Documentation – Sprint 3 (Notifications Feature)

1. Feature Name

AI / Rule-Based Health Tips & Chatbot Assistant

2. Overview

This feature provides personalized health tips to patients based on their profile data such as age, gender, condition, and doctor specialization.

The tips are generated using a rule-based engine (not heavy AI) and displayed either as a list on the dashboard or through an interactive chatbot interface that simulates an AI health assistant.

Admins can manage a library of tips, and the chatbot intelligently filters and responds with relevant messages.

3. Objectives

- To give patients personalized health awareness using rule-based recommendations.
- To create a chat-like experience that feels interactive and AI-driven.
- To allow admins to easily add and update health advice without modifying code.

4. Scope (Sprint 3)

Included in Sprint 3:

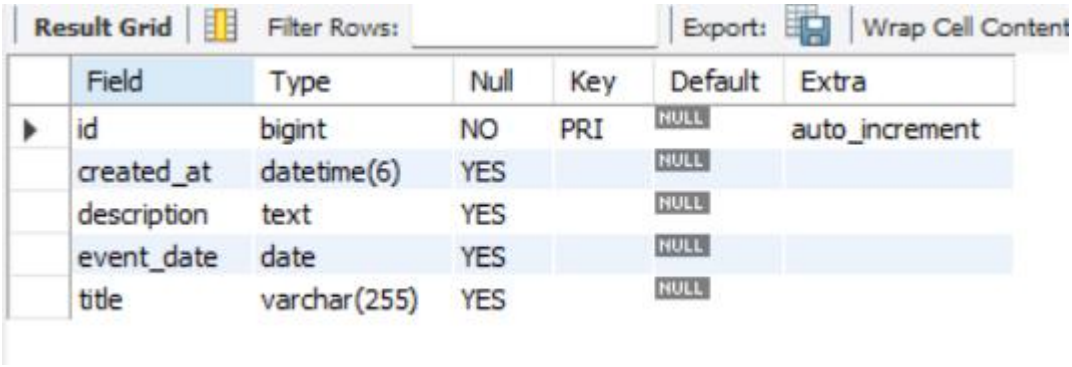
- Creation of health_tips table in the database.
- Implementation of a rule-based engine for filtering tips.
- Frontend dashboard display for patients.
- Chatbot interface for interactive advice.
- Admin panel for managing tips library (CRUD).

Not in scope:

- Integration with external AI or ML services (future enhancement).
- Predictive analytics or real-time monitoring.

5. Database Design

Table: health_tips



Result Grid | Filter Rows: | Export: | Wrap Cell Content

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_at	datetime(6)	YES		NULL	
description	text	YES		NULL	
event_date	date	YES		NULL	
title	varchar(255)	YES		NULL	

6. APIs (Spring Boot)

Endpoint	Method	Description
`/api/health-tips`	GET	Fetch all health tips (Admin view).
`/api/health-tips/user/{userId}`	GET	Get personalized health tips using rule engine.
`/api/health-tips/chat?userId=&query=`	GET	Get chatbot-style responses based on query keywords.
`/api/health-tips`	POST	Add new tip (Admin).
`/api/health-tips/{id}`	PUT	Update existing tip (Admin).
`/api/health-tips/{id}`	DELETE	Delete tip (Admin).

7. Rule Engine Logic (Backend)

The backend uses a simple condition-based filter to match health tips to each user:

```
@Query("SELECT t FROM HealthTip t WHERE
       (t.condition = :condition OR t.condition = 'General') AND
       (t.age_min <= :age AND t.age_max >= :age) AND
       (t.gender = :gender OR t.gender = 'All') AND
       (t.specialization = :specialization OR t.specialization IS NULL)")
List<HealthTip> findMatchingTips(...);
```

Additionally, for the chatbot:

```
IF query CONTAINS 'diet' → return tips related to nutrition
IF query CONTAINS 'exercise' → return tips about physical activity
IF query CONTAINS 'checkup' → return general wellness tips
ELSE → show default personalized advice
```

8. Frontend (React)

- Admin Panel
  - Page: Manage Health Tips
  - Functions: Add, Edit, Delete tips
  - Fields: Condition, Age Range, Gender, Specialization, Message

- Patient Dashboard
  - Section: My Health Tips
  - Functions: Fetch & display personalized tips
  - UI: Cards or carousel of advice based on backend data

- Chatbot Component (Health Assistant)
  - React component HealthChatbot.js
  - Displays conversation bubbles between user and bot
  - Fetches chatbot responses from /api/health-tips/chat

9. Expected Output

- Admin: Can create and manage a library of health tips.
- Backend: Matches user profile and query keywords with relevant tips.
- Patient Dashboard: Displays personalized health tips automatically.
- Chatbot: Responds conversationally with tips related to user profile or input keywords.

10. Example Scenario

Patient:

- Age = 52
- Gender = Female
- Condition = “Diabetes”
- Specialization = “General Medicine”

Dashboard Output:

- Maintain a balanced diet and monitor blood sugar regularly.
- Regular check-ups are important for adults over 50.

Chatbot Conversation:

```
Bot: Hi! I’m your Health Assistant 🤖
Bot: Based on your profile, here are some helpful tips.
Bot: - Maintain a balanced diet and monitor blood sugar regularly.
Bot: - Regular check-ups are important for adults over 50.
User: What about exercise?
Bot: Try 30 minutes of walking at least 5 days a week.
```

11. Benefits

- Engaging, interactive health education.
- Easy for admins to manage and expand.
- Lightweight “AI-like” experience without real AI cost.
- Enhances patient awareness and wellness proactively.