Below is a **copy-paste ready Technical Stack Document** for **HealthMate AI**, aligned with your **PRD + Design Document** and written in an enterprise-style format that you can submit in hackathon or portfolio.

HealthMate AI – Technical Stack Document

1. Overview

HealthMate AI is a **cross-platform AI-enabled preventive healthcare application** built using **Flutter** with **Supabase as the primary backend infrastructure** and **optional Firebase services** for messaging and analytics.

The technology stack is selected to ensure:

- Scalability
- Security
- Real-time performance
- AI/ML capability
- Rapid development
- Cost-effectiveness (Free tiers)

2. High Level Architecture

Client Layer (Mobile App)

- Flutter UI
- Local Storage
- Offline Caching
- Local AI inference (optional)

Service Layer

- Supabase APIs
- AI Microservices
- Notification Services

Data Layer

- PostgreSQL (Supabase)
- Object Storage
- Analytics Logs

External Integrations

- LLM APIs (optional)
- Wearables (future)
- Firebase Messaging

3. Frontend Technology Stack

Framework

- **Flutter 3.x**
- Dart 3.x

State Management

- **Riverpod** (Primary)
- Provider (Fallback)

UI & Design System

- Material 3
- Custom ThemeData
- Figma Tokens → Flutter Theme

Core Packages

Purpose	Package
Navigation	go_router
Charts	fl_chart
Forms	flutter_form_builder
Icons	flutter_svg
Localization	intl
Notifications	flutter_local_notifications
Animations	lottie
HTTP	dio
Dependency Injection	get_it

4. Backend Technology Stack

Primary Backend – Supabase

Service	Usage
Auth	User login & OAuth

Service	Usage
PostgreSQL	Main database
Storage	Profile images & exports
Edge Functions	AI logic / APIs
Realtime	Community trend updates
Row Level Security	Data privacy

Optional Backend – Firebase

Service	Usage
Firebase Cloud Messaging	Push notifications
Firebase Analytics	Usage insights
Crashlytics	Error tracking

5. Database Technology

Engine

- **Supabase PostgreSQL 15+**

ORM / Query Layer

- Supabase Dart SDK

Core Tables

Table	Purpose
users	Profiles & preferences
health_logs	Sleep, water, steps
symptoms_logs	User symptom entries
recommendations	AI suggestions
community_trends	Aggregated data
reminders	Notification schedules

Security

- Row Level Security Policies
- Encrypted sensitive columns
- UUID primary keys

6. AI / Machine Learning Stack

HealthMate AI uses **Hybrid AI Approach**.

A. Symptom Classification

Layer	Technology
Model	TensorFlow Lite / ONNX
Language Processing	NLP preprocessing
Deployment	On-device or Edge Function

Purpose: Urgency prediction + basic condition mapping.

B. Habit Pattern Analysis

Method	Technology
Time Series	Python / Dart logic
Algorithms	Rolling average, anomaly detection

C. Recommendation Engine

| Type | Hybrid Rule + ML |
| Hosting | Supabase Edge Functions |

D. Chatbot NLP

Option	Technology
Offline	Ollama / Local LLM
Online	Gemini / OpenAI API
Guardrails	Prompt filtering & restrictions

7. API & Microservices Stack

API Style

- RESTful APIs

- JSON responses

Hosting

- Supabase Edge Functions (Deno)
- Optional Node.js server

Example Services

Service	Purpose
/symptom/analyze	AI classification
/insights/generate	Trend analysis
/chat/respond	NLP assistant
/community/stats	Aggregated trends

8. Authentication & Security Stack

Feature	Technology
Auth Provider	Supabase Auth
OAuth	Google / GitHub
Token	JWT
Encryption	AES-256
Transport	HTTPS TLS 1.3
RBAC	Role-based policies

9. Storage Stack

Storage Type	Technology
User Files	Supabase Storage
Local Cache	Hive / SharedPreferences
Secure Local	flutter_secure_storage

10. Notification Stack

Type	Tool
Local Reminders	flutter_local_notifications
Push Notifications	Firebase Cloud Messaging

11. Analytics & Monitoring

Tool	Purpose
Firebase Analytics	User behavior
Crashlytics	Crash logs
Supabase Logs	API performance
Sentry (Optional)	Error monitoring

12. DevOps & Deployment Stack

Layer	Technology
Version Control	Git + GitHub
CI/CD	GitHub Actions
Build	Flutter CLI
Testing	Flutter Test
Distribution	Firebase App Distribution / Play Console
Environment Config	dotenv

13. Testing Stack

Type	Tool
Unit Testing	flutter_test
Widget Testing	flutter_driver
API Testing	Postman
Load Testing	k6

Type	Tool
Security Testing	OWASP ZAP

14. Performance Optimization Stack

- Flutter DevTools
- Lazy loading lists
- Isolates for AI processing
- Image compression
- Query indexing in PostgreSQL

15. Accessibility Stack

- Flutter Semantics
- Screen Reader Support
- Dynamic Font Scaling
- Color Contrast Validation

16. Scalability Strategy

Layer	Strategy
Database	Horizontal scaling
AI	Edge Functions autoscale
Frontend	Lazy loading
Storage	CDN caching

17. Cost Optimization (Free Tier Strategy)

Service	Free Tier Benefit
Supabase	500MB DB + Auth
Firebase	Free notifications
Flutter	Free framework
Ollama	Free local AI

18. Future Stack Expansion

- Kubernetes for AI microservices
- GraphQL APIs

- Wearable SDK integrations
- HIPAA-grade encryption modules

19. Technology Selection Rationale

Need	Chosen Tech	Reason
Cross Platform	Flutter	Single codebase
Backend	Supabase	Open source + free tier
AI	Hybrid	Performance + cost
Notifications	Firebase	Reliable push
Charts	fl_chart	Lightweight
State	Riverpod	Scalability

20. Conclusion

The HealthMate AI technical stack is designed to deliver **enterprise-grade reliability, AI capability, scalability, and security** while remaining **cost-efficient and hackathon-feasible**. The hybrid use of **Flutter + Supabase + AI microservices + optional Firebase** ensures rapid development without sacrificing long-term extensibility or professional architecture standards.