointment request.

Alternative Flows	At step 4, if the patient has no connected doctors: 4a. The system displays a message informing the patient that they need to connect with a doctor first. At step 8, if the system fails to send the appointment request: 8a. The system displays an error message. 8a1. The patient can retry sending the request. Use case returns to step 8.
Post-conditions	1. The appointment request is sent to the doctor. 2. After a successful appointment request, it is added to the patient's reminder.
Special requirements	3. Communication must be secure and compliant with healthcare regulations.
Extension Points	None

2. Use-Case Specification: Create Medical Reports

Use case ID	UC02
Use case Name	Create Medical Reports
Brief description	Users create daily health reports about their condition, using a visual interface to select specific body parts and enter descriptions.
Pre-conditions	1. The app is already installed. 2. The user is logged into the app. 3. The user has at least one active connection with a doctor.
Basic Flow	1. The user accesses the app. 2. The user navigates to the "Medical Record" screen. 3. The user selects a "connection". 4. The user selects "Create Medical Report". 5. The system prompts for details about the health issue. 6. The user enters a description of the health issue. 7. The user saves the report. 8. The system confirms the report is saved and updates the user's health reports.
Alternative Flows	At step 6, if the user cannot identify the body part: 6a. The user select "Display Human Body" 6a1. The system displays a human body diagram. 6a2. The user selects a specific body part. 6a3. The system prompts for details about the selected area. Use case continues from step 7.
Post-conditions	1. The medical report is created and saved. 2. The system updates the user's health records.
Special requirements	1. The interface must be user-friendly and intuitive. 2. Health data must be securely stored and accessible only to authorized users.
Extension Points	None

3. Use-Case Specification: View Medical Records

Use case ID	UC03
Use case Name	View Medical Records
Brief description	Allow patients to access and view their medical records, including past treatments, reports, and health information, within the LOGiT app.
Pre-conditions	1. The user is authenticated and logged into the LOGiT application as a patient. 2. The patient has at least one active connection with a doctor. 3. The doctor has uploaded medical records for the patient.
Basic Flow	1. The user accesses the app. 2. The user navigates to the "Health Diary" screen. 3. The system displays the list of treatments. 4. The user selects a treatment from the list. 5. The user select "Medical Records" 6. The system displays the medical records associated with that treatment. This may include: - Treatment details (e.g., date, doctor, diagnosis) - Medical reports (e.g., daily symptom reports) - Notes from the doctor - Prescriptions - Lab results (if available) 7. The user can browse through the medical records. 8. The user can view detailed information for each record.
Alternative Flows	At step 4, if the user has no past treatments: 4a. The system displays a message informing the user. At step 5, if the user has no medical records for the selected treatment: 5a. The system displays a message informing the user. 5a1. The user can select another treatment from the list.
Post-conditions	The user is able to access and view their medical records.
Special requirements	1. Medical records must be displayed in a clear and organized manner. 2. The system must ensure the security and privacy of patient data. 3. Access to medical records must comply with healthcare regulations.
Extension Points	None

4. Use-Case Specification: Receive Notifications

Use case ID	UC04
Use case Name	Receive Notifications
Brief description	Allow users to receive messages and notifications from connected healthcare providers through the app.
Pre-conditions	1. The user has installed the LOGiT app. 2. The user is logged in to their account. 3. The user has enabled push notifications for the app (optional).
Basic Flow	1. The user receives a notification from the app. 2. The app displays a notification alerting the user of incoming updates/events from healthcare providers.. 3. The user taps the notification to view detailed information.
Alternative Flows	At step 2, if the user has not enabled push notifications, they will not receive a notification when they receive a new message. 2a. The user logs in the app and navigates to "Notifications" screen, the latest updates from healthcare providers will pop up. 2b. The user clicks on the notification, the detailed information will show up and the user is switched to the source of the notification where they can perform actions. At step 2, if the device is offline, they will not receive a notification when they have a new message. 2a. When the user connects to the internet, then they will receive the notifications. At step 3, if the app is not open, the app will be opened automatically. 3a. The detailed information will show up and the user is switched to the source of the notification where they can perform actions.
Post-conditions	1. The user is able to receive new information/updates from healthcare providers. 2. The notification is received in real-time.
Special requirements	The user grants permission to the app to receive real-time notifications.
Extension Points	None

5. Use-Case Specification: Read Articles

Use case ID	**UC05**
Use case Name	Read Articles
Brief description	Allow users to access and read educational health articles within the LOGiT app.
Pre-conditions	1. The user has launched the LOGiT app. 2. The user is logged in to their account.

Basic Flow	1. The user navigates to the "Health Blog" screen.
	2. The app displays a list of available health articles. Articles may be categorized by:
	   - Topic
	   - Author
	   - Publication date
	   - Relevance to the user's health profile/conditions.
	3. The user can browse the list to find articles.
	4. The user selects an article of interest.
	4. The app opens the original web page of the article using the default browser of the device.
	6. The user can:
	   - Mark articles for later reading or favorite (optional)
	   - Rate or provide feedback on the article (optional)

Alternative Flows	At step 2, if there are no available articles 2a. The app displays a message informing the user and may suggest alternative actions (e.g., browse by category, search for specific topics).
Post-conditions	1. The user is able to access and read educational health information. 2. The user gains knowledge about various medical topics, reports, and research findings. 3. If the user browse to certain articles, the app prompts them to do so securely
Special requirements	1. The article must be selectively chosen from reliable and verified sources. 2. The user should be able to filter and search for medical articles by keywords, self-interests, and other criteria. 3. The content of the web page must precisely be the selected article when the default internal browser is completely loaded.
Extension Points	None

6. Use-Case Specification: Request Connection

Use case ID	UC06
Use case Name	Request Connection
Brief description	The patient scans a QR code or manually enters a unique code to request a connection with a doctor.
Pre-conditions	1. The app is already installed. 2. The patient is logged into the app. 3. The patient has the doctor's QR or unique code.

Basic Flow	1. The patient accesses the app.
	2. The patient navigates to the "Health Diary" screen.
	3. The patient selects "Connect" (represented by a QR symbol).
	4. The system opens the "Connecting" interface.
	5. The patient can:
	   - Scan the QR code.
	   - Manually enter the doctor's unique code.
	6. The system sends a connection request to the doctor.
	7. The system notifies the patient that the request has been sent.

Alternative Flows	At step 5, if the code is invalid: 5a. The system displays an error message. Use case returns to step 3. At step 6, if the system fails to send the request: 6a. The system displays an error message. 6a1. The patient retries sending the request. Use case returns to step 5.
Post-conditions	1. The connection request is sent to the doctor. 2. The system records the connection request status.
Special requirements	1. QR codes must be valid and unique to each doctor, so as the unique code. 2. Communication must be secure and compliant with healthcare regulations.
Extension Points	None

7. Use-Case Specification: Bookmark Articles

Use case ID	UC07
Use case Name	Bookmark Articles
Brief description	Users can bookmark health articles to read later from the health blogs screen.
Pre-conditions	1. The app is already installed. 2. The user is logged into the app.
Basic Flow	1. The user accesses the "Health Blog" screen. 2. The user chooses an article from the list. 3. The user selects the "Bookmark" button. 4. The system saves the article to the user's bookmarked list. 5. The system confirms the article is bookmarked.
Alternative Flows	At step 5, if the system fails to bookmark the article: 5a. The system displays an error message. 5a1. The user retries bookmarking the article. Use case returns to step 4.
Post-conditions	1. The article is saved to the user's bookmarked list. 2. The system records the bookmarking activity.
Special requirements	1. Users must be able to access their bookmarked articles from any device.
Extension Points	None

8. Use-Case Specification: Manage Connection Request

Use case ID	UC08
Use case Name	Manage Connection Request
Brief description	When a patient requests a connection, the doctor can accept or refuse the connection request.
Pre-conditions	1. The app is installed. 2. The patient has sent a connection request. 3. The doctor is logged into the app. 4. The patient sent a connection request.
Basic Flow	1. The system sends a connection request to the doctor. 2. The doctor receives the connection request notification. 3. The doctor reviews the requester's basic information. 4. The doctor accepts the connection. 5. The system establishes the connection between the doctor and the patient. 6. The system prompts the doctor to enter a patient's conditions and other information to create a treatment profile for the newly established connection.
Alternative Flows	At step 4, if the doctor refuses the connection: 4a. The doctor selects "Refuse". 4a1. The system notifies the patient of the refusal. Use case ends. At step 6, if the doctor cancel the input process: 6a. This action is considered to be a refusal to the request, the system notifies the patient of the refusal. Use case ends.
Post-conditions	1. The doctor and patient are successfully connected. 2. The system records the connection status. 3. The system creates a new treatment profile for both patient and doctor.
Special requirements	Connection requests should expire if not responded to within 24 hours.

Extension Points	None

9. Use-Case Specification: Verify account

Use case ID	UC09
Use case Name	Verify Account
Brief description	Users verify their email address after registering for an account on the LOGiT app.
Pre-conditions	1. The user has launched the LOGiT app. 2. The user has created an account. 3. The user has not verified the account.
Basic Flow	1. The user accesses the app. 2. The user navigates to the "Profile" section. 3. The user selects the "Verify Account" option. 4. The system displays a message confirming the verification link has been sent.
Alternative Flows	At step 4, if errors occur in the verification link creating process: 4a. The system displays an error message and requests the user to retry. The use case continues from step 3. At step 3, if the user completed the verification process: 3a. The "Verify Account" option disappears and the system displays a message to inform the user of the verified status. Use case end.
Post-conditions	1. The user's account is verified. 2. The user is able to fully utilize the application.. 3. The system records the verification activity.
Special requirements	1. The system should provide clear feedback to the user about the status of their verification. 2. The verification link should be valid for a limited time. 3. The system should comply with security and privacy regulations.
Extension Points	None

10. Use-Case Specification: Manage Appointment

Use case ID	**UC10**
Use case Name	Manage Appointment
Brief description	Allow healthcare providers to manage patient appointment requests through the LOGiT app.
Pre-conditions	1. The healthcare provider has launched the LOGiT app. 2. The healthcare provider is logged in to their account. 3. The healthcare provider has upcoming appointment requests from patients.

Basic Flow	1. The healthcare provider opens the "Notifications" screen of the app.
	2. The app displays a list of upcoming notifications, including appointment requests from patients. Each request may include details like:
	   - Patient name
	   - Appointment date and time requested
	   - Reason for appointment (if provided by patient)
	   - Appointment type (e.g., initial consultation, follow-up)
	3. The healthcare provider selects an appointment request to view more details.
	4. The healthcare provider can choose one of the following
	   - Accept Appointment
	   - Deny Appointment

Alternative Flows	At step 2, if the healthcare provider has no upcoming appointment requests, t
	2a. The app displays a message informing them.
	
	At step 4, if the healthcare provider accepts the appointment:
	4a. The app automatically sends a notification to the patient confirming the appointment details (date, time, location) and creates a reminder for both the patient and doctor.
	If the healthcare provider deny the appointment:
	4b. The healthcare provider can:
	   - Navigate to the "Messaging" screen to inform the patient of the refusal and suggest a reschedule.
	   - Navigate back to the "Notification" screen and review other pending appointment requests.

Post-conditions	1. The user is able to receive updates/responses from healthcare providers. 2. The notification is received in real-time.
Special requirements	1. The health care provider must agree to consult and give the patient an available time frame. 2. The health care provider and patient must be connected during a previous treatment or follow-up session.
Extension Points	None

11. Use-Case Specification: View Medical Reports

Use case ID	**UC11**
Use case Name	View Medical Reports
Brief description	Allow health care providers to view medical reports uploaded by the patient through the app.
Pre-conditions	1. The user has launched the LOGiT app. 2. The user is logged in the app. 3. The user has medical reports uploaded by their patients.
Basic Flow	1. The user navigates to the "Medical Records" section of the app. 2. The app displays a list of current/past medical treatment. 3. The user selects a treatment from the list. 4. The user chooses "View symptom report". 5. The user then can view a list of medical reports..
Alternative Flows	At step 2, if the patient has no medical reports uploaded: 2a. The app displays a message informing the user. 2a1. The user can navigate back to the treatment list to choose another treatment. At step 2, if the current treatment has finished: 2a.The medical record will be moved to the "Past treatment" tab. 2a1.The user switches to the past treatment tab. 2a2.The user can choose to view a medical record in the past.
Post-conditions	1. The user is able to view medical reports uploaded by their patients. 2. The medical report history is updated within 5 minutes..
Special requirements	1. The user must have attended a treatment or follow-up through the app. 2. The patient agreed to upload their self medical reports
Extension Points	None

12. Use-Case Specification: Create medical records

Use case ID	UC12
Use case Name	Create Medical Records
Brief description	After connecting with a patient, the doctor updates the patient's information and medical records.
Pre-conditions	1. The app is installed. 2. The doctor and patient are connected. 3. The doctor is logged into the app.
Basic Flow	1. The doctor accesses the app. 2. The doctor navigates to the "Health Diary" screen. 3. The system displays the list of treatments. 4. The doctor selects a treatment. 5. The doctor selects "View Medical Records" 6. The system displays the patient's information and medical records. 7. The doctor creates a new medical record for the patient. 8. The doctor saves the new medical record.. 9. The system confirms the new update and records the changes.
Alternative Flows	At step 6, if the doctor fails to create the records: 6a. The system displays an error message. 6a1. The doctor retries updating the records. Use-case returns to step 6. At step 7, if the system fails to save the updates: 7a. The system displays an error message. 7a1. The doctor retries saving the updates. Use-case returns to step 7.
Post-conditions	1. The patient's medical records are updated. 2. The system records the update activity.
Special requirements	1. Medical records must comply with healthcare data standards and regulations.
Extension Points	None

13. Use-Case Specification: Send message

Use case ID	UC13
Use case Name	Send Message
Brief description	Allows doctors to send messages to patients for additional instructions or sudden schedule changes, minimizing patient-initiated spam.
Pre-conditions	1. The app is installed. 2. The doctor is logged into the app. 3. The patient is connected to the doctor.
Basic Flow	1. The doctor accesses the app. 2. The doctor selects the "Messaging" section. 3. The system displays the list of connected patients. 4. The doctor selects a patient. 5. The system opens a message interface. 6. The doctor types a message. 7. The doctor sends the message. 8. The system confirms the message is sent. 9. The system notifies the patient of the new message.
Alternative Flows	At step 8, if the system fails to send the message: 8a. The system displays an error message. 8a1. The doctor retries sending the message. Use-case returns to step 7.
Post-conditions	1. The message is successfully sent to the patient. 2. The system records the message in the patient's communication history.
Special requirements	1. Messages must be delivered promptly and reliably. 2. Communication must be secure and compliant with healthcare regulations.
Extension Points	None

14. Use-Case Specification: Request Appointment (Physician)

Use case ID	UC14
Use case Name	Request Appointment (Physician)
Brief description	Physicians request an appointment with a connected patient through the LOGiT app.
Pre-conditions	1. The user must be authenticated and logged into the LOGiT application as a physician. 2. The patient that the doctor wants to request an appointment must have an active account in the LOGiT system. 3. The patient that the doctor wants to request an appointment with must have an active connection with the doctor.
Basic Flow	1. The physician accesses the LOGiT app. 2. The system displays a list of "Treatment" cards, each represents a patient or a condition to monitor. 3. The physician selects a "Treatment". 4. The system displays a "Treatment" interface. 5. The physician selects "Request Connection". 6. The system displays a pop-up for the physician to confirm. 7. The system sends an appointment request notification to the selected patient/treatment.
Alternative Flows	At step 2, if the doctor has not created any connection: 2a. The system displays a message to inform the physician. Use-case end. At step 7, the doctor can provide additional information like the suitable time frame or purpose of the sudden appointment for the patient: 7a1. The doctor selects "Message" in the "Treatment" interface. 7a2. The doctor input additional information in the chat box. 7a3. The doctor selects "Send". 7a4. The message is sent and the patient will receive a notification about the message. Use-case end.
Post-conditions	- The appointment request is sent to the patient. - The doctor will then receive the patient's appointment request together with the time and date of the appointment.
Special requirements	- Communication must be secure and compliant with healthcare regulations.
Extension Points	None

15. Use-Case Specification: Provide Physician Account

Use case ID	UC15
Use case Name	Provide Physician Account
Brief description	The administrator provides Physician's accounts with their corresponding role.
Pre-conditions	1. The administrator has installed the LOGiT app. 2. The administrator has completed the verification process and is logged in to their account.
Basic Flow	1. The administrator accesses the app. 2. The administrator navigates to the "Account Management" section. 3. The system displays the user lists. 4. The administrator selects the Physician's account. 5. The account information is displayed for the administrator to verify. 6. The administrator selects the "Role" option. 7. The administrator selects the "Physician" option. 8. The administrator confirms the action. 9. The system sends a notification to the physician.
Alternative Flows	At step 4, if the Physician's account is not found: 4a1. The Physician has not created an account in the LOGiT application. Request the Physician to create an account and update the account's information to match with records. Use-case continues from step 5. At step 7, if the "Physician" option is not available: 7a1. The Physician's account is already configured with the "Physician" role, no further action is needed. Use-case ends.
Post-conditions	1. The physician's account is assigned the role "Physician". 2. The system sends a notification to the physician with the details. 3. The system records the activity.
Special requirements	1. The system should comply with security and privacy regulations. 2. The administrator should have the ability to edit or delete physician accounts as needed.
Extension Points	None

16. Use-Case Specification: Administrate Database System

Use case ID	**UC16**
Use case Name	Administrate Database System
Brief description	The hospital's technical department uses the app to monitor the status of patient and doctor accounts and receive notifications of errors reported by users.
Pre-conditions	1. The app is already installed. 2. The administrator has completed the verification process and is logged in to their account.
Basic Flow	1. The admin accesses the app. 2. The admin logs in with special credentials. 3. The system verifies the credentials. 4. The system displays the admin dashboard with account statuses and error notifications.
Alternative Flows	At step 2, if the credentials are invalid: 2a. The system displays an error message. 2a1. The admin re-enters credentials. Use case returns to step 3. At step 2, if the admin forgets the password: 2a. The admin selects "Forgot Password". 2a1. The system sends a password reset link. Use case resumes from step 3 after resetting the password.
Post-conditions	1. The admin successfully logs into the system. 2. The system records the admin login activity.
Special requirements	1. Admin password must be strong (min 8 characters, including upper/lower case and special characters). 2. Login attempts must be limited to prevent brute-force attacks.
Extension Points	None

17. Use-Case Specification: Assist Customers

Use case ID	**UC17**
Use case Name	Assist Customers
Brief description	Provide users with various ways to get help and answers to their questions about the app.
Pre-conditions	1. The user has launched the LOGiT app. 2. The user is logged in to their account.

Basic Flow	1. The user taps the "Help" button or icon.
	2. The app displays a menu with various help options, such as:
	   - FAQs (Frequently Asked Questions)
	   - User manuals or guides
	   - Live chat support
	   - Contact information (phone number or email)
	3. The user selects an option.
	4. The user chooses how to interact and receive help.
	5. The user receives support from the administrator.

Alternative Flows	At step 2, if the user selects FAQs: 2a. The app displays a list of commonly asked questions and answers. If the user selects User manuals or guides: 2b. The app provides access to documents or links that explain how to use the app's features. If the user selects Live chat support: 2c. The app connects the user with a customer service representative through a chat window. If the user selects Contact information: 2d. The app displays the phone number or email address for customer support. At step 3, if the user selects a help option that is not available: 3a. The app displays a message informing the user that the option is currently unavailable. 3b. Use case returns to step 2 to choose another available help option.
Post-conditions	1. The user is able to get help or answers to their questions about the app.
Special requirements	1. The administrators provide timely support.
Extension Points	None

18. Use-Case Specification: Manage Patient and Doctor Information

Use case ID	UC18
Use case Name	Manage Patient and Doctor Information
Brief description	Allow the Administrator to create, update, view, and delete patient and doctor information in the hospital management system, with specific permissions for changes based on hospital authorization.
Pre-conditions	1. The Administrator must be authenticated and have the necessary permissions to manage patient and doctor information. 2. The Administrator must possess the required hospital authorization documents for changes to patient or doctor information.
Basic Flow	1. The system displays the patient and doctor information management interface. 2. The Administrator selects the action to perform: Create, Update, View, or Delete patient or doctor records. 3. The system displays an interface for the selected action. 4. The Administrator enters or modifies patient or doctor information and submits it to the system. 5. The system checks the information for completeness and validity. 6. The system processes the request and updates the patient or doctor database accordingly. 7. The system provides feedback after confirming the action and displays the result to the Administrator.
Alternative Flows	At step 2, if the Administrator tries to delete a patient or doctor record that is referenced elsewhere in the database: 2. The system displays an error message indicating the record cannot be deleted. 2a1. The Administrator can choose to archive the record instead of deleting it. At step 4, if the Administrator enters invalid or incomplete information: 4a. The system displays an error message indicating the specific issue. 4a1. The Administrator corrects the information and resubmits it If the Administrator does not have the required hospital authorization documents for changes: 4b. The system displays an error message requesting the necessary authorization. 4b1. The Administrator obtains the required documents and resubmits the request. At step 6, If the system encounters a network or database error: 6a. The system displays an error message indicating the problem. 6a1. The Administrator may retry the operation after the issue is resolved.
Post-conditions	1. Patient and doctor information is accurately created, updated, viewed, or deleted in the system. 2. The system logs the action performed by the Administrator.

	3. The Administrator is provided with feedback and confirmation of the action performed. 4. Changes to patient or doctor information are reflected in the database (Firebase).
Special requirements	1. The system should validate the data entered by the Administrator to ensure it meets predefined criteria. The system should provide a user-friendly interface for managing patient and doctor information. The system must maintain audit logs of all actions performed by the Administrator for security and compliance purposes. The Administrator must have hospital authorization documents when making changes to patient or doctor information as requested by the hospital. Changes to patient or doctor information must be linked to the database (Firebase) to ensure data consistency.
Extension Points	None

19. Use-Case Specification: Manage Articles and News Information

Use case ID	**UC19**
Use case Name	Manage Articles and News Information
Brief description	Allow the Administrator to collect and post articles and news from verified and scientifically valid sources to the news section of the app.
Pre-conditions	1. The Administrator must be authenticated and have the necessary permissions to manage articles and news information. 2. The Administrator must use verified and credible sources for collecting articles and news.
Basic Flow	1. The system displays the articles and news management interface. 2. The Administrator selects the action to perform: Post, Update, or Delete articles and news. 3. The system displays an interface for the selected action. 4. The Administrator collects information from verified and scientifically valid sources, enters or modifies the articles and news, and submits it to the system. 5. The system checks the information for completeness and validity. 6. The system processes the request and updates the articles and news database accordingly. 7. The system provides feedback after confirming the action and displays the result to the Administrator.
Alternative Flows	At step 4, if the Administrator enters invalid or incomplete information: 4a. The system displays an error message indicating the specific issue. 4a1. The Administrator corrects the information and resubmits it. If the system finds that an article or news entry violates content policies: 4b. The system displays an error message and details on the violation. 4b2. The Administrator edits the content to comply with policies and resubmits. At step 6, if the system encounters a network or database error: 6a. The system displays an error message indicating the problem. 6a1. The Administrator may retry the operation after the issue is resolved.
Post-conditions	1. Articles and news information is accurately collected, posted, updated, or deleted in the system. 2. The system logs the action performed by the Administrator. 3. The Administrator is provided with feedback and confirmation of the action performed. 4. Articles and news posted to the app are accurate and scientifically valid.
Special requirements	1. The system should validate the data entered by the Administrator to ensure it meets predefined criteria. 2. The system should provide a user-friendly interface for managing articles and news information. 3. The system must maintain audit logs of all actions performed by the Administrator for security and compliance purposes. 4. The Administrator must collect information from verified and scientifically valid sources to ensure accuracy before posting to the news section of the app.
Extension Points	None

20. Use-Case Specification: Register

Use case ID	UC18
Use case Name	Register
Brief description	New users can create an account on the LOGiT app to access its features.
Pre-conditions	1. The user has installed the LOGiT app. 2. The user is not logged in.

Basic Flow	1. The user accesses the app.
	2. The user selects the "Register" option.
	3. The system displays the registration form.
	4. The user enters required information:
	   - Name
	   - Email
	   - Password
	5. The user agrees to the terms and conditions.
	6. The user submits the registration form.
	7. The system verifies the information and checks for existing accounts with the same email address.
	8. The system sends a verification link to the user's email address.
	9. The system displays a message informing the user that the verification link has been sent.

Alternative Flows	At step 4, if the user enters invalid information: 4a. The system displays an error message. 4a1. The user corrects the information. Use case continues from step 4. At step 8, if the system fails to send the verification link: 8a. The system displays an error message. 8a1. The user retries submitting the registration form. Use case continues from step 8.
Post-conditions	1. The user's account is created. 2. The system sends a verification link to the user's email address. 3. The system records the registration activity.
Special requirements	1. The registration form should be user-friendly and intuitive. 2. The system should provide clear feedback to the user about the status of their registration. 3. Passwords must meet security requirements (e.g., minimum length, complexity). 4. The system should comply with privacy regulations.
Extension Points	None

21. Use-Case Specification: Log in

Use case ID	UC19
Use case Name	Log In
Brief description	Users can access their LOGiT account by entering their credentials.
Pre-conditions	1. The user has launched the LOGiT app. 2. The user has created an account.
Basic Flow	1. The user accesses the app. 2. The user selects the "Log In" option. 3. The system displays the login form. 4. The user enters their email address. 5. The user enters their password. 6. The user submits the login form. 7. The system verifies the credentials. 8. The system logs the user into their account. 9. The system displays the user's home screen.
Alternative Flows	At step 4, if the user enters an invalid email address: 4a. The system displays an error message. 4a1. The user corrects the email address. Use case continues from step 4. At step 5, if the user enters an incorrect password: 5a. The system displays an error message. 5a1. The user can retry entering the password. Use case returns to step 5. At step 7, if the system fails to verify the credentials: 7a. The system displays an error message. 7a1. The user can retry entering their credentials. Use case returns to step 4.
Post-conditions	1. The user is logged in to their account. 2. The system records the login activity.
Special requirements	1. The login form should be user-friendly and intuitive. 2. The system should provide clear feedback to the user about the status of their login. 3. Passwords must be securely stored and protected. 4. The system should comply with security and privacy regulations.
Extension Points	Verify account

22. Use-Case Specification: Log out

Use case ID	**UC20**
Use case Name	Log Out
Brief description	Users can securely exit their LOGiT account and end their session.
Pre-conditions	1. The user has launched the LOGiT app. 2. The user is logged in to their account.
Basic Flow	1. The user accesses the app. 2. The user navigates to the "Profile" section. 3. The user selects the "Log Out" option. 4. The system displays a confirmation message (optional). 5. The user confirms the log out action. 6. The system logs the user out of their account. 7. The system returns the user to the login screen.
Alternative Flows	At step 4, if the user cancels the log out action: 4a. The system returns the user to the previous screen.
Post-conditions	1. The user's session has ended. 2. The user is logged out of their account.
Special requirements	1. The log out process should be secure and have measures to prevent unauthorized access to the user's account. 2. The system should provide clear feedback to the user about the status of their log out.
Extension Points	None

23. Use-Case Specification: Manage account

Use case ID	**UC21**
Use case Name	Manage Account
Brief description	Allow users to perform various actions to manage their accounts' information.
Pre-conditions	1. The user has launched the LOGiT app. 2. The user is logged in to their account.

Basic Flow	1. The user accesses the app.
	2. The user navigates to the "Account" section.
	3. The system displays the user's account profile.
	4. The user can choose to update one of the following and make the desire changes:
	   - User's full name
	   - User's date of birth
	   - User's phone number
	   - User's email
	   - Current hospital user is receiving treatment
	5. The user saves the changes.
	6. The system confirms the changes.

Alternative Flows	At step 4, if the user attempts to change a field that requires confirmation: 4a. The system prompts the user for additional authentication (e.g., password confirmation). 4b. The user enters the confirmation information. 4c. The system verifies the confirmation. Use case continues from step 4. At step 6, if the system fails to save the changes: 6a. The system displays an error message. 6a1. The user retries saving the changes. Use case continues from step 6.
Post-conditions	1. The user's account information is updated successfully. 2. The system records the changes.
Special requirements	1. Changes to sensitive information (e.g., contact details) must be secured with appropriate validation. 2. Account management must comply with privacy regulations. 3. Certain information must follow a predefined format (e.g., phone number must only contain numbers, date must follow a predetermined format)
Extension Points	None

24. Use-Case Specification: Manage reminders

Use case ID	UC22
Use case Name	Manage Reminders
Brief description	Allows users to create, edit, and delete reminders for medication intake or other health-related reminders as instructed by their doctor.
Pre-conditions	1. The app is installed. 2. The user is logged into the app. 3. The user is on the reminder screen.
Basic Flow	1. The user accesses the app. 2. The user navigates to the "Reminders" section. 3. The system displays the reminder management screen. 4. The user creates a new reminder. 5. The user enters the details of the reminder (e.g., medication name, time, frequency). 6. The user saves the reminder. 7. The system confirms the reminder is set. 8. The system sends notifications according to the reminder schedule.
Alternative Flows	At step 6, if the user edits an existing reminder: 6a1. The user selects an existing reminder. 6a2. The user updates the reminder details. 6a3. The user saves the changes. 6a4. The system confirms the changes. At step 6, if the user deletes a reminder: 6b1. The user selects an existing reminder. 6b2. The user deletes the reminder. 6b3. The system confirms the deletion.
Post-conditions	1. The reminder is successfully created, edited, or deleted. 2. The system sends notifications and alerts as scheduled.
Special requirements	1. Notifications must be timely and reliable. 2. Reminders should support multiple types and frequencies.
Extension Points	None

25. Use-Case Specification: Report issues

Use case ID	**UC23**
Use case Name	Report Issues
Brief description	Users can report any problems or bugs they encounter within the LOGiT app to the administrator.
Pre-conditions	1. The user has launched the LOGiT app. 2. The user is logged in to their account. 3. The user has encountered an issue or bug.

Basic Flow	1. The user accesses the app.
	2. The user navigates to the "Help" section.
	3. The user selects the "Report Issue" option.
	4. The system displays the issue reporting form.
	5. The user provides details about the issue:
	   - A brief description of the problem.
	   - The specific screen or feature where the issue occurred.
	   - Any relevant error messages or screenshots.
	   - The steps they took leading up to the issue.
	6. The user submits the issue report.
	7. The system sends a confirmation message to the user.
	8. The system records the issue report.

Alternative Flows	At step 5, if the user is unable to provide a screenshot: 5a. The user can describe the issue in detail. Use-case continues from step 6. At step 6, if the system fails to submit the report: 6a. The system displays an error message. 6a1. The user retries submitting the report. Use-case returns to step 6.
Post-conditions	1. The issue report is submitted to the administrator. 2. The system records the issue report.
Special requirements	1. The issue reporting form should be user-friendly and intuitive. 2. The system should provide clear feedback to the user about the status of their report. 3. The system should allow users to attach screenshots or other relevant files. 4. The administrator should be notified of new issue reports promptly.
Extension Points	None