**Business Requirements Document (BRD)**

**Project Title:** HealthHub: Personal Health Management Web Application

**Date:** August 29, 2024

**1. Project Overview**

The Health Management System is designed to make managing health easier and more accessible for everyone. It offers user-friendly tools that allow individuals to track their physical health, monitor mental well-being, and manage diet plans and food intake. By simplifying the health management process, the system helps users who may struggle with maintaining a healthy lifestyle to take control of their health. With automated features and clear, actionable insights into their health status, users can make informed decisions, avoid health issues, and work towards a healthier future.

**2. Business Objectives**

* **Enhance Personal Health Management:** Provide a platform that empowers individuals to take control of their health by centralizing health data and providing actionable insights.
* **Increase Health Awareness:** Improve users' understanding of their health status and progress through comprehensive tracking and visualization of health metrics.
* **Support Lifestyle Changes:** Enable flexible goal setting and progress tracking to support users in making and maintaining positive lifestyle changes.
* **Improve Health Outcomes:** Ensure timely interventions and recommendations through personalized insights and alerts based on user data.

**3. Stakeholders**

* **Users:** Individuals who want to track and improve their health.
* **Healthcare Providers:** May integrate with the system to access user health data (future feature).
* **Nutritionists and Fitness Trainers:** May provide recommendations through the platform (future feature).

**4. Functional Requirements**

**4.1 User Management**

* **User Registration:** Users can create accounts with email and password.
* **Authentication:** Secure authentication mechanisms, including username and password, will be used to log in to the application.
* **Profile Management:** Users can update their profile information, including age, weight, height, and health goals.

**4.2 Health Metrics Tracking**

* **Metrics Logging:** Users can log various health metrics, including weight, blood pressure, heart rate, and water intake.
* **Goal Setting:** Users can set health-related goals, such as target weight or daily water intake.
* **Progress Tracking:** The application will track progress towards goals and provide visual representations of trends over time.
* **External API:** We have used external API to get the daily based health metrics of a user based on user data .We have used HEALTH CALCULATOR API from rapid API.

**Link:** <https://health-calculator-api.p.rapidapi.com>

**4.3 Diet Management**

* **Food Logging:** Users can log their meals and snacks, including details on portion sizes and nutritional content.
* **Nutritional Analysis:** The application will provide nutritional breakdowns of logged meals and compare them to recommended daily values.
* **Meal Planning:** Users can access meal recommendations based on their dietary preferences and health goals.
* **External API:** We have used external API to get the diet plan and nutrition analysis of a food intake by the user .We have used GEMINI API for Diet Plan, EDAMAM NUTRITION API for nutrition analysis.

**Links:** [Nutrition Analysis API - Nutrition Data Service - Edamam](https://developer.edamam.com/edamam-nutrition-api), <https://generativelanguage.googleapis.com/v1beta/models/gemini-1.5-flash-latest:generateContent>

**4.4 Exercise Tracking**

* **Activity Logging:** Users can log their exercise activities, including type, duration, and intensity.
* **Calorie Burn Estimation:** The application will estimate calories burned based on logged activities and user data.
* **Exercise Recommendations:** Users can receive personalized exercise recommendations based on their fitness level and goals.
* **External API:** We have used external API to get the exercise Recommedaions and based on activity calories burned.

**Link:**<https://trackapi.nutritionix.com/v2/natural/exercise>, <https://exercisedb.p.rapidapi.com>

**4.5 Stress Management**

* **Stress Reason :** Users can be able to give the reason for the Stress.
* **Relaxation Techniques:** The application will provide recommendations for stress-reduction techniques and exercises.
* **External API:** Based on reason and relaxation taken from user the API recommends the solution to get rid of Stress.

**Link:**<https://generativelanguage.googleapis.com/v1beta/models/gemini-1.5-flash-latest:generateContent>

**4.6 Reporting and Insights**

* **Health Dashboard:** Users can view a personalized dashboard summarizing their health metrics, progress towards goals, and key insights.
* **Trend Analysis:** The application will provide visualizations of health trends over time.
* **Recommendations:** Users will receive personalized health recommendations based on their data and progress.

**5. Non-Functional Requirements**

**5.1 Security**

* **Data Protection:** User data will be encrypted to ensure privacy and security.
* **Access Control:** Access to data will be controlled by the user.

**5.2 Performance**

* **Response Time:** The application should provide quick response times, with page loads and data updates happening within a few seconds.
* **Scalability:** The system should be able to handle a growing number of users and increasing data volumes.

**5.3 Usability**

* **User Interface:** The application will have a user-friendly interface that is intuitive and easy to navigate.
* **Mobile Responsiveness:** The web application will be responsive and functional on both desktop and mobile devices.

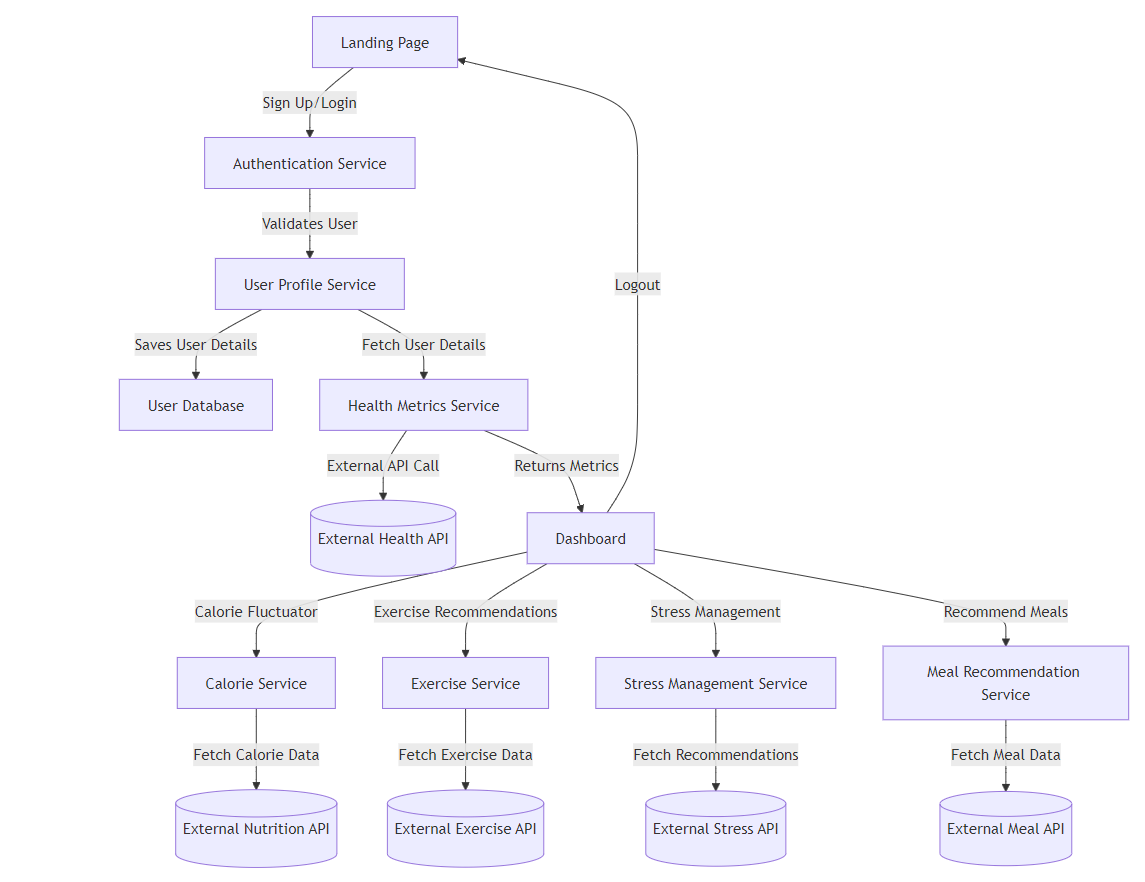
**5.4 Reliability**

* **Uptime:** The application should be available 99.9% of the time, with minimal downtime for maintenance and updates.
* **Data Accuracy:** The system will implement checks to ensure the accuracy of user-inputted data and calculations.

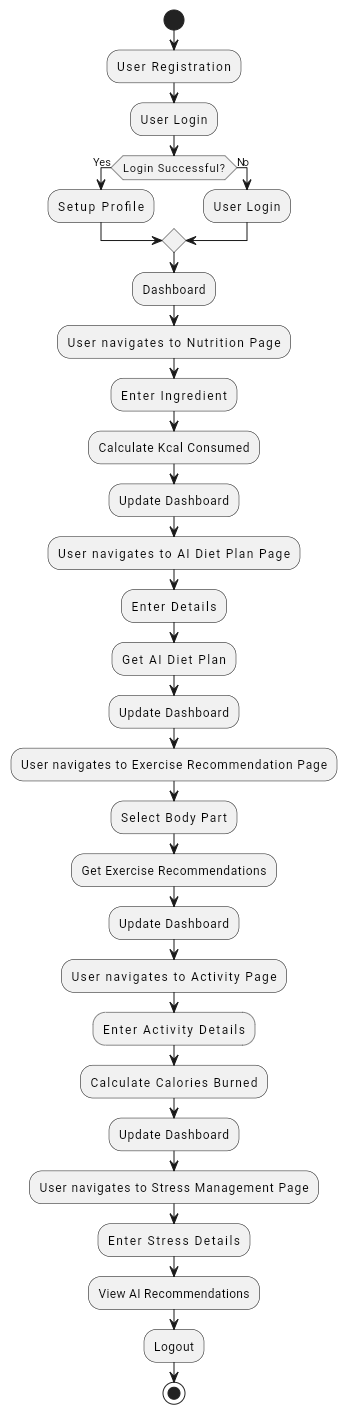
**6. Project Architecture**

The project architecture follows a microservices approach with a front end built using React and Redux, and a back end developed with Spring Boot. The system leverages multiple Eureka server-based microservices to handle core functionalities such as user management, health metrics tracking, diet management, exercise tracking, and recommendations. The database is MongoDB, chosen for its scalability and flexibility in managing varied health data.

* Front End: React with Redux for state management.
* Back End: Spring Boot with RESTful APIs.
* Microservices: Registered via Eureka servers for service discovery and load balancing.
* Database: MongoDB for efficient data storage and retrieval.



**7. Project Flow chart**

****

**8. Database Schema Overview**

The application will include the following collections in its MongoDB database schema:

* **Users:** Stores user details, including authentication information
* **HealthMetrics:** Contains health measurements such as weight, height, age etc.
* **Daily\_Metrics:** Contains required health measurements like blood pressure, and heart rate, water intake, etc
* **Diet:** Stores food intake information, including nutritional data.
* **Exercises:** Records exercise activities and related data.
* **Stress:** Records the stress level and the inputs of the user.



**9. Assumptions**

* Users will have access to stable internet connections.
* Users will regularly input their health data for the most accurate insights and recommendations.
* The application will integrate with external APIs for nutritional information and exercise data

**10. Constraints**

* Development and deployment timelines must be met to ensure the application is available for use within the specified project deadlines.

**11. Data Requirements**

* **User Data:** Information related to users, including authentication details, profile information, and preferences.
* **Health Metric Data:** Daily health measurements including weight, blood pressure, heart rate, and water intake.
* **Diet Data:** Details of food intake, including nutritional information and portion sizes.
* **Exercise Data:** Information about physical activities, including type, duration.
* **Data for Stress Management:** Taking inputs from the user to manage the mental health.
* **Recommendation Data:** Personalized health, diet, and exercise recommendations generated for users.