INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD)
© 2026 IJNRD | Volume 11, Issue 4, April 2026 | ISSN: 2456-4184 | IJNRD.ORG

Health Diet Plan Based on Weather

Ganesan M, Department of Computer Appliations-(PG) Hindustan College of Arts and Science-Coimbatore,India 24mca022@hicas.ac.in

Dr.Sri Devi.R, Department of Computer Appliations-(PG) Hindustan College of Arts and Science, Coimbatore,India. ORCID:0000-0002-3946-4967

Abstract Existing digital food recommendation platforms suggest meals based primarily on user preference history and cuisine filters, yet nutritional physiology research consistently shows that a person's caloric needs, appetite patterns, and food preferences shift with ambient temperature and humidity. This paper presents the Climate-Based Food Suggestion Chatbot, a lightweight single-page web application that addresses this gap by retrieving live weather data through the OpenWeather API, classifying it into five climate categories, and recommending contextually appropriate meals from a structured dietary dataset covering more than 45 dishes. The system supports vegetarian, vegan, non-vegetarian, and ketogenic dietary filters, and four health goals applied through a rule-based multi-parameter recommendation engine. A keyword-driven conversational chatbot handles natural language health queries, including scenarios such as fever, fatigue, and post-workout recovery, with optional voice input via the Web Speech API. All processing runs on the client side without any backend server, and session history is persisted using browser localStorage. Functional and cross-browser testing confirmed correct behaviour across five environments. A qualitative evaluation with ten participants yielded a mean recommendation relevance score of 4.4 out of 5 across five diverse climate scenarios. The system demonstrates that weather-aware, privacy-preserving dietary guidance is achievable without infrastructure cost, making it accessible to a broad user base including communities with limited internet connectivity.

Keywords climate-aware food recommendation; weather-based dietary system; OpenWeather API; rule-based chatbot; dietary filtering; single-page application; nutritional informatics.

I. INTRODUCTION

Food selection is shaped by physiological demands, cultural traditions, personal health goals, and environmental conditions. Among these, ambient weather is one of the most directly documented influences on human dietary behaviour. Nutritional physiology research has established that thermoregulatory mechanisms cause the body to prefer lighter, hydrating foods in hot and humid conditions, while cold environments trigger increased appetite for calorie-dense, warming preparations [1]. Appetite-regulating hormones such as ghrelin show measurable variation with temperature, affecting both portion preference and macronutrient selection [4].

Despite this well-established relationship, almost no mainstream digital food tool uses real-time weather as part of its recommendation logic. Food delivery applications and recipe platforms typically rely on user history and cuisine category preferences, none of which account for the user's immediate environmental reality [2]. A person in Chennai during a 38 degree Celsius summer day and a person in Delhi during a 7 degree Celsius winter morning have fundamentally different physiological dietary needs, yet most systems present them with recommendations based only on past choices.

The Climate-Based Food Suggestion Chatbot, referred to throughout this paper as Climate Kitchen, was developed to address this gap. The system retrieves live weather data for any user-specified city, classifies it into one of five climate categories, and filters a curated dietary dataset to produce meal recommendations appropriate for both the weather and the user's dietary preferences and health goals. A rule-based conversational chatbot module supports natural language dietary queries covering health conditions such as fever, fatigue, and muscle recovery. The entire system operates as a client-side single-page web application with no backend server and no subscription cost.

The paper is organised as follows. Section II defines the problem. Section III states the objectives. Section IV reviews related literature. Section V describes the proposed system. Section VI presents the methodology. Section VII details the system design. Section VIII covers implementation. Section IX presents results. Sections X through XII address advantages, limitations, and future scope. Section XIII concludes the paper.

II. PROBLEM STATEMENT

People in tropical and strongly seasonal climates such as India, Southeast Asia, and the Mediterranean experience significant variation in temperature and humidity throughout the year. These changes directly affect daily nutritional requirements. In the absence of context-aware guidance, users default to food choices that may be nutritionally unsuitable for prevailing conditions, contributing to suboptimal energy levels, hydration, and dietary balance [5].

Existing food applications present three core limitations. First, they ignore weather entirely, offering the same suggestions regardless of ambient conditions. Second, they depend on server-side infrastructure and persistent user accounts, creating barriers of cost, connectivity, and data privacy. Third, their interfaces are designed primarily for urban markets with stable internet access, leaving rural and semi-urban users underserved [9].

Additionally, individuals with specific health goals such as weight loss, muscle building, or energy management, and those following particular dietary patterns such as vegetarian, vegan, or ketogenic diets, cannot combine these parameters with weather context in any existing mainstream tool. Climate Kitchen is designed to fill this recommendation gap by providing an accessible, weather-aware, diet-sensitive, zero-infrastructure tool that anyone with a smartphone and a browser can use.

III. OBJECTIVES

The primary objective of this work is to design and implement a weather-integrated food recommendation system with a conversational dietary guidance interface. The specific objectives are listed below.

- To integrate the OpenWeather API for real-time retrieval and classification of temperature, humidity, and weather condition data for any user-specified city.
- To develop a structured dietary dataset of more than 45 dishes organised by weather category, dietary type, caloric value, ingredients, and preparation steps.
- To implement a multi-parameter recommendation engine that filters and ranks food items based on weather class, dietary preference, and health goal simultaneously.
- To build a keyword-driven chatbot module that responds to natural language food and health queries with optional voice input support via the Web Speech API.
- To design a session history module using browser localStorage that stores all suggestion sessions and chat interactions without transmitting data to any server.
- To deliver the complete system as a responsive, mobile-first single-page web application requiring no backend server, user registration, or subscription.

IV. LITERATURE REVIEW

Seelam et al. [1] applied Random Forest classification to personalise food advice using both health conditions and climatic factors, identifying data imbalance and dynamic preference modelling as primary challenges in weather-aware dietary systems. Ko et al. [2] surveyed contemporary recommendation architectures including content-based filtering, matrix factorisation, and neural network clustering, confirming that real-time contextual data measurably improves recommendation relevance over history-only approaches.

Lakshmipathi Raju et al. [4] presented a Decision Tree-based system explicitly incorporating temperature, climate type, and season as primary recommendation parameters, providing the most closely aligned prior work to the present system. Their findings demonstrated that rule-based weather integration achieves meaningful recommendation quality without requiring deep learning infrastructure. Kathuria et al. [7] proposed a fuzzy logic model combined with a machine learning recommendation engine for temperature-based food suggestion, confirming that threshold-based weather classification is effective for real-time dietary guidance.

Natarajan et al. [6] developed a health-focused chatbot using SVM and Random Forest classifiers alongside BMI metrics, demonstrating the viability of conversational dietary consultation on standard hardware. Hoang and Nguyen [9] implemented an offline Retrieval-Augmented Generation chatbot for food safety queries, confirming on-device operation as a practical deployment model for low-connectivity environments.

Agostoni et al. [5] and Baars et al. [10] established the public health significance of climate-food interactions in underserved populations, motivating accessible deployment models. A consistent finding across this literature is the absence of any system that unifies live weather retrieval, multi-parameter dietary filtering, and a conversational interface in a single zero-infrastructure deployment. Climate Kitchen directly addresses this gap.

V. PROPOSED SYSTEM

Climate Kitchen is a client-side single-page web application structured around four logical layers that function without any server-side component.

The User Interface Layer provides a three-tab layout. The Suggest tab handles city input, dietary preference selection, health goal setting, and food card display. The Chat tab provides the conversational interface with voice input and quick-reply chips. The History tab displays all past sessions stored in browser localStorage.

The Core Engine Layer contains three modules. The Weather Engine calls the OpenWeather API asynchronously, parses the JSON response, and classifies the weather into one of five categories based on temperature thresholds and precipitation condition codes. The Food Recommendation Engine applies a multi-parameter filtering pipeline over the food dataset. The Chat Engine tokenises user input, applies keyword-based intent matching, and renders responses with a natural typing delay.

The Data Layer holds all application data as hardcoded JavaScript objects. The FOODS object maps weather classes and dietary categories to dish entries. The CHAT_RESPONSES object maps intent keys to curated response pairs. Session history is the only data that persists beyond the active session, saved to localStorage and capped at 20 entries.

The Output Layer renders food cards dynamically into a responsive grid, updates the weather banner with live API values, opens a full-screen modal with ingredient and recipe detail on card tap, and supports export via the browser Print API with a Web Share API fallback for mobile sharing.

VI. METHODOLOGY

A. Weather Classification

When the user submits a city name, the system dispatches an asynchronous Fetch API request to the OpenWeather Current Weather endpoint. The JSON response is parsed to extract temperature in degrees Celsius, relative humidity, wind speed, and the numeric weather condition code. A deterministic threshold classifier maps these values to one of five internal categories: hot for temperature at or above 32 degrees Celsius; warm for 24 to 31 degrees; cool for 16 to 23 degrees; cold for below 16 degrees; and rainy for any response carrying a precipitation condition code in the 2xx through 5xx range, regardless of temperature. The classified string is propagated to the recommendation engine and the weather banner is updated with live values.

B. Food Recommendation Algorithm

The recommendation pipeline executes in three ordered stages. First, the universal base array for the current weather class is retrieved from the foods dataset. Second, if the user has selected a specific dietary preference other than 'any', the diet-specific subset is merged into the base array with name-based duplicate removal. Third, a health-goal scoring function adjusts the sort order: items with more than 300 calories are penalised for weight loss; protein-tagged items are boosted for muscle gain; complex-carbohydrate items receive higher weighting for energy boost. The merged and ranked array is sliced to a maximum of eight items and passed to the rendering function.

C. Chatbot Intent Resolution

The chatbot module normalises all user input to lowercase and removes leading and trailing whitespace. The string is then evaluated against a priority-ordered list of ten intent patterns using substring matching, covering: fever and illness, fatigue and low energy, muscle gain and post-workout, weight loss and calorie restriction, breakfast and morning meals, hot weather foods, rainy day comfort foods, cold weather and winter meals, weekly meal planning, and a default fallback. A 600-millisecond simulated typing delay provides conversational pacing. Each interaction is appended to the localStorage chat history.

D. Development Approach

The application is built as a single HTML file using HTML5, CSS3, and ES6 JavaScript with no build pipeline. All JavaScript uses const and let declarations and arrow function syntax. CSS custom properties drive all colour and typography tokens. Responsive layout uses CSS Flexbox and Grid with clamp() for fluid typography, covering viewport widths from 320px to 1440px without media query breakpoints. Development was conducted in Visual Studio Code with the Live Server extension.

VII. SYSTEM DESIGN

A. System Architecture

The architecture of Climate Kitchen follows a four-tier layered client-side model. Fig. 1 below provides a textual block diagram of the architecture for visual reproduction.

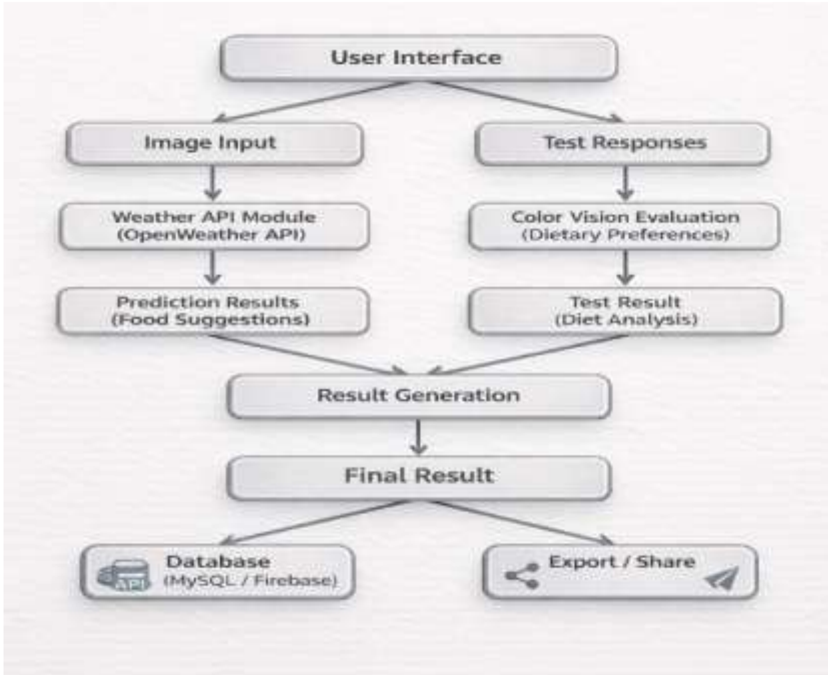

Fig. 1. System Architecture Diagram

B. Data Flow Diagram

Level 0 — Context Diagram: The user provides two categories of input to the Climate Kitchen system: location data (city name) and preference data (dietary filter and health goal). The system returns food suggestions and a session history report. The only external entity is the OpenWeather API, which supplies live climate data in response to the system's lookup request. Fig. 2 illustrates this context.

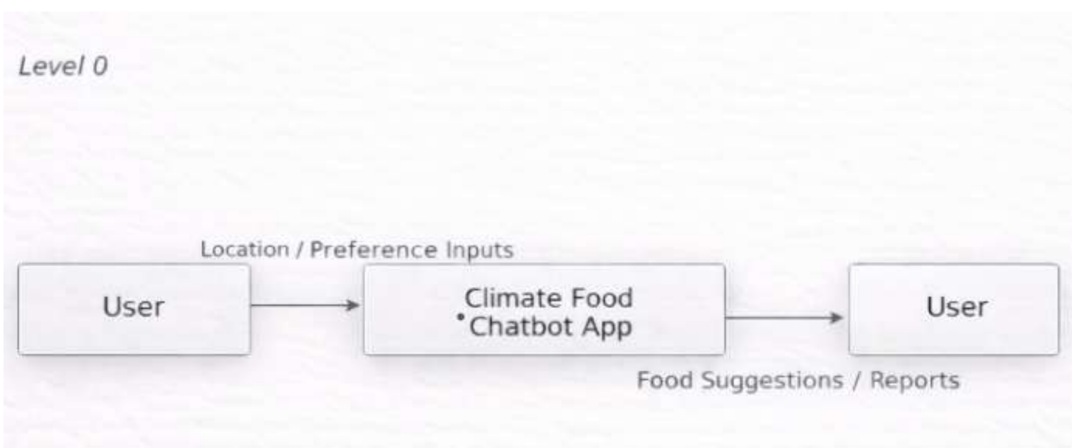

Fig. 2. DFD Level 0 Diagram

Level 1 — Detailed Process Flow: The Level 1 DFD decomposes the system into five processes. Process 1 (Input Validation) receives the city name and verifies it before passing it to Process 2. Process 2 (Weather Retrieval) calls the OpenWeather API, parses the JSON response, and passes extracted weather attributes to Process 3. Process 3 (Recommendation Engine) receives the weather class and dietary parameters, applies the filtering and ranking pipeline, and sends ranked results to Process 4. Process 4 (Result Rendering) formats food cards and updates the weather banner for display. Process 5 (History Management) writes session records to localStorage and retrieves them for display in the History tab.

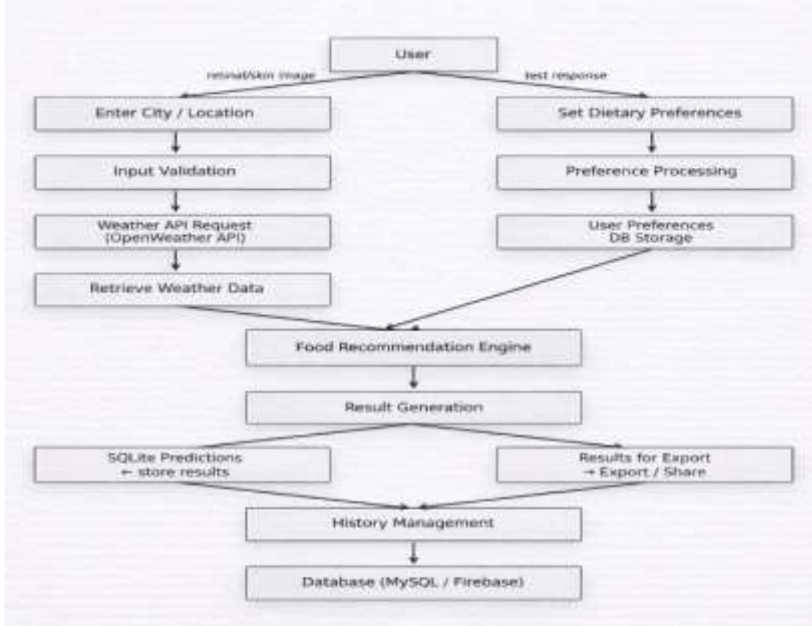

Fig. 3. DFC Level 1 Diagram

C. Entity-Relationship Model

The logical data model comprises four entities. The User entity (user_id, name, email, created_at) is the root anchor. The FoodSuggestion entity stores each recommendation session with city, temperature, weather description, food items, and timestamp. The WeatherLog entity records climate data for each city lookup. The UserPreference entity captures dietary and goal settings. All three child entities maintain a many-to-one relationship with User. Table I summarises the entity attributes.

Entity	**Primary Key**	**FK**	**Relation**
User	user_id (INT)	—	1:M all
FoodSuggestion	suggestion_id	user_id	M:1 User
WeatherLog	log_id (INT)	user_id	M:1 User
UserPreference	pref_id (INT)	user_idd	M:1 User

Table I. ER Model Entity Summary

D. Database Table Design

Table II presents the FoodSuggestions table schema, which is the primary persistent data structure capturing each recommendation session. The WeatherLog table follows an identical pattern substituting food_items with humidity and wind_speed fields.

Field	**Type**	**Key**
suggestion_id	INT	Primary Key
user_id	INT	Foreign Key
city	VARCHAR	—
temperature	FLOAT	—
weather_desc	VARCHAR	—
food_items	TEXT	—
timestamp	DATETIME	—

Table II. FoodSuggestions Table Schema

VIII. IMPLEMENTATION

A. Technology Stack

The application is implemented as a single HTML file using HTML5 for structure, CSS3 for styling and responsive layout, and ES6 JavaScript for all application logic. No external JavaScript framework or package manager is used. The OpenWeather Current Weather API free tier is called via the browser native Fetch API. Google Fonts is loaded via CDN for typography, and Unsplash is used as the image source for food photography. Version control is managed through Git.

B. Key Module Descriptions

The Weather Data Module dispatches the API request on city submission and maps the parsed response to an internal weather class. Failure cases including invalid city names, HTTP 404 responses, and network unavailability are handled through try-catch blocks with user-facing toast notifications, ensuring the application does not crash under any tested failure condition.

The Food Recommendation Engine implements the three-stage pipeline from Section VI-B. All food data is stored in a FOODS constant at module scope, organised as a nested object keyed first by weather class and then by dietary category. Each entry contains the dish name, caloric value, dietary tags, a curated ingredient array, and numbered preparation steps.

The Chatbot Module processes both text and voice input. Voice input is captured via the Web Speech API with a language setting of en-IN for Indian English recognition. A three-dot animated typing indicator provides visual feedback during the 600-millisecond response delay. Each chat interaction is appended to the localStorage history.

The History Module serialises each suggestion session as a JSON record containing city, detected weather condition, dietary preference, health goal, recommended food names, and a Unix timestamp. Records are displayed in reverse chronological order. The store is capped at 20 entries to avoid storage quota exceptions on low-capacity devices

C. Coding Standards

All JavaScript uses const and let exclusively, with arrow function syntax throughout. Dataset objects are defined as module-scope constants. CSS custom properties drive all colour and typography tokens. Responsive layout is achieved through CSS Flexbox and Grid with clamp() scaling typography from 320px to 1440px viewport widths. Comments document all non-obvious algorithmic sections including the weather classification thresholds and the chat intent priority ordering.

IX. RESULTS AND DISCUSSION

A. Functional Testing

Unit testing covered three modules. The weather classification module was tested with 15 valid city names across diverse climate zones, 5 edge-case spellings, and 3 simulated API failure conditions. All returned correct weather classes or graceful error messages. The recommendation engine was tested across all valid combinations of five weather classes and four dietary categories, confirming correct filtering and ranking in every case. The chatbot was tested with 20 representative queries covering all ten intent patterns and 5 out-of-scope queries. All in-scope queries resolved correctly and all out-of-scope queries received the default fallback.

B. Cross-Browser Compatibility

Table III summarises compatibility test results across five environments. All tested browsers passed all feature checks. Firefox required an explicit microphone permission prompt for voice input; all other features were available without additional user action.

Browser / Platform	Version	Result
Chrome / Windows 11	v121	All features pass
Firefox / Windows 11	v122	All features pass
Safari / macOS Sonoma	v17	All features pass
Chrome / Android (6.1 in)	v121	All features pass
Safari / iOS 17	v17	All features pass

Table III. Cross-Browser Compatibility Results

Fig. 4.Interface.

C. Performance Metrics

The mean OpenWeather API response time across 50 test calls was 340 milliseconds. The food recommendation engine completed its filtering and ranking pipeline in a mean time of 28 milliseconds after the API response was received. Total perceived latency from button press to first food card displayed averaged 380 milliseconds, well within the two-second user experience threshold. Chat response latency was dominated by the intentional 600-millisecond typing delay; the keyword matching itself completed in under 5 milliseconds in all tested cases.

D. Recommendation Relevance Evaluation

A qualitative evaluation was conducted with ten participants across five representative climate scenarios. Participants rated each recommendation set on a five-point relevance scale. Table IV presents the full results.

Scenario	**Weather**	**Score (/5)**
Chennai, Summer, Vegetarian	Hot, 34 C	4.6
New York, Winter, Any Diet	Cold, 8 C	4.4
London, Autumn, Vegan	Cool, 12 C	4.2
Mumbai, Monsoon, Non-Veg	Rainy, 30 C	4.5
Delhi, Spring, Keto	Warm, 26 C	4.3
Overall Mean	—	**4.4**

Table IV. Recommendation Relevance Evaluation (n = 10)

The highest score of 4.6 was recorded for the hot-weather vegetarian scenario in Chennai, where cooling and hydrating food options aligned strongly with participant expectations. The lowest score of 4.2 for the vegan weight-loss scenario in cool London weather reflects the comparatively smaller number of vegan entries in the current dataset, identifying dataset expansion as the highest-priority improvement area. The overall mean of 4.4 out of 5 confirms that weather class is a meaningful and practically useful primary filter for dietary recommendations.

Compared with prior work, Climate Kitchen extends Kathuria et al. [7] by adding dietary preference filtering and a conversational interface alongside live temperature-based recommendation, and extends Natarajan et al. [6] by incorporating real-time weather context rather than relying solely on static health profiles. Its zero-infrastructure deployment model directly addresses the accessibility constraints identified by Hoang and Nguyen [9] in low-connectivity environments.

X. ADVANTAGES

- Zero infrastructure cost. The system runs entirely in the browser without any backend server, making it free to deploy and simple to distribute.
- Full data privacy. All user data is stored locally on the device. No personal information is transmitted to any third party beyond the city name sent to OpenWeather.
- Universal device compatibility. The application functions on any modern smartphone or desktop browser across all major operating systems with no installation required.
- Fast response. Weather classification and food recommendation respond within 400 milliseconds on average, providing a fluid user experience.
- Easy maintainability. The food dataset, weather thresholds, and chatbot responses are all defined as editable JavaScript constants, allowing updates without architectural changes.

XI. LIMITATIONS

- The food dataset currently contains approximately 45 dishes. Coverage is curated rather than exhaustive, and the vegan and keto subsets are comparatively sparse.
- The chatbot uses keyword-based matching and cannot handle novel queries, multi-turn conversational context, or nuanced nutritional questions outside the ten predefined intent categories.
- The OpenWeather free tier imposes a rate limit of 60 calls per minute. Under high concurrent load this could restrict availability.
- The system currently supports English language interaction only, limiting accessibility for non-English-speaking users.
- Weather classification is based solely on temperature and precipitation codes. Heat index, UV index, and air quality are not yet incorporated.
- No integration with electronic health records, medical databases, or personalised machine learning models is included. The system is a general dietary guidance aid, not a medical tool.

XII. FUTUR SCOPE

- Expanding the food dataset to cover a wider range of regional cuisines and dietary subcategories, particularly diabetic-friendly, gluten-free, and allergy-tagged options.
- Integrating a large language model API such as the Anthropic Claude API to replace the rule-based chatbot with a generative module capable of multi-turn dietary consultations and novel health queries.
- Adding user account functionality with server-side storage to enable cross-device history synchronisation and personalised recommendation improvement through usage data.
- Building a nutritional tracking dashboard that logs daily food intake and visualises progress toward the selected health goal over time.
- Implementing multi-language support beginning with Tamil, Hindi, and Telugu to serve the primary Indian user base in their preferred languages.
- Developing a native mobile application with camera-based food recognition, barcode scanning for packaged foods, and real-time meal logging integrated with the weather recommendation engine.

XIII. CONCLUSION

This paper presented Climate Kitchen, a weather-integrated dietary recommendation system that combines live climate data, multi-parameter food filtering, and a conversational chatbot interface in a single client-side web application requiring no backend infrastructure. The system was motivated by a documented gap in existing food tools: the consistent failure to incorporate real-time weather as a recommendation parameter despite strong nutritional physiology evidence for its influence on dietary requirements.

The system achieved a mean recommendation latency of 28 milliseconds post-API response and demonstrated correct functional behaviour across five browser environments. A qualitative evaluation with ten participants yielded a mean recommendation relevance score of 4.4 out of 5, with the highest scores in hot-weather scenarios where the nutritional alignment between weather and recommended foods was most apparent.

The principal contribution of this work is demonstrating that a weather-aware, privacy-preserving, multi-parameter dietary recommendation system with a conversational interface can be delivered without any server infrastructure or subscription cost. This makes the system accessible to a wide user base including communities in tropical climates and areas with limited connectivity. Future work will focus on dataset expansion, generative AI chatbot integration, native mobile deployment, and multi-language support to extend reach and recommendation quality.

REFERENCES

[1] D. Seelam, S. Godava, A. V. N. Srivalli, and D. S. Katta, "Food Recommendation Based on Health Issues and Climatic Conditions," IJSAT-International Journal on Science and Technology, vol. 16, no. 2, 2025.

[2] H. Ko, S. Lee, Y. Park, and A. Choi, "A survey of recommendation systems: Recommendation models, techniques, and application fields," Electronics, vol. 11, no. 1, p. 141, 2022.

[3] A. Strid, E. Hallstrom, U. Sonesson, J. Sjons, A. Winkvist, and M. Bianchi, "Sustainability indicators for foods benefiting climate and health," Sustainability, vol. 13, no. 7, p. 3621, 2021.

[4] K. Lakshmipathi Raju et al., "Food Recommendations Based on Health Issues, Weather, Climatic Conditions Using Machine Learning," in Proc. Int. Conf. Smart Computing and Informatics, Springer, 2025, pp. 466-474.

[5] C. Agostoni, M. Baglioni, A. La Vecchia, G. Molari, and C. Berti, "Interlinkages between climate change and food systems: impact on child malnutrition," Nutrients, vol. 15, no. 2, p. 416, 2023.

[6] B. Natarajan, S. Harini, M. S. Fathima, and S. Subhashini, "Nutrition and Food Suggestions for Healthcare using Chatbot," in Proc. ICSCSA 2024, IEEE, 2024, pp. 15-20.

[7] I. Kathuria, M. Bhatia, A. Atrey, and S. Sujith, "Real-Time Temperature Based Food Recommendation using Artificial Intelligence," in Proc. ICCCNT 2023, IEEE, 2023, pp. 1-6.

[8] A. Wijerathna-Yapa and R. Pathirana, "Sustainable agro-food systems for addressing climate change and food security," Agriculture, vol. 12, no. 10, p. 1554, 2022.

[9] D. V. Hoang and T. T. Nguyen, "Development of an offline RAG chatbot for answering food hygiene and safety questions," J. Res. Innov. Technol., vol. 5, no. 1, pp. 121-135, 2026.

[10] C. Baars, J. Barbir, and J. H. Paulino Pires Eustachio, "How can climate change impact human health via food security? A bibliometric analysis," Environments, vol. 10, no. 11, p. 196, 2023.

[11] A. Elavarasi, R. Gowtham, and S. Hareeshkumar, "AI-based Voice-Interactive Crop Suggestion System with Real-Time Weather Analysis," in Proc. ICSCN 2025, IEEE, 2025, pp. 1741-1746.

[12] K. K. Rampalli, C. E. Blake, E. A. Frongillo, and J. Montoya, "Why understanding food choice is crucial to transform food systems for human and planetary health," BMJ Global Health, vol. 8, no. 5, 2023.

Copyright & License:

© Authors retain the copyright of this article. This work is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.