Software Requirements Specification (SRS)

Version 1.0

Software Requirements Specification

For

Personalized Health Plans

**Project Title:** Personalized Health Companion

**Prepared by:**

Vignesh K

Sathish V

Pranavesh S

Yogithadevi V

Pravina A

Vedha K

**Organization:** Sri Manakula Vinayagar Engineering College

**Date Created:** 11/8/2024

**Table of Contents**

1. **Introduction - Page 4**
   1. Purpose - Page 4
   2. Scope - Page 4
   3. Definitions, Acronyms, and Abbreviations - Page 4
   4. References - Page 5
   5. Overview - Page 5
2. **Overall Description - Page 5**
   1. Product Perspective - Page 5
   2. Product Features - Page 6
   3. User Classes and Characteristics - Page 6
   4. Operating Environment - Page 6
   5. Design and Implementation Constraints - Page 7
   6. Assumptions and Dependencies - Page 7
3. **System Features - Page 8**
   1. Risk Prediction Model - Page 8
   2. Personalized Health Plan - Page 10
   3. Real-Time Data Monitoring - Page 13
   4. Prescription Management - Page 16
   5. Hospital Visit Tracking - Page 19
   6. Medicine Ordering Feature - Page 22
4. **External Interface Requirements - Page 25**
   1. User Interfaces - Page 26
   2. Application Programming Interfaces (APIs) - Page 27
   3. Hardware Interfaces - Page 28
   4. Communication Interfaces - Page 29
5. **System Non-Functional Requirements - Page 30**
   1. Performance Requirements - Page 30
   2. Safety Requirements - Page 30
   3. Security Requirements – Page 31
   4. Software Quality Attributes - Page 31
   5. Other Requirements - Page 31
6. **Other Requirements - Page 32**
   1. Legal and Regulatory Requirements - Page 32
   2. Documentation Requirements - Page 32
   3. Dependencies on Third-Party Products - Page 33
   4. Performance Requirements - Page 33
   5. Usability Requirements - Page 34
   6. Ethical Requirements - Page 34
   7. Cultural and Political Requirements - Page 35
   8. Environmental Requirements - Page 35

**1. Introduction**

**1.1 Purpose**

The purpose of this SRS document is to provide a detailed description of the Personalized Health Companion AI system. This software is designed to address the growing need for personalized and proactive health management by predicting potential health risks and generating tailored health plans for users. The intended audience includes software developers, designers, project managers, healthcare professionals, and stakeholders involved in the system's development and deployment.

**1.2 Scope**

The Personalized Health Companion AI system will:

* Predict potential health risks and diseases based on real-time health data.
* Generate personalized health plans including diet, exercise, and medication schedules.
* Continuously monitor health data from wearable devices and notify users of significant changes.
* Manage and store user prescriptions, track hospital visits, and check for potential interactions between different medications.
* Recommend specialized doctors based on their medical needs and location.

The system will not:

* Replace professional medical advice or diagnosis.
* Provide emergency medical services or immediate intervention.
* Handle payment processing for medication orders or doctor consultations.

**1.3 Definitions, Acronyms, and Abbreviations**

* AI: Artificial Intelligence
* SRS: Software Requirements Specification
* ML: Machine Learning
* UI: User Interface
* API: Application Programming Interface
* Azure Cloud: Microsoft Azure Cloud Platform
* Google Auth: Google Authentication Service

**1.4 References**

* Azure Cloud Documentation: <https://docs.microsoft.com/en-us/azure/>
* Google Authentication API: <https://developers.google.com/identity>
* Machine Learning for Health Datasets: <https://www.kaggle.com/health-datasets>

**1.5 Overview**

This document outlines the software requirements for the Personalized Health Companion AI system. It will cover system objectives, detailed system features, and the functional and non-functional requirements. The document is structured to provide clarity on the expected functionalities and serves as a guide for the development team.

**2. Overall Description**

**2.1 Product Perspective**

The "Personalized Health Companion" is designed as a comprehensive health management system that fits seamlessly within the broader digital health ecosystem. It is not a standalone product but rather an integrated solution that interacts with various external systems, such as wearable health monitoring devices, electronic health records, and pharmacy systems.

The platform leverages these integrations to provide personalized health plans and real-time risk predictions, enhancing the user's ability to manage their health proactively. It is part of a larger suite of health-related applications, potentially linking with other health and wellness platforms to provide a holistic approach to personal health management.

**2.2 Product Features**

Key features of the Personalized Health Companion include:

1. **Risk Prediction Model:**
   * Predicts potential health risks and diseases using real-time data and machine learning algorithms.
   * Generates risk scores for various health conditions.
2. **Personalized Health Plan:**
   * Provides customized health recommendations based on user data and risk predictions.
   * Includes tailored diet plans, exercise routines, and preventive measures.
3. **Real-Time Data Monitoring:**
   * Tracks health data continuously from wearable devices and user inputs.
   * Sends alert notifications for significant changes or potential health issues.
4. **Prescription Management:**
   * Manages and stores user prescriptions.
   * Tracks hospital visits and reasons for visits.
   * Checks for potential interactions between different medications.
5. **Medicine Ordering:**
   * Allows direct ordering of prescribed medications through the platform.
   * Provides order tracking and status updates.

**2.3 User Classes and Characteristics**

1. **General Users:**
   * Individuals managing their personal health.
   * Varying levels of technological proficiency.
   * May include adults of all ages, individuals with chronic conditions, and health-conscious people seeking to optimize wellness.
2. **Healthcare Providers:**
   * Medical professionals accessing patient data securely.
   * Includes doctors, specialists, nutritionists, and fitness trainers who may collaborate on personalized health plans.
3. **Pharmacy Partners:**
   * Pharmacies processing medication orders.
   * Managing deliveries and inventory.

**2.4 Operating Environment**

* The platform operates on AWS Cloud.
* Accessible via web browsers and mobile applications.
* Integrates with various wearable devices for health data collection.
* Utilizes external APIs for data monitoring and medicine ordering.

**2.5 Design and Implementation Constraints**

1. **Data Security:**
   * Must comply with health data regulations (e.g., HIPAA, GDPR).
   * Implement robust encryption and access controls.
2. **Real-Time Processing:**
   * Efficient handling and analysis of real-time health data.
   * Minimal latency for risk predictions and alerts.
3. **Device Compatibility:**
   * Integration with a wide range of wearable health monitoring devices.
   * Support for various data formats and communication protocols.
4. **Scalability:**
   * Ability to support an increasing number of users and growing data volumes.
   * Efficient data storage and retrieval mechanisms.
5. **User Interface:**
   * Design must be intuitive and accessible to users with varying levels of tech proficiency.
   * Support for multiple languages and cultural adaptations.

**2.6 Assumptions and Dependencies**

1. **Assumptions:**
   * Users will utilize compatible wearable devices for health data collection.
   * Reliable internet access is available for real-time data synchronization.
   * Users are willing to share their health data for personalized recommendations.
2. **Dependencies:**
   * External APIs (e.g., wearable device APIs, pharmacy systems) are stable and accessible.
   * Continuous compliance with evolving health regulations is maintained.
   * Availability of up-to-date health datasets for training and refining AI models.
   * AWS Cloud services for hosting, data processing, and scalability.

**3. System Features**

**3.1 Risk Prediction Model**

**Description**

The Risk Prediction Model is the cornerstone of the Personalized Health Companion, employing advanced machine learning algorithms to analyze user health data and predict potential health risks and diseases. This feature aims to provide early warnings and personalized insights to users, enabling proactive health management.

**Functional Requirements**

* FR1.1: The system shall collect and process user health data from connected devices and manual inputs.
  + FR1.1.1: Support integration with at least 5 major health monitoring device brands (e.g., Fitbit, Apple Watch, Garmin, Withings, Omron).
  + FR1.1.2: Allow manual input of health metrics such as weight, blood pressure, glucose levels, and symptom reporting.
  + FR1.1.3: Implement data validation to ensure accuracy of manually entered data (e.g., range checks, format validation).
  + FR1.1.4: Support data import from electronic health records (EHRs) using standard formats (e.g., HL7, FHIR).
  + FR1.1.5: Implement data normalization techniques to standardize data from various sources.
* FR1.2: The system shall use ML algorithms to analyze health data and generate risk predictions.
  + FR1.2.1: Implement at least three different ML models (e.g., Random Forest, Neural Networks, Gradient Boosting) for risk prediction.
  + FR1.2.2: Update risk predictions at least once every 24 hours or when significant new data is received.
  + FR1.2.3: Utilize ensemble methods to combine predictions from multiple models for improved accuracy.
  + FR1.2.4: Implement feature importance analysis to identify key factors contributing to each risk prediction.
  + FR1.2.5: Continuously retrain models using federated learning techniques to improve prediction accuracy while maintaining user privacy.
* FR1.3: The system shall display risk scores and predictions to users in an easily understandable format.
  + FR1.3.1: Provide a numeric risk score (e.g., 1-100) for each predicted health risk.
  + FR1.3.2: Categorize risks into low, medium, and high categories with clear definitions for each category.
  + FR1.3.3: Implement trend analysis to show how risk scores have changed over time.
  + FR1.3.4: Provide comparative analysis showing user's risk levels compared to demographic peers.
  + FR1.3.5: Generate plain-language explanations for each risk prediction, describing key contributing factors.
* FR1.4: The system shall provide recommended actions based on the risk predictions.
  + FR1.4.1: Generate at least three actionable recommendations for each identified risk.
  + FR1.4.2: Prioritize recommendations based on the severity of the risk and the potential impact of the action.
  + FR1.4.3: Tailor recommendations to user's personal preferences and lifestyle factors.
  + FR1.4.4: Provide scientific references or explanations for each recommendation.
  + FR1.4.5: Allow healthcare providers to review and modify recommendations for their patients.

**User Interface Requirements**

* UI1.1: The risk prediction results shall be displayed in a visually appealing dashboard.
  + UI1.1.1: Use intuitive icons or graphics to represent different health risks.
  + UI1.1.2: Implement interactive elements to allow users to explore details of each risk prediction.
  + UI1.1.3: Provide a summary view showing overall health risk status at a glance.
  + UI1.1.4: Implement responsive design to ensure optimal viewing on various devices (desktop, tablet, mobile).
  + UI1.1.5: Include accessibility features (e.g., screen reader support, keyboard navigation) for users with disabilities.
* UI1.2: The dashboard shall include color-coded risk levels for easy interpretation.
  + UI1.2.1: Use green for low risk, yellow for medium risk, and red for high risk.
  + UI1.2.2: Provide a legend explaining the color coding system.
  + UI1.2.3: Implement options for color-blind friendly palettes.
  + UI1.2.4: Allow users to customize color schemes based on personal preferences.
  + UI1.2.5: Use consistent color coding across all sections of the application for coherence.

**Acceptance Criteria**

* The risk prediction model achieves an accuracy of at least 85% in identifying potential health risks, as validated through cross-validation and testing on a holdout dataset.
* Risk predictions are updated in real-time (within 5 minutes) when new significant data is received.
* 90% of users surveyed find the risk prediction dashboard easy to understand and navigate.
* The system can handle data input and risk calculation for at least 100,000 concurrent users without performance degradation.
* Risk predictions comply with relevant healthcare regulations and standards (e.g., HIPAA, GDPR) as verified by legal and compliance teams.

**Dependencies**

* Integration with health monitoring device APIs
* Access to a comprehensive health dataset for initial model training
* Secure cloud infrastructure for data storage and processing
* Compliance with healthcare data protection regulations
* Collaboration with medical professionals for model validation and recommendation generation

**3.2 Personalized Health Plan**

**Description**

The Personalized Health Plan feature generates customized health recommendations and plans based on the risk prediction model, user profile, health goals, and lifestyle factors. It aims to provide users with actionable, tailored guidance to improve their health outcomes.

**Functional Requirements**

* FR2.1: The system shall create customized diet plans based on user health data and risk factors.
  + FR2.1.1: Generate weekly meal plans considering user preferences, dietary restrictions, and nutritional needs.
  + FR2.1.2: Provide nutritional information for each recommended meal, including macronutrient breakdown and calorie count.
  + FR2.1.3: Allow users to swap meals while maintaining nutritional balance.
  + FR2.1.4: Integrate with popular recipe databases to provide diverse meal options.
  + FR2.1.5: Adjust meal plans based on user feedback and adherence data.
* FR2.2: The system shall generate personalized exercise routines considering user fitness levels and health conditions.
  + FR2.2.1: Offer a variety of exercise types (e.g., cardio, strength training, flexibility, balance) suitable for user's fitness level.
  + FR2.2.2: Adjust exercise intensity and duration based on user progress and feedback.
  + FR2.2.3: Provide video demonstrations or detailed instructions for each recommended exercise.
  + FR2.2.4: Integrate with fitness tracking devices to monitor exercise adherence and performance.
  + FR2.2.5: Implement adaptive algorithms to progressively increase workout difficulty as user fitness improves.
* FR2.3: The system shall provide medication schedules aligned with user prescriptions.
  + FR2.3.1: Send reminders for medication doses through multiple channels (e.g., push notifications, SMS, email).
  + FR2.3.2: Track medication adherence and provide detailed reports to users and authorized healthcare providers.
  + FR2.3.3: Implement drug interaction checking to alert users of potential conflicts between medications.
  + FR2.3.4: Provide information on proper medication storage and potential side effects.
  + FR2.3.5: Allow users to log side effects or concerns related to their medications.
* FR2.4: The system shall suggest preventive measures based on identified risk factors.
  + FR2.4.1: Recommend relevant health screenings and check-ups based on user's age, gender, and risk factors.
  + FR2.4.2: Provide educational content on risk mitigation strategies, including articles, videos, and interactive modules.
  + FR2.4.3: Suggest lifestyle modifications tailored to user's specific health risks and personal preferences.
  + FR2.4.4: Implement a goal-setting feature allowing users to set and track health-related objectives.
  + FR2.4.5: Provide access to health coaching services for additional support and guidance.

**User Interface Requirements**

* UI2.1: The health plan shall be presented in an easy-to-follow, interactive format.
  + UI2.1.1: Implement a calendar view for scheduling health activities, with options for daily, weekly, and monthly views.
  + UI2.1.2: Use progress bars or similar visual indicators to show advancement towards health goals.
  + UI2.1.3: Provide a customizable dashboard where users can prioritize different aspects of their health plan.
  + UI2.1.4: Implement a drag-and-drop interface for rescheduling activities within the health plan.
  + UI2.1.5: Include a virtual assistant or chatbot to guide users through their health plan and answer questions.
* UI2.2: Users shall be able to mark tasks as completed and track their progress.
  + UI2.2.1: Provide a daily checklist of health-related tasks with satisfying visual feedback for completed items.
  + UI2.2.2: Offer weekly and monthly progress summaries with detailed analytics and insights.
  + UI2.2.3: Implement a point or reward system to gamify task completion and motivate users.
  + UI2.2.4: Allow users to set reminders for specific tasks within their health plan.
  + UI2.2.5: Provide options to share progress or achievements on social media platforms.

**Acceptance Criteria**

* Health plans are updated automatically within 24 hours when there are significant changes in user health data or risk predictions.
* Users report a satisfaction rate of at least 80% with the personalized recommendations, as measured through in-app surveys.
* At least 70% of users show improvement in at least one health metric within 3 months of following the personalized health plan.
* The system can generate personalized health plans for 10,000 new users within 1 hour during peak registration periods.
* 95% of medication reminders are delivered within 1 minute of the scheduled time.

**Dependencies**

* Risk Prediction Model
* Integration with nutrition databases for meal planning
* Access to a database of exercise routines and instructional videos
* Integration with medication databases for drug information and interaction checking
* Collaboration with nutritionists, fitness experts, and healthcare providers for plan validation

**3.3 Real-Time Data Monitoring**

**Description**

The Real-Time Data Monitoring feature continuously tracks user health data from connected devices and manual inputs, providing timely alerts for significant changes or anomalies. This feature aims to enable rapid response to health issues and support ongoing health management.

**Functional Requirements**

* FR3.1: The system shall integrate with various health monitoring devices to collect real-time data.
  + FR3.1.1: Support data collection from at least 10 different types of health metrics (e.g., heart rate, blood pressure, blood glucose, sleep patterns, activity levels, body temperature, oxygen saturation).
  + FR3.1.2: Implement error handling for device disconnections or data transmission failures, including automatic retry mechanisms.
  + FR3.1.3: Support both continuous data streams (e.g., heart rate monitoring) and periodic measurements (e.g., daily weight check).
  + FR3.1.4: Implement data validation algorithms to detect and flag potentially erroneous readings.
  + FR3.1.5: Provide a manual data entry option for metrics that cannot be automatically collected.
* FR3.2: The system shall analyze incoming data to detect significant changes or anomalies.
  + FR3.2.1: Define thresholds for normal ranges for each health metric, customizable by healthcare providers.
  + FR3.2.2: Implement algorithms to detect sudden changes, trends, or patterns in health data.
  + FR3.2.3: Utilize machine learning models to identify personalized anomalies based on individual user baselines.
  + FR3.2.4: Implement multi-parameter analysis to detect complex health events (e.g., combining heart rate and activity data).
  + FR3.2.5: Provide a confidence score for each detected anomaly to indicate the certainty of the finding.
* FR3.3: The system shall send real-time notifications to users when attention is required.
  + FR3.3.1: Allow users to set preferred notification methods (e.g., in-app, SMS, email) and frequency.
  + FR3.3.2: Provide different urgency levels for notifications based on the severity of the detected anomaly.
  + FR3.3.3: Implement a smart notification system that learns from user responses to reduce alert fatigue.
  + FR3.3.4: Include actionable information in notifications, such as recommended next steps or relevant health information.
  + FR3.3.5: Provide an option for notifications to be sent to designated emergency contacts for critical health events.

**User Interface Requirements**

* UI3.1: A real-time health dashboard displaying current health metrics.
  + UI3.1.1: Use graphs and charts to visualize trends in health data, with options for different time frames (e.g., daily, weekly, monthly, yearly).
  + UI3.1.2: Implement interactive elements allowing users to zoom in on specific time periods or data points.
  + UI3.1.3: Provide a summary view of all monitored health metrics with clear indications of normal, borderline, and abnormal values.
  + UI3.1.4: Include a feature to compare multiple health metrics on the same graph to visualize potential correlations.
  + UI3.1.5: Implement customizable widgets allowing users to prioritize display of specific health metrics.
* UI3.2: Configurable alert settings for users to customize notification thresholds.
  + UI3.2.1: Allow users to set personal thresholds for each health metric, with guidance on recommended ranges.
  + UI3.2.2: Provide a simple interface for enabling/disabling specific types of alerts.
  + UI3.2.3: Implement a scheduling system for alerts, allowing users to set quiet hours or specific alert times.
  + UI3.2.4: Include an option to temporarily suspend alerts (e.g., during exercise for heart rate alerts).
  + UI3.2.5: Provide a log of all generated alerts, allowing users to review past notifications and the data that triggered them.

**Acceptance Criteria**

* The system processes and analyzes data within 5 seconds of receipt for 99.9% of data points.
* Alerts are sent within 10 seconds of detecting a significant change or anomaly for 99% of cases.
* False positive rate for anomaly detection is less than 5%, as verified through periodic audits and user feedback.
* 95% of critical health alerts (e.g., extremely high blood pressure) are acknowledged by users within 30 minutes.
* The system can handle real-time data processing for up to 1 million concurrent users without performance degradation.
* Data synchronization with connected devices is successful on the first attempt for at least 95% of sync operations.

**Dependencies**

* Integration with health monitoring device APIs
* Reliable push notification service with high throughput capabilities
* Machine learning infrastructure for real-time anomaly detection
* High-performance, scalable database for storing and querying time-series health data
* Compliance with data protection regulations for handling sensitive health information

**3.4 Prescription Management**

**Description**

The Prescription Management feature allows users to store, track, and manage their medication prescriptions, ensuring proper adherence to treatment plans and providing a comprehensive overview of a user's medication history.

**Functional Requirements**

* FR4.1: The system shall allow users to input and store prescription information.
  + FR4.1.1: Support manual entry of prescription details (medication name, dosage, frequency, duration, prescribing doctor, pharmacy).
  + FR4.1.2: Implement optical character recognition (OCR) to scan and import prescription details from images or PDF documents.
  + FR4.1.3: Allow direct import of prescription data from partnered healthcare providers and pharmacies.
  + FR4.1.4: Implement a medication database with auto-complete functionality to ensure accurate and standardized medication names.
  + FR4.1.5: Support entry of as-needed (PRN) medications with specific usage instructions.
* FR4.2: The system shall track medication dosage and administration schedules.
  + FR4.2.1: Generate a medication schedule based on prescription details, accounting for multiple medications and potential interactions.
  + FR4.2.2: Track remaining quantities of medications and alert users when refills are needed, considering prescription duration and refill rules.
  + FR4.2.3: Support complex dosing schedules, including tapering doses and medications taken at specific times of day.
  + FR4.2.4: Implement a dose tracking system allowing users to log when they've taken their medication.
  + FR4.2.5: Provide a visual calendar view of medication schedules, clearly showing dosing times across multiple medications.
* FR4.3: The system shall send reminders for medication doses.
  + FR4.3.1: Allow users to customize reminder timing and frequency for each medication.
  + FR4.3.2: Support snooze and dismiss options for reminders, with follow-up reminders for snoozed notifications.
  + FR4.3.3: Implement smart reminders that adjust based on user behavior and medication adherence patterns.
  + FR4.3.4: Provide audio, visual, and haptic reminder options to accommodate different user preferences and needs.
  + FR4.3.5: Allow caregivers or designated contacts to receive medication reminder notifications for the user, with proper consent.
* FR4.4: The system shall allow users to share prescription information with healthcare providers securely.
  + FR4.4.1: Generate shareable reports of medication history and adherence, exportable in common formats (PDF, CSV).
  + FR4.4.2: Implement secure data sharing protocols compliant with health data regulations (e.g., HIPAA, GDPR).
  + FR4.4.3: Provide an option for real-time sharing of medication data with authorized healthcare providers.
  + FR4.4.4: Implement an audit trail for all data sharing activities, viewable by the user.
  + FR4.4.5: Allow users to set expiration dates for shared access to their medication data.

**User Interface Requirements**

* UI4.1: An intuitive interface for adding and managing prescriptions.
  + UI4.1.1: Provide a searchable database of common medications to simplify data entry, including generic and brand names.
  + UI4.1.2: Implement a drag-and-drop interface for adjusting medication schedules visually.
  + UI4.1.3: Use icons and color coding to represent different medications and dosage forms (e.g., pills, injections, topical applications).
  + UI4.1.4: Provide a "medication profile" view for each prescription, showing all relevant details in one place.
  + UI4.1.5: Implement a version history feature allowing users to view changes to their prescriptions over time.
* UI4.2: A calendar view of medication schedules.
  + UI4.2.1: Use color coding to distinguish between different medications in the calendar view.
  + UI4.2.2: Allow users to view medication schedules alongside other health-related activities and appointments.
  + UI4.2.3: Provide day, week, and month view options for the medication calendar.
  + UI4.2.4: Implement a timeline view showing medication doses throughout the day.
  + UI4.2.5: Include a printable version of the medication schedule for offline reference.

**Acceptance Criteria**

* Users can successfully add, edit, and delete prescriptions with 99% accuracy, as verified through user testing and error logging.
* Medication reminders are sent accurately according to the set schedule, with a system reliability of 99.9%.
* 85% of users report improved medication adherence after using the prescription management feature for 3 months, as measured through in-app surveys and adherence data.
* The system can handle at least 10,000 concurrent prescription updates without performance degradation.
* Data sharing with healthcare providers is completed successfully within 30 seconds for 95% of attempts.
* The OCR functionality for prescription scanning achieves at least 90% accuracy in extracting correct medication details.

**Dependencies**

* Integration with a comprehensive medication database for accurate drug information
* Secure cloud storage for prescription data with encryption at rest and in transit
* OCR technology for prescription scanning feature
* Integration with healthcare provider systems for direct prescription import
* Compliance with relevant healthcare data protection regulations

**3.5 Hospital Visit Tracking**

**Description**

The Hospital Visit Tracking feature allows users to record and manage details of their hospital visits, creating a comprehensive medical history. This feature aims to provide users and their healthcare providers with a complete overview of past medical encounters, improving continuity of care.

**Functional Requirements**

* FR5.1: The system shall allow users to log hospital visit details.
  + FR5.1.1: Capture key information such as date, hospital name, doctor seen, reason for visit, and diagnosis.
  + FR5.1.2: Allow users to attach relevant documents or images (e.g., discharge summaries, test results, prescriptions).
  + FR5.1.3: Implement a voice-to-text feature for users to dictate visit details.
  + FR5.1.4: Provide a template system for common visit types (e.g., annual check-up, specialist consultation) to streamline data entry.
  + FR5.1.5: Allow users to set follow-up reminders associated with specific hospital visits.
* FR5.2: The system shall store information about healthcare providers, visit reasons, and treatments.
  + FR5.2.1: Maintain a database of user's healthcare providers with contact information and specialties.
  + FR5.2.2: Categorize visit reasons for easy filtering and reporting (e.g., routine check-up, emergency, follow-up).
  + FR5.2.3: Track treatments and procedures performed during each visit.
  + FR5.2.4: Implement a feature to record prescribed medications and link them to the Prescription Management system.
  + FR5.2.5: Allow users to rate and review their hospital visits for personal reference.
* FR5.3: The system shall provide a historical view of all hospital visits.
  + FR5.3.1: Implement search and filter functionality for past visits based on date, doctor, hospital, and reason.
  + FR5.3.2: Generate summary reports of hospital visit history, including visualizations of visit frequency and types.
  + FR5.3.3: Provide a timeline view of hospital visits in relation to other health events and metrics.
  + FR5.3.4: Allow users to compare details across multiple visits (e.g., changes in vital signs or test results over time).
  + FR5.3.5: Implement a feature to export hospital visit history in various formats (e.g., PDF, CSV) for personal records or sharing with healthcare providers.

**User Interface Requirements**

* UI5.1: A form for inputting hospital visit details.
  + UI5.1.1: Use auto-complete functionality for hospital and doctor names based on previous entries and a database of healthcare providers.
  + UI5.1.2: Implement a user-friendly date picker for selecting visit dates, with options for recurring visits.
  + UI5.1.3: Provide a drag-and-drop interface for attaching documents and images to visit records.
  + UI5.1.4: Implement a dynamic form that adjusts required fields based on the type of visit selected.
  + UI5.1.5: Include a feature to easily duplicate previous visit details for quick entry of similar visits.
* UI5.2: A timeline view of past hospital visits.
  + UI5.2.1: Display visits chronologically with key information visible at a glance (e.g., date, doctor, reason).
  + UI5.2.2: Allow users to expand each visit entry for more detailed information.
  + UI5.2.3: Implement color coding or icons to distinguish between different types of visits.
  + UI5.2.4: Provide interactive elements to filter and sort the timeline based on various criteria.
  + UI5.2.5: Include a feature to zoom in/out of the timeline to focus on specific time periods.

**Acceptance Criteria**

* Users can successfully log and retrieve hospital visit information with 100% accuracy, as verified through data integrity checks.
* The system maintains an accurate and complete history of all recorded hospital visits, with no data loss.
* 90% of users report that the hospital visit tracking feature helps them better manage their healthcare, as measured through user surveys.
* The system can handle the import of at least 1000 historical hospital visit records per user within 5 minutes.
* Search and filter operations on hospital visit history return results within 2 seconds for 95% of queries.
* The voice-to-text feature for logging visit details achieves at least 95% accuracy in transcription.

**Dependencies**

* Secure cloud storage for hospital visit data with appropriate backup and recovery mechanisms
* Integration with the user's personal calendar for optional visit scheduling and reminder setting
* Natural Language Processing (NLP) capabilities for the voice-to-text feature
* Integration with the Prescription Management system for linking visit-related prescriptions
* Compliance with healthcare data protection regulations for storing and processing medical history

**3.6 Medicine Ordering Feature**

**Description**

The Medicine Ordering Feature enables users to order prescribed medications directly through the platform, streamlining the process of obtaining necessary treatments. This feature aims to improve medication adherence by making the procurement process more convenient and efficient.

**Functional Requirements**

* FR6.1: The system shall integrate with partner pharmacies for medication ordering.
  + FR6.1.1: Maintain a network of at least 100 partner pharmacies across major cities, with options for both chain and independent pharmacies.
  + FR6.1.2: Implement real-time inventory checking with partner pharmacies to confirm medication availability.
  + FR6.1.3: Support multiple fulfillment options, including in-store pickup and home delivery.
  + FR6.1.4: Implement a pharmacy rating and review system to help users choose their preferred pharmacy.
  + FR6.1.5: Provide options for users to set preferred pharmacies for automatic routing of orders.
* FR6.2: The system shall allow users to place orders for prescribed medications.
  + FR6.2.1: Verify that ordered medications match user prescriptions stored in the Prescription Management system.
  + FR6.2.2: Support recurring orders for long-term medications with customizable frequency.
  + FR6.2.3: Implement a feature for users to easily reorder previously purchased medications.
  + FR6.2.4: Allow users to set up automatic refills for eligible prescriptions.
  + FR6.2.5: Provide options for users to add non-prescription items to their order (e.g., over-the-counter medications, medical supplies).
* FR6.3: The system shall provide order tracking and status updates.
  + FR6.3.1: Integrate with pharmacy and courier systems for real-time status updates on order processing and delivery.
  + FR6.3.2: Provide estimated delivery times based on user location, pharmacy processing times, and courier information.
  + FR6.3.3: Implement a map-based tracking interface for orders that are out for delivery.
  + FR6.3.4: Allow users to communicate with the pharmacy or delivery service through the app for order-related queries.
  + FR6.3.5: Provide options for users to modify or cancel orders before they are processed by the pharmacy.
* FR6.4: The system shall send notifications about order status and delivery.
  + FR6.4.1: Send push notifications for key order milestones (order confirmed, processed by pharmacy, shipped, out for delivery, delivered).
  + FR6.4.2: Allow users to opt-in for SMS notifications in addition to app notifications.
  + FR6.4.3: Implement a notification preference center where users can customize which updates they receive and how.
  + FR6.4.4: Send reminders for upcoming automatic refills with options to modify or cancel the order.
  + FR6.4.5: Provide notifications for any issues with the order, such as out-of-stock medications or delivery delays.

**User Interface Requirements**

* UI6.1: An easy-to-use interface for browsing and ordering medications.
  + UI6.1.1: Implement a search function with filters (e.g., medication name, prescribed, recently ordered, over-the-counter).
  + UI6.1.2: Provide clear pricing information and insurance coverage details where applicable, including options to compare prices across different pharmacies.
  + UI6.1.3: Display medication images and detailed information (e.g., usage instructions, side effects) on product pages.
  + UI6.1.4: Implement a shopping cart feature allowing users to review and modify their order before checkout.
  + UI6.1.5: Provide a streamlined checkout process with options for saved payment methods and delivery addresses.
* UI6.2: A tracking page showing order status and estimated delivery time.
  + UI6.2.1: Use a visual progress bar to show current order status with clear labeling of each stage.
  + UI6.2.2: Provide a map view for orders that are out for delivery, with real-time updates of courier location.
  + UI6.2.3: Display a detailed order history with options to reorder or set up recurring orders.
  + UI6.2.4: Implement a feature to easily contact customer support or the pharmacy directly from the tracking page.
  + UI6.2.5: Provide options to rate and review the ordering experience after delivery is completed.

**Acceptance Criteria**

* Users can successfully place and track medication orders with a 99% success rate, as measured by order completion statistics.
* Order status updates are provided in real-time, with a maximum delay of 5 minutes for 95% of status changes.
* 95% of orders are delivered within the estimated delivery window communicated to the user.
* User satisfaction with the medicine ordering feature is at least 4.5 out of 5 based on post-order surveys.
* The system can handle peak ordering periods (e.g., flu season) with the ability to process up to 10,000 orders per hour without performance degradation.
* Integration with partner pharmacies achieves 99.9% uptime for inventory checking and order placement.

**Dependencies**

* Integration with pharmacy APIs for inventory checking and order placement
* Partnership with reliable courier services for medication delivery
* Secure payment processing system for handling transactions
* Integration with the Prescription Management system for verifying and linking orders to prescriptions
* Compliance with regulations regarding online pharmacy services and medication dispensing
* Real-time inventory management system to ensure accuracy of stock information

**4. External Interface Requirements**

**4.1 User Interfaces (UI)**

The Personalized Health Companion shall provide a web-based user interface that is intuitive, accessible, and responsive across various devices and platforms.

**4.1.1 Web Interface**

* FR-UI-1: The system shall provide a web-based interface accessible via modern web browsers (Chrome, Firefox, Safari, Edge).
  + FR-UI-1.1: The web interface shall be responsive and optimized for desktop, tablet, and mobile viewing.
  + FR-UI-1.2: The interface shall adhere to WCAG 2.1 Level AA accessibility standards.
  + FR-UI-1.3: The web application shall support keyboard navigation for all features.
* FR-UI-2: The web interface shall include a customizable dashboard.
  + FR-UI-2.1: Display personalized health metrics, risk scores, health recommendations, and alerts.
  + FR-UI-2.2: Allow users to customize their dashboards to prioritize information based on their health goals.
  + FR-UI-2.3: Provide interactive charts, graphs, and data visualizations for users to explore their health data.
* FR-UI-3: The web interface shall provide intuitive navigation.
  + FR-UI-3.1: Implement easy navigation to different sections such as Risk Prediction, Health Plan, Prescription Management, and Medicine Ordering.
  + FR-UI-3.2: Include an integrated search bar for quick access to health records, prescriptions, and recommendations.
* FR-UI-4: The web interface shall include comprehensive user settings.
  + FR-UI-4.1: Allow users to manage personal details, preferences, and health goals.
  + FR-UI-4.2: Provide configuration options for alerts and reminders.
  + FR-UI-4.3: Include data privacy settings for managing data sharing permissions, including sharing with healthcare providers.
* FR-UI-5: The web interface shall provide real-time health monitoring features.
  + FR-UI-5.1: Implement browser notifications for immediate alerts related to health data changes or prescription reminders.
  + FR-UI-5.2: Provide quick actions for emergency contacts, reordering medicines, or accessing health records.
* FR-UI-6: The web interface shall integrate with wearable devices.
  + FR-UI-6.1: Display real-time health data from connected wearable devices.
  + FR-UI-6.2: Allow users to view live data such as heart rate, steps, and sleep patterns directly on their web interface.

**4.1.2 Common UI Requirements**

* FR-UI-7: The user interface shall implement consistent branding and design language.
* FR-UI-8: The system shall provide a dark mode option for the web interface.
* FR-UI-9: The interface shall support localization in at least 5 languages (English, Spanish, French, German, and Mandarin).

**4.1.3 Data Visualization**

* FR-UI-10: The system shall provide interactive data visualization components.
  + FR-UI-10.1: Implement zoomable and pannable charts for displaying health trends.
  + FR-UI-10.2: Provide customizable dashboards where users can select and arrange their preferred health metrics.

**4.1.4 Accessibility**

* FR-UI-11: The web interface shall be compatible with common screen readers (e.g., JAWS, NVDA, VoiceOver).
* FR-UI-12: The system shall provide options for adjusting font sizes and contrast ratios.

**4.2 Application Programming Interfaces (APIs)**

The Personalized Health Companion shall interact with various external software systems and APIs to enhance functionality and data integration.

**4.2.1 Health Monitoring Devices API**

* FR-API-1: The system shall interface with major health data platforms and devices.
  + FR-API-1.1: Implement APIs for integrating with wearable devices (e.g., Fitbit, Apple Watch, Garmin) to gather health data such as heart rate, steps, and sleep patterns.
  + FR-API-1.2: Support data exchange with popular health platforms (e.g., Apple HealthKit, Google Fit, Samsung Health).
* FR-API-2: The system shall ensure secure transmission of collected health data.
  + FR-API-2.1: Implement encryption for all data in transit.
  + FR-API-2.2: Provide real-time data synchronization capabilities.

**4.2.2 Electronic Health Records (EHR) API**

* FR-API-3: The system shall provide interfaces for integration with EHR systems.
  + FR-API-3.1: Implement HL7 FHIR-based APIs for data exchange with EHR systems.
  + FR-API-3.2: Support OAuth 2.0 for secure authorization with EHR portals.
  + FR-API-3.3: Enable data import and export functionality for user health records, prescriptions, and visit history.
  + FR-API-3.4: Ensure compatibility with standard EHR formats (e.g., HL7, FHIR) for seamless data exchange.

**4.2.3 Pharmacy Integration API**

* FR-API-4: The system shall interface with pharmacy management systems for medication ordering.
  + FR-API-4.1: Implement a standardized API for real-time inventory checking with partner pharmacies.
  + FR-API-4.2: Provide a secure interface for transmitting prescription orders to pharmacies.
  + FR-API-4.3: Implement order tracking functionality to provide real-time updates on the status of medicine orders, including shipment tracking.

**4.2.4 Payment Gateways**

* FR-API-5: The system shall integrate with secure payment gateways for processing medication orders and subscription payments.
  + FR-API-5.1: Support major payment processors including Stripe, PayPal, and Square.
  + FR-API-5.2: Implement secure payment processing within the web interface.

**4.2.5 Machine Learning Services**

* FR-API-6: The system shall interface with Azure-based machine learning services for risk prediction and data analysis.
  + FR-API-6.1: Utilize Azure Machine Learning for training and deploying machine learning models.
  + FR-API-6.2: Implement Azure Cognitive Services for advanced AI capabilities such as natural language processing and computer vision.
  + FR-API-6.3: Use Azure Databricks for big data analytics and collaborative machine learning projects.

**4.2.6 Notification Services**

* FR-API-7: The system shall integrate with web-based notification services for user alerts and reminders.
  + FR-API-7.1: Implement Web Push API for browser notifications.
  + FR-API-7.2: Utilize Azure Communication Services for email and SMS notifications.

**4.3 Hardware Interfaces**

The Personalized Health Companion shall interface with various hardware devices to collect health data.

**4.3.1 Wearable Devices**

* FR-HW-1: The system shall interface with wearable fitness trackers and smartwatches.
  + FR-HW-1.1: Support data collection from devices using Bluetooth Low Energy (BLE) protocol via web browsers that support Web Bluetooth API.
  + FR-HW-1.2: Implement interfaces for major brands including Fitbit, Apple Watch, Garmin, and Samsung Galaxy Watch.
  + FR-HW-1.3: Support Wi-Fi connectivity for devices that offer this feature, where applicable.

**4.3.2 Health Monitoring Devices**

* FR-HW-2: The system shall interface with home health monitoring devices.
  + FR-HW-2.1: Support data collection from smart scales for weight and body composition data.
  + FR-HW-2.2: Interface with blood pressure monitors that support wireless data transmission.
  + FR-HW-2.3: Collect data from continuous glucose monitors for diabetic users.

**4.3.3 Computer Sensors**

* FR-HW-3: The web application shall utilize available computer sensors when permissions are granted.
  + FR-HW-3.1: Use the webcam for features such as pill identification (where supported by the browser).
  + FR-HW-3.2: Utilize geolocation for location-based services (with user permission).

**4.3.4 External Medical Devices**

* FR-HW-4: The system shall provide an API for integration with external medical devices used in clinical settings.
  + FR-HW-4.1: Support HL7 FHIR standards for medical device data integration.
  + FR-HW-4.2: Implement a secure protocol for pairing with hospital-based devices during user hospital visits.

**4.4 Communication Interfaces**

The Personalized Health Companion shall implement various communication protocols and standards to ensure secure and efficient data exchange.

**4.4.1 Network Protocols**

* FR-COM-1: The system shall use HTTPS for all client-server communications.
* FR-COM-2: Implement WebSocket protocol for real-time data updates in the web application.

**4.4.2 API Standards**

* FR-COM-3: All external APIs shall follow RESTful architecture principles.
* FR-COM-4: Implement GraphQL for complex data queries and to minimize over-fetching of data.

**4.4.3 Data Formats**

* FR-COM-5: The system shall support JSON for data interchange in all API communications.
* FR-COM-6: Implement XML support for integration with legacy healthcare systems that require it.

**4.4.4 Encryption and Security**

* FR-COM-7: All data transmissions shall be encrypted using TLS 1.3 or higher.
* FR-COM-8: Implement end-to-end encryption for sensitive health data transmissions.
* FR-COM-9: Utilize Azure Key Vault for managing encryption keys and secrets.

**4.4.5 Interoperability Standards**

* FR-COM-10: The system shall adhere to healthcare interoperability standards.
  + FR-COM-10.1: Implement support for HL7 FHIR R4 for exchanging electronic health records.
  + FR-COM-10.2: Support DICOM standard for handling medical imaging data when applicable.

**4.4.6 Authentication Protocols**

* FR-COM-11: The system shall implement OAuth 2.0 and OpenID Connect for user authentication and authorization.
* FR-COM-12: Support SAML 2.0 for enterprise single sign-on (SSO) integration.

**4.4.7 Secure Messaging**

* FR-COM-13: The system shall provide secure communication channels between users and healthcare providers.
  + FR-COM-13.1: Implement text, audio, and video messaging capabilities within the web interface using Azure Communication Services.
  + FR-COM-13.2: Ensure all communications are encrypted end-to-end.

**4.4.8 Notification System**

* FR-COM-14: The system shall support multiple notification channels.
  + FR-COM-14.1: Implement SMS notifications for critical alerts using Azure Communication Services (requires user phone number).
  + FR-COM-14.2: Provide email notifications for non-urgent communications using Azure Communication Services.
  + FR-COM-14.3: Utilize web push notifications for browser alerts when the web application is open.

**4.5 External System Interfaces**

The Personalized Health Companion shall interface with various external systems to provide comprehensive health management services.

**4.5.1 Third-Party Health Services**

* FR-EXT-1: The system shall integrate with insurance provider systems.
  + FR-EXT-1.1: Implement APIs for claims processing and coverage verification.
  + FR-EXT-1.2: Provide real-time eligibility checks for prescribed medications and treatments.
* FR-EXT-2: The system shall interface with telemedicine platforms.
  + FR-EXT-2.1: Enable integration with telemedicine services for virtual doctor consultations within the web interface.
  + FR-EXT-2.2: Provide secure data sharing of relevant health information during telemedicine sessions.

**4.5.2 Government Health Databases**

* FR-EXT-3: The system shall support reporting to national or regional health registries.
  + FR-EXT-3.1: Implement interfaces for reporting to chronic disease registries.
  + FR-EXT-3.2: Provide capabilities for epidemic tracking and reporting as required by regulations.

**4.5.3 Regulatory Compliance**

* FR-EXT-4: The system shall ensure compliance with healthcare regulations.
  + FR-EXT-4.1: Implement safeguards and processes to comply with HIPAA regulations.
  + FR-EXT-4.2: Ensure adherence to GDPR requirements for users in applicable regions.
  + FR-EXT-4.3: Support compliance with other regional data protection laws as applicable.
  + FR-EXT-4.4: Utilize Azure Compliance Manager to maintain and demonstrate compliance with various regulatory standards.

**4.5.4 Cloud Infrastructure**

* FR-EXT-5: The system shall utilize Azure cloud services for its infrastructure and operations.
  + FR-EXT-5.1: Use Azure App Service for hosting the web application.
  + FR-EXT-5.2: Implement Azure SQL Database for relational data storage.
  + FR-EXT-5.3: Utilize Azure Cosmos DB for NoSQL data storage requirements.
  + FR-EXT-5.4: Employ Azure Blob Storage for storing large unstructured data such as medical images or documents.
  + FR-EXT-5.5: Implement Azure API Management for managing and securing APIs.
  + FR-EXT-5.6: Use Azure Active Directory for identity and access management.
  + FR-EXT-5.7: Employ Azure Monitor and Azure Application Insights for monitoring and diagnostics.

**5. System Non-Functional Requirements**

**5.1 Performance Requirements**

* **Response Time:** The system should process and respond to user inputs within 2 seconds.
* **Throughput:** Capable of handling 10,000 simultaneous users without performance degradation.
* **Resource Usage:** Optimized to run efficiently on both mobile and desktop platforms.

**5.2 Safety Requirements**

* **Critical Applications:** The system must ensure that health data is accurately monitored and that alerts are reliable, as any failure could impact user safety.

**5.3 Security Requirements**

* **Data Protection:** All user data must be encrypted both in transit and at rest.
* **Authentication:** Implement multi-factor authentication via Google Authentication.
* **Authorization:** Role-based access control to ensure that only authorized users can access sensitive features and data.

**5.4 Software Quality Attributes**

* **Maintainability:** Code should be modular and well-documented to facilitate easy updates and maintenance.
* **Portability:** The system should be easily deployable across different cloud environments if needed.
* **Scalability:** The architecture should support scaling to accommodate an increasing number of users and data points.
* **Usability:** The user interface should be intuitive and accessible to users of all age groups.
* **Reliability:** The system should maintain 99.9% uptime and be resilient to failures.

**5.5 Other Requirements**

* **Compliance:** The system must comply with healthcare data regulations such as HIPAA.
* **Localization:** Support for multiple languages and regional health data standards.

**6. Other Requirements**

**6.1 Legal and Regulatory Requirements**

**6.1.1 HIPAA Compliance**

* **Data Encryption:** All user health data, including personal and medical information, must be encrypted both at rest and in transit.
* **Access Controls:** Implement role-based access control (RBAC) to ensure only authorized personnel have access to sensitive health data.
* **Audit Trails:** Maintain logs of all access to personal health information (PHI) to ensure accountability and traceability.
* **Data Breach Notification:** Notify users and authorities in a timely manner in the event of a data breach, as per HIPAA guidelines.

**6.1.2 GDPR Compliance (for European Users)**

* **User Consent:** Obtain explicit consent from users before collecting their personal data.
* **Right to Access:** Allow users to access their personal data stored within the system.
* **Right to Erasure:** Provide users with the ability to request the deletion of their personal data from the system.
* **Data Portability:** Enable users to download their data in a structured, commonly used format.

**6.1.3 FDA Regulations (for Medical Devices)**

* **Device Classification:** Determine the appropriate classification for the system based on intended use.
* **Pre-market Approval:** Obtain necessary approvals from the FDA before the product can be marketed.
* **Post-market Surveillance:** Monitor the product post-launch to ensure it continues to meet safety standards.

**6.2 Documentation Requirements**

**6.2.1 User Documentation**

* **User Manual:** Provide a detailed guide explaining system usage, including UI navigation, health data input, risk score interpretation, and medication ordering.
* **FAQ and Troubleshooting Guide:** Include a section dedicated to common issues and their resolutions.

**6.2.2 Technical Documentation**

* **System Architecture Document:** Detail the system's architecture, including component relationships.
* **API Documentation:** Document all APIs used or provided by the system, including endpoints, I/O formats, authentication methods, and error handling.
* **Database Schema Documentation:** Provide a detailed schema of all database tables, fields, and relationships, along with data flow diagrams.
* **Source Code Documentation:** Include in-line and standalone documentation for the codebase, explaining key modules, functions, and algorithms.

**6.2.3 Regulatory Compliance Documentation**

* **HIPAA/GDPR Compliance Reports:** Provide documentation proving compliance with relevant regulatory standards.
* **Risk Assessment Reports:** Include analysis of potential system risks and mitigation strategies.

**6.2.4 Deployment Documentation**

* **Installation Guide:** Provide instructions for deploying the system in a cloud environment, including Azure service configuration.
* **Backup and Recovery Procedures:** Document steps for system backup and restoration in case of data loss or system failure.

**6.3 Dependencies on Third-Party Products**

**6.3.1 APIs for Wearable Devices**

* Integrate with Fitbit API and Apple HealthKit API for collecting real-time health data.

**6.3.2 Pharmacy APIs**

* Integrate with partner pharmacy APIs for medicine ordering, inventory management, and prescription order tracking.

**6.3.3 Cloud Services**

* Utilize Azure services for cloud infrastructure:
  + Azure Blob Storage: For storing user health data, prescriptions, and related documents.
  + Azure SQL Database: For managing relational databases.
  + Azure Functions: For serverless computing to handle real-time data processing tasks.
  + Azure Monitor: For monitoring system performance and generating operational alerts.

**6.3.4 Machine Learning Libraries**

* Use TensorFlow/PyTorch for developing the risk prediction model, ensuring continuous updates and maintenance.

**6.3.5 Data Visualization Tools**

* Integrate Tableau/Power BI for creating and displaying interactive dashboards and detailed reports.

**6.4 Performance Requirements**

**6.4.1 Real-time Data Processing**

* Process health data from wearable devices in real-time with minimal latency for timely risk predictions and alerts.

**6.4.2 Scalability**

* Design the system to scale horizontally, accommodating user growth without performance compromise.

**6.4.3 Availability**

* Maintain a high availability of 99.9%, ensuring minimal downtime for user access to the platform and its features.

**6.5 Usability Requirements**

**6.5.1 User Interface**

* Design an intuitive interface allowing easy navigation, health data input, and result interpretation without requiring technical expertise.

**6.5.2 Accessibility**

* Comply with WCAG (Web Content Accessibility Guidelines) to ensure accessibility for users with visual, auditory, or motor impairments.

**6.6 Ethical Requirements**

* **Transparency:** Clearly communicate to users when AI is being used for recommendations or predictions.
* **Fairness:** Regularly audit AI algorithms to prevent discrimination against any group based on protected characteristics.
* **User Consent:** Obtain clear consent from users for data collection, storage, and processing.

**6.7 Cultural and Political Requirements**

* **Localization:** Support multiple languages and adapt to different cultural norms and healthcare practices.
* **Regional Health Standards:** Incorporate and adapt to different regional health standards and metrics.

**6.8 Environmental Requirements**

* **Energy Efficiency:** Design the system to minimize energy consumption, particularly in cloud resource usage.
* **Sustainable Practices:** Follow sustainable practices in development and maintenance processes where possible.