© Volume 2, Issue 10, October 2026 | JATIR
e-ISSN: 2839-5786, p-ISSN: 3139-8464

Smartfit – AI-Based Personalized Fitness, Diet Workout and Grooming Assistant

[1]Sayali Gaikwad, [2]Sakshi Landage, [3]Shrutika Shembade, [4]Sejal Dhekale
[5]Sanika Chavan
[1]*Assistant Professor Department of Information Technology Parvatibai Genba Moze College of Engineering, Wagholi, Pune, India*
[2,3,4,5]*Student Parvatibai Genba Moze College of Engineering, Wagholi, Pune, India*

***Abstract*—Fitness, nutrition and personal grooming are three closely linked aspects of everyday wellbeing, yet most individuals in India lack a single, affordable platform to manage all three together. Generic international fitness applications ignore Indian dietary patterns, regional food availability and household budgets, while grooming advice remains locked behind expensive consultations or scattered content. This paper presents SmartFit, a unified web-based assistant that lets a user build a personal profile and receive integrated diet, workout and grooming recommendations from a single dashboard. The system computes BMI and daily calorie requirement, generates a budget- and region-aware diet plan, recommends workouts with pre-/post-workout guidance, flags iron-deficiency-related food choices, auto-generates a grocery list, and offers grooming suggestions based on the user's skin/hair profile. Weekly progress is tracked using interactive charts, and a scheduler reminds the user to maintain adequate water intake. SmartFit is implemented as a responsive HTML/CSS/JavaScript prototype that is being progressively migrated to a Core Java, Spring Boot, Spring Data JPA, MySQL and JWT-based full-stack architecture, keeping the product demonstrable at every stage while building toward a secure, multi-user, production-grade system.**

***Index Terms*—SmartFit, personalized fitness, diet recommendation, grooming assistant, BMI, Spring Boot, MySQL, JWT authentication, web application.**

I. INTRODUCTION

Lifestyle-related health issues such as obesity, poor nutrition and stress-related skin problems are increasingly common among students and working professionals in India, driven by sedentary routines, irregular eating habits and limited access to personalised guidance. While awareness of fitness and self-care has grown considerably, the tools available to act on that awareness remain fragmented. A person who wants to lose weight, eat better, follow a workout routine and take basic

care of their skin today typically needs three or four separate applications, most of which are built for a global audience and do not account for Indian food items, regional cuisines, or household budget constraints.

Web technologies have matured to the point where a single, well-designed application can comfortably host profile management, rule-based recommendation logic, data visualisation and scheduled reminders without requiring heavyweight infrastructure. This makes it practical to build a genuinely usable, end-to-end wellness assistant rather than a narrow proof-of-concept limited to one feature. This paper presents SmartFit, a system-built module by module — starting with user profiling and BMI/calorie computation, followed by budget- and region-aware diet recommendation, workout recommendation, grooming/skincare suggestions, grocery list generation, progress tracking and a water-reminder scheduler — with each module fully working before the next is added.

II. LITERATURE SURVEY

1. World Health Organization BMI classification guidelines provide the standard BMI thresholds used to classify users into underweight, normal, overweight and obese categories, forming the basis of the profile module's health classification logic [1].
2. The Mifflin-St Jeor / Harris–Benedict equations for basal metabolic rate are widely used formulae for estimating daily calorie requirement from age, weight, height and activity level, and are referenced for the calorie-estimation module [2].
3. Studies on regional dietary pattern variation in India highlight significant differences in staple foods and meal composition across regions, motivating the region-aware filtering used in the diet recommendation module.
4. Research on mobile health (mHealth) application adoption and retention indicates that personalisation and single-app convenience significantly improve user retention compared to generic, one-size-fits-all fitness applications, supporting the unified-dashboard approach taken by SmartFit.
5. JSON Web Token (JWT), defined in RFC 7519, specifies the compact, self-contained token format used for stateless authentication between the front end and the REST API in the project's planned Spring Boot backend [3].

III. PROBLEM STATEMENT

Most individuals who want to improve their fitness, diet and grooming habits face three related gaps. First, popular fitness and diet-tracking applications are built for a global, generally western audience — their food databases, portion sizes and recipe suggestions do not map well onto Indian regional diets, and they rarely account for the household grocery budget a user is working within. Second, workout recommendation in these apps is usually generic and not tied to the user's own BMI, calorie target or daily schedule. Third, grooming and basic skincare guidance is almost

entirely absent from fitness applications, even though skin and hair health are directly affected by diet, hydration and lifestyle.

Because these needs are currently served by separate, disconnected tools — or not served at all in the case of grooming — users lack a single place to track fitness, diet and self-care together, and often lose motivation to continue any one of them in isolation. There is a clear need for an affordable, India-aware, single-dashboard system that ties fitness, diet and grooming recommendations to one consistent user profile.

IV. PROPOSED SYSTEM

The proposed system, SmartFit, is a single web application that ties fitness, diet, workout and grooming recommendations to one consistent user profile, specifically designed around Indian food habits, regional preferences and household budgets. Unlike existing tools, SmartFit does not treat diet, workout and grooming as separate products — all three read from, and contribute to, the same profile, so that recommendations in one area can take the others into account; for example, iron-deficiency food suggestions feed both the diet plan and the grooming guidance.

The system is being built and delivered in two stages. The current, fully working stage is implemented with a responsive HTML, CSS and JavaScript front end, using structured local data files to drive recommendation logic for diet, workout, grooming, grocery list generation, progress tracking (visualised with Chart.js) and a water-reminder scheduler. The second stage migrates this feature set onto a Core Java/Spring Boot backend with Spring Data JPA/Hibernate, a MySQL database, and JWT-based authentication exposed through REST APIs — converting the prototype into a secure, multi-user, deployable system without changing the validated user-facing feature set.

V. PROPOSED METHODOLOGY

Step 1 — Registration & Authentication

The user registers and logs in; in the full-stack version this is secured using JWT-based authentication issued by the Spring Boot backend.

Step 2 — Profile Creation

The user enters height, weight, age, gender, activity level, regional food preference, household grocery budget, and basic skin/hair details.

Step 3 — BMI & Calorie Computation

The system calculates BMI and estimates the user's daily calorie requirement from the profile data using standard formulae.

Step 4 — Diet Plan Generation
A diet plan is generated by filtering the food/recipe data set according to the user's budget range and regional preference, with iron-rich options highlighted where relevant.

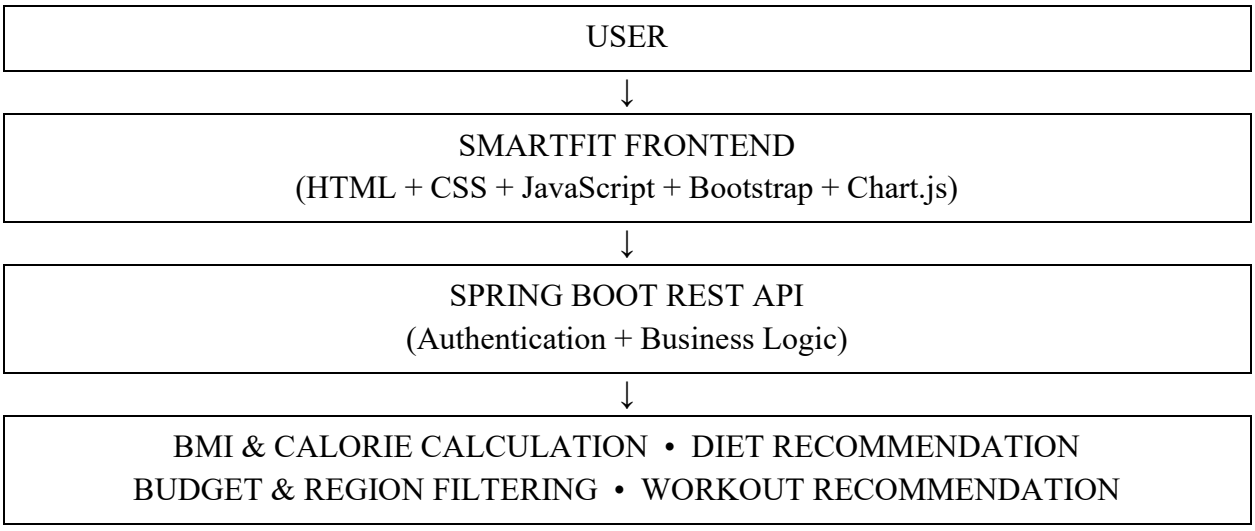
Step 5 — Workout Recommendation
Morning and evening workout routines are recommended along with pre-workout and post-workout suggestions appropriate to the user's profile.

Step 6 — Grooming & Grocery Output
General grooming/skincare suggestions are generated from the skin/hair profile, and a consolidated grocery list is produced from the recommended diet plan.

Step 7 — Tracking & Reminders
Weekly progress is logged and rendered as charts, and a scheduler (Spring Scheduler/@Scheduled in the full-stack version) issues periodic water-intake reminder. Each module is built and verified independently before the next is added, ensuring the system remains demonstrable and testable at every stage of development.

VI. SYSTEM ARCHITECTURE

Fig. 1 shows the layered architecture of SmartFit. The front end (HTML, CSS, JavaScript, Bootstrap and Chart.js) communicates with a Spring Boot REST API responsible for authentication and business logic. The API layer hosts the core recommendation engines — BMI/calorie calculation, budget- and region-aware diet filtering, workout recommendation, grooming/skincare suggestion, grocery generation and progress tracking — and persists data to a MySQL database. JWT secures communication between the front end and the API, Chart.js renders progress visualisations on the client, and Spring Scheduler drives periodic water-reminder notifications from the backend.



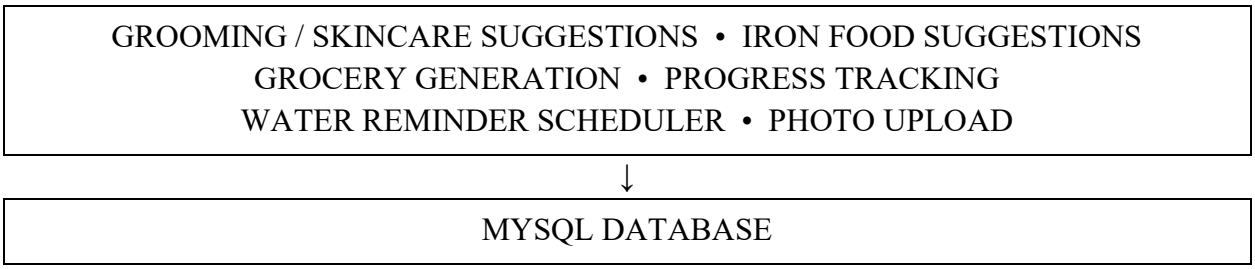


Fig. 1. Layered system architecture of SmartFit.

VII. MODULE DESCRIPTION

Table I summarises the responsibility of each functional module in the SmartFit system.

TABLE I
MODULE DESCRIPTION

Module	Responsibility
User Authentication	Handles registration, login and JWT-based authentication for secure, multi-user access.
Profile & BMI Module	Captures user details and computes BMI and estimated daily calorie requirement.
Diet Recommendation	Generates a diet plan filtered by household budget and regional food preference, with iron-rich options highlighted.
Workout Recommendation	Suggests morning/evening routines with pre-workout and post-workout guidance.
Grooming / Skincare	Provides general skincare and grooming suggestions based on the user's skin/hair profile.
Grocery Generator	Builds a consolidated, shareable grocery list from the recommended diet plan.
Progress Tracking	Logs weekly metrics and renders them as interactive Chart.js visualisations.
Water Reminder Scheduler	Issues periodic hydration reminders using a backend scheduler.
Photo Upload	Allows the user to attach a profile/progress photo to their account.

VIII. IMPLEMENTATION DETAILS

A. Hardware Requirements

- Processor: Intel i5 (8th Gen) or equivalent / AMD Ryzen 5 or higher

- RAM: 8 GB minimum; 16 GB recommended
- Storage: 5 GB free disk space
- Internet connectivity for development, package installation and deployment

B. Software Requirements

- Frontend: HTML5, CSS3, JavaScript, Bootstrap, Chart.js
- Backend: Java, Spring Boot, Spring MVC/REST APIs, Spring Data JPA, Hibernate
- Database: MySQL
- Authentication: JWT (JSON Web Token)
- Scheduler: Spring Scheduler / @Scheduled
- Tools: Maven, Git, GitHub, Postman
- IDE: VS Code / IntelliJ IDEA / Eclipse
- Operating System: Windows 10/11, Linux, macOS

IX. RESULTS AND EXPECTED OUTCOMES

- A fully functional, responsive web application covering profile, diet, workout, grooming, grocery, progress and reminder features.
- Personalised, budget- and region-aware diet and workout recommendations for each user.
- Automatically generated grocery lists tied to the active diet plan.
- Interactive, chart-based visualisation of weekly progress.
- A working demonstration of the migration path from a vanilla JavaScript prototype to a secure Spring Boot/MySQL/JWT backend.

X. ADVANTAGES AND LIMITATIONS

A. Advantages

- Single dashboard for diet, workout and grooming, reducing the need for multiple apps.
- Budget- and region-aware recommendations suited to Indian households.
- Incrementally verifiable architecture — each module is tested before the next is built.
- Fully open-source technology stack with no licensing cost.
- Clear, mentorship-friendly upgrade path from front-end prototype to secure full-stack system.

B. Limitations

- Recommendations are rule-based on structured data rather than machine-learning-driven personalisation at this stage.
- Grooming/skincare guidance is general in nature and does not replace professional dermatological advice.
- The current prototype is single-user/local; multi-user security depends on completing the Spring Boot/JWT migration.

- No integration yet with wearables or step-counting hardware.

XI. CONCLUSION AND FUTURE SCOPE

SmartFit demonstrates that a single, well-designed web application can bring together fitness, diet and grooming recommendations under one consistent, India-aware user profile, removing the friction of maintaining several disconnected tools. The staged development approach — a fully working vanilla JavaScript prototype evolving into a secure Spring Boot/MySQL/JWT system — keeps the product demonstrable at every stage while building toward a production-grade, multi-user deployment.

Future work includes completing the migration to Spring Boot, MySQL and JWT for secure, multi-user deployment; introducing AI-based body/skin photo analysis for more precise grooming suggestions; adding wearable and step-counter integration for automatic activity tracking; incorporating multilingual support (Hindi, Marathi) for wider accessibility; deploying the system as a hosted SaaS product accessible on mobile devices; and introducing optional telemedicine-style consultation links for advanced grooming and dermatology queries.

ACKNOWLEDGMENT

The authors thank the Department of Information Technology, Parvatibai Genba Moze College of Engineering, Wagholi, Pune, for the guidance and resources extended during the development of this project.

REFERENCES

[1]. World Health Organization, “Body Mass Index (BMI) classification,” WHO. [Online]. Available: https://www.who.int.

[2]. M. D. Mifflin et al., “A new predictive equation for resting energy expenditure in healthy individuals,” American Journal of Clinical Nutrition, vol. 51, no. 2, pp. 241–247, 1990.

[3]. M. Jones and P. Hazel, “JSON Web Token (JWT),” IETF RFC 7519, May 2015. [Online]. Available: https://www.rfc-editor.org/rfc/rfc7519.

[4]. Spring Documentation, “Spring Boot Reference Documentation,” Spring.io. [Online]. Available: https://docs.spring.io/spring-boot.

[5]. Oracle, “MySQL 8.0 Reference Manual,” Oracle Corporation. [Online]. Available: https://dev.mysql.com/doc.