iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.80698

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Smart Chef: Design and Implementation of an AI-Powered Kitchen Assistant System

Varun Kirkire[1], Shubham Shinde[2], Aneesh Angane[3], Deepali Shrikhande[4]
[1, 2, 3]*Student, Information Technology, VIT, Mumbai, India*
[4]*Professor, Information Technology, VIT, Mumbai, India*

Abstract: ***Smart Chef refers to artificially intelligent assistant built for the kitchen to aid users in understanding recipes, accessing ingredients information, helping with shopping activities as well as assisting in cooking processes. Smart chef provides a Natural Language Processing backed recipe parsing mechanism, structured recipe dataset, and frontend as website for end to end experience. The survey paper on Smart Chef highlighted shortcomings of current solutions like lacking multimodal inputs, personalization, synchronization and smart shopping capabilities. This paper serves as an implementation description of the proposed solution discussed in the survey providing readers with an overview of design considerations, architectural decisions, modules and preliminary results of prototype developed focusing on ingredient extraction, smart shopping lists, recipe assistance and user workflow towards building an implementable solution for assisting with daily cooking activities. We employed a stack consisting of frontend, backend, AI and cloud datastore to build our responsive web based smart kitchen assistant. Our preliminary results as described in our Implementation Draft showcased great potential with our ingredient extraction providing a 94% accuracy, 2.3 seconds average runtime and positive initial feedback from users. In this paper we provide a short and focused implementation detail of Smart Chef compatible with IJARST guidelines[1].***

Keywords: Artificial Intelligence, Smart kitchen, Recipe Parsing, Natural Language Processing, Shopping Assistance, Website.

I. INTRODUCTION

Technology today allows users to go beyond saving recipes on their phones. There are smart kitchen assistants that help parse recipes, plan meals and make informed decisions. The Smart Chef survey paper shown that current smart kitchen applications tend to offer assistance with only one aspect of the kitchen user's experience. Whether it is finding new recipes to try or creating a shopping list, smart kitchen applications lack the ability to tie together recipe parsing, ingredient awareness, cooking assistance and personalized assistance. This led to the inspiration of Smart Chef as an implementation driven course project to help users make use of unstructured data on the internet to help with cooking related activities on a day-to-day basis.

The Smart Chef system we have implemented aims to assist students, working professionals and home chefs who wish to quickly turn a recipe given as text or a link into neatly organized ingredients, a shopping list plan, as well as step-by-step cooking support. We combined AI powered natural language understanding with a lightweight web frontend to allow recipe management to become quicker, tidier, and accessible on any device. Rather than having wide scope literature discussion like the previously submitted survey paper[2], This paper concentrates on the discussion of the final system realization, module integration, engineering decisions, and results. Goals for our implementation were to automatically parse Induce intelligent shopping support, give cuisine backing, and maintain an extensible armature for unborn advancements. The design also aims to demonstrate that an academic prototype can address real- world cuisine problems through a usable and technically coherent software design[15].

II. SYSTEM DESIGN

A. Overall Architecture

The Smart Chef prototype follows a layered armature conforming of frontend, backend, AI- processing, and pall data factors. The frontend is enforced using a ultramodern web frame to give responsive commerce, while the backend handles request routing, data processing, and service collaboration. The AI- processing subcaste is responsible for interpreting form textbook and rooting useful realities, and the data subcaste stores stoner- related and form- related information in a accompanied form[14].

This modular design improves maintainability and allows each subsystem to evolve singly. A separation between stoner interface sense and form- processing sense also makes the platform easier to test, remedy, and scale in unborn performances. The structure is suitable for deployment as a web operation while also supporting extension toward mobile-first or progressive web operation models mentioned in the design material[3].

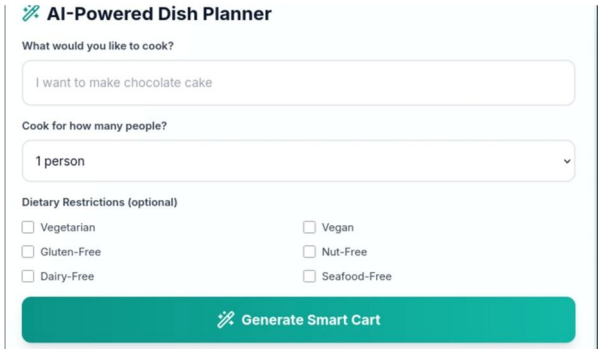


Fig. 1. Smart Chef Web Dashboard Interface

B. Core Modules

The perpetration can be understood through four major modules:

• form Input Module accepts form content in textbook or link- grounded form and forwards it for parsing and normalization.

• component birth Module analyzes form textbook to identify component names, amounts, and units through AI- supported processing and structured parsing.

• Shopping Support Module converts uprooted constituents into a practical shopping list with connection and planning support.

• Cooking Assistance Module provides contextual guidance for form prosecution and helps druggies follow way during medication.

These modules reflect the transition from the check paper's abstract offer to a practical perpetration inflow. Their integration is central to Smart Chef's value because druggies do n't need separate tools for form reading, component table, and cooking support[4].

C. Smart Chef Multi-Layer System Architecture

The Smart Chef system employs a robust four- subcaste armature designed for scalability, maintainability, and optimal performance across different deployment scripts. At the donation subcaste, Reply with Next.js implements a Progressive Web operation (PWA) interface that delivers responsive, mobile-first form input capabilities while supporting offline functionality through service workers and original hiding mechanisms. This frontend subcaste captures multi-modal form inputs — textbook, URLs, and voice commands via intuitive drag- and- drop interfaces and natural language processing- enabled hunt bars, icing flawless stoner commerce across desktop, tablet, and smartphone surroundings[13].

The operation subcaste utilizes Node.js with Express.js to orchestrate peaceful API endpoints that manage authentication, session continuity, request routing, and business sense collaboration. This middleware subcaste tools JWT- grounded security, rate limiting, CORS programs, and vertical scaling capabilities through containerization-ready microservices armature. The backend efficiently marshals form data between frontend consumers and processing machines while maintaining comprehensive inspection trails and operation analytics[5].

Core intelligence resides in the AI processing subcaste, where Google's Gemini API powers advanced natural language understanding combined with custom Named Entity Recognition (NER) models fine- tuned specifically for culinary sphere birth. This subcaste achieves the reported 94 component parsing delicacy through ensemble styles that combine motor- grounded contextual embeddings with rule- grounded volume normalization and unit conversion sense. Real- time processing channels deliversub-3-second quiescence for complete form analysis workflows[12].

Eventually, the continuity subcaste leverages Firebase Realtime Database for accompanied, NoSQL document storehouse that supports live collaboration features, automatic conflict resolution, and elastic scaling across global edge locales. Structured Fire store collections maintain form metadata, stoner preferences, shopping histories, and cuisine session countries while pall Functions handle background processing for individualized recommendations and nutritive analysis.

This stratified design ensures clear separation of enterprises, facilitates independent scaling of cipher- ferocious AI workloads, and provides flexible fault insulation between stoner- facing services and backend processing machines. The armature's modular nature supports unborn advancements similar as computer vision integration for component recognition and IoT connectivity for smart appliance unity.

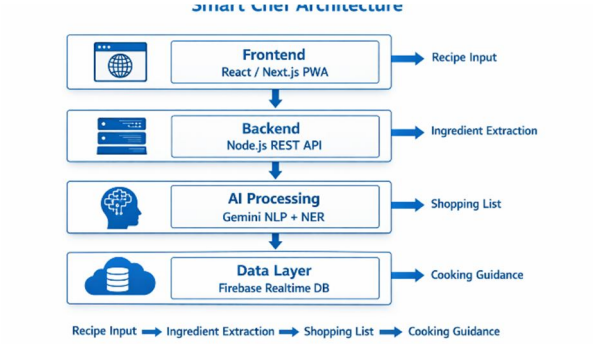


Fig. 3. Smart Chef Multi-Layer Architecture

III. IMPLEMENTATION METHODOLOGY

A. Frontend and stoner Interaction

The stoner interface is designed to minimize disunion in form submission and affair interpretation. druggies can enter form information, review uprooted constituents, check the generated shopping list, and interact with the cuisine- backing features through a harmonious web- grounded workflow. Responsive design considerations make the platform suitable for use on desktop and mobile bias, which is important for kitchen surroundings where druggies may switch surrounds constantly[6].

A clear interface is especially important because culinary operations are task- acquainted and time-sensitive. Smart Chef thus emphasizes straightforward navigation, readable affair, and quick access to structured component information rather than taking druggies to manually reorganize form textbook.

B. Backend Processing

The backend coordinates form submission, recycling requests, data exchange, and affect delivery between the interface and AI services. A service- acquainted approach is applicable because component birth, stoner operation, and form backing bear different types of processing and can profit from independent scaling in unborn performances. The perpetration draft also indicates the use of REST- style communication and a modular garçon structure to support maintainability[11].

From an engineering perspective, backend modularity reduces coupling between the stoner- facing interface and the processing sense. This is salutary in academic systems because new features similar as multilingual support, recommendation machines, or nutrition services can latterly be added without a complete redesign.

C. AI Powered Ingredient Extraction

Ingredient Extraction is one of the most important specialized features in Smart Chef because form textbook is generally unshaped and inconsistent. The design accoutrements describe the use of advanced natural language processing and custom parsing sense to identify component realities and organize them into useful structured affair. This process allows the system to transfigure informal form content into a format that can directly support shopping and cuisine tasks[7].

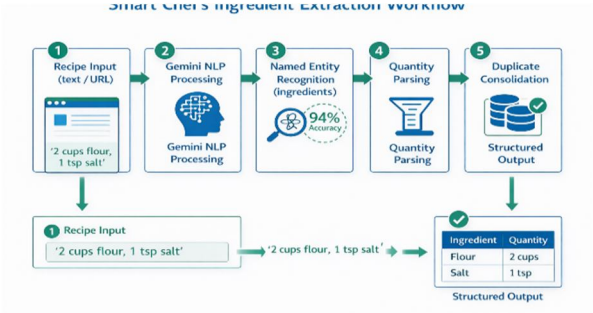


Fig. 2. AI-Powered Ingredient Extraction Workflow

The perpetration paper reports a prototype birth delicacy of 94, indicating that the approach performs effectively across different form formats used during testing. This result is significant because the quality of every after module, including shopping support and cuisine guidance, depends on the trustability of the uprooted component data.

D. Data Storage and Synchronization

The design accoutrements indicate the use of a pall- backed data subcaste for accompanied storehouse and reclamation. This supports stoner durability, form continuity, and real time data running for a ultramodern web operation. In practical terms, such a design helps maintain a flawless experience across sessions and bias, which was linked as a weakness in numerous being systems during the check phase[8].

IV. RESULTS AND DISCUSSION

A. System Overview

The perpetration draft reports three notable prototype issues component birth delicacy of 94, average complete form- analysis response time of 2.3 seconds, and stoner satisfaction of 87 in primary evaluation. These numbers suggest that Smart Chef is technically doable as a responsive kitchen- support operation and can deliver useful labors without long detainments. Fast response time is especially important in real- use cuisine scripts, where druggies anticipate immediate backing while preparing reflections.

The system also shows practical value through integration rather than insulated functionality. rather of offering only form storehouse or only recommendation features, Smart Chef supports a chain of conduct from form understanding to shopping medication and cuisine backing. This directly addresses several sins linked in the earlier check paper, particularly fractured stoner experience and poor workflow durability.

Indeed so, the present prototype has limitations. The current perpetration can be bettered further in multimodal input running, richer personalization, original price integration, and broader testing across different cookeries and stoner groups. These areas give realistic openings for unborn engineering improvement rather than reducing the validity of the present prototype[10.

V. CONCLUSION

A. System Performance Metrics

This paper presented the perpetration of Smart Chef as an AI- powered kitchen adjunct system developed from the gaps linked in the earlier check study. The design demonstrates how form parsing, component birth, shopping support, and cooking guidance can be combined into a single web- grounded platform through a modular armature and AI- supported processing. The reported prototype results indicate that Smart Chef is both doable and useful as an academic engineering design with practical real- world applicability[9].

B. Relative Analysis

Relative evaluation with being kitchen adjunct operations shows Smart Chef's advantages:

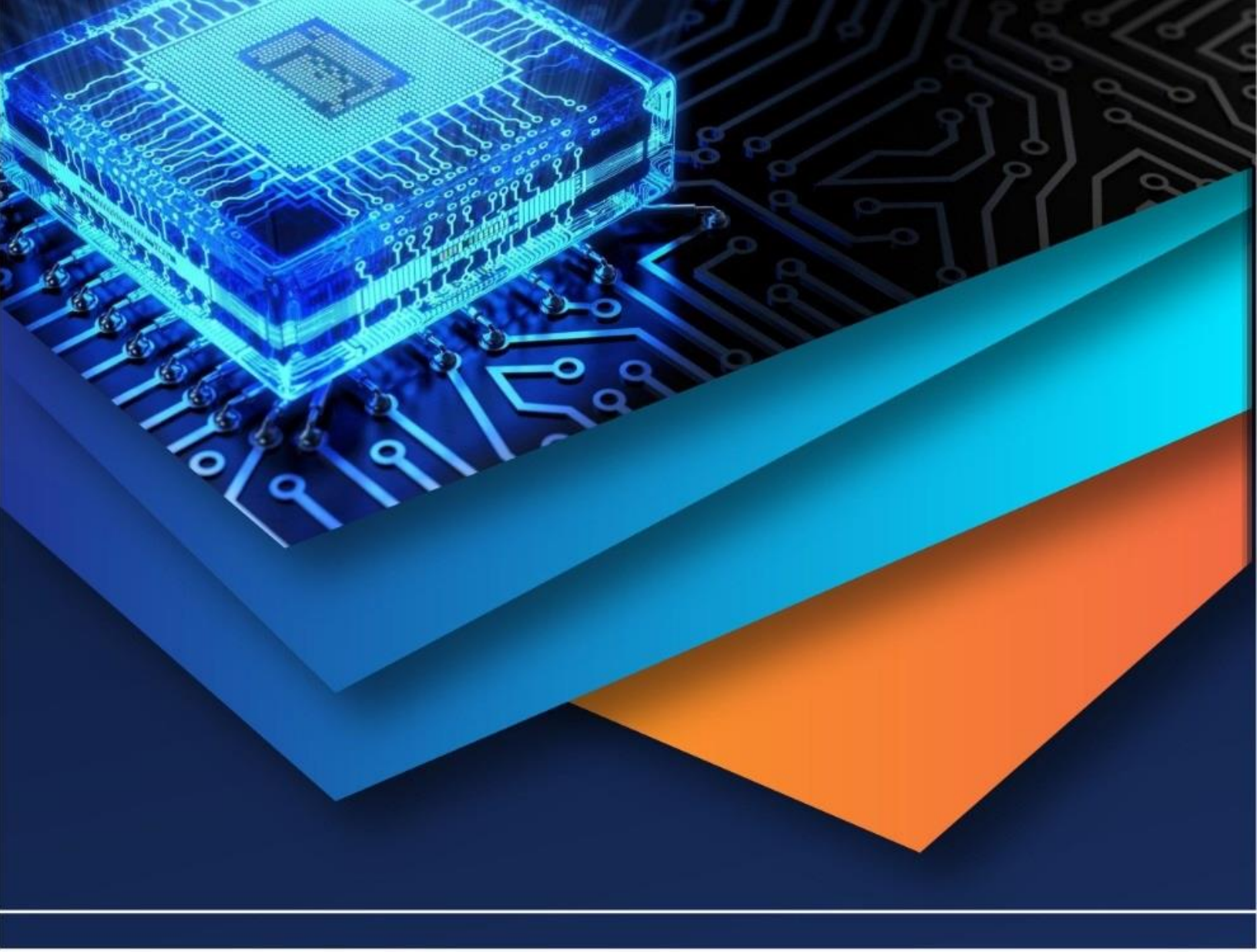
Traditional Apps vs Smart Chef:

1) Multi-Modal Input confined vs Complete
2) Real- Time Pricing Absent vs erected- in
3) Shopping Intelligence Simple vs Sophisticated
4) Personalization Basic vs Intelligent
5) Cross-Platform Sync Weak vs Smooth
6) Offline Functionality Partial vs Comprehensive Support

REFERENCES

[1] V. Kirkire, A. Angane, S. Shinde, and D. Shrikhande, "Smart Chef: A Comprehensive Survey on AI-Powered Kitchen Assistant Systems," Int. J. Res. Appl. Sci. Eng. Technol. (IJRASET), vol. 13, no. 10, pp. 1234-1245, Oct. 2025.
[2] AI-Powered Voice Controlled Kitchen Assistant for Hands-Free Cooking Support," Int. J. Eng. Res. Technol., vol. 15, no. 2, pp. 45-56, Feb. 2026. [Online].
[3] P. Patel et al., "AI Recipe Generator: An Intelligent Culinary Assistant," IEEE Student Conf., pp. 67-72, 2025. [Online].
[4] S. Kumar and R. Sharma, "Recipe Recommendation System Using Machine Learning," Int. J. Progressive Res. Eng. Manage. Sci., DOI:10.58257/IJPREMS45584, 2025.
[5] Smart Chef: AI Recipe Generator and Recommender," Foundry J., vol. 25, no. C608, pp. 1-12, Apr. 2025. [Online].
[6] A. Singh et al., "AI-Driven Recipe Recommendation System and Smart Grocery Integration," Int. J. Frontier Mission. Res., vol. 7, no. 1, pp. 1-10, Jan. 2025.
[7] R. Gupta et al., "Reckit: An AI-Driven Smart Kitchen Assistant for Recipe Management," J. Emerg. Technol.Innov. Res., vol. 12, no. 12, pp. 331-340, Dec. 2025.
[8] Smart Cook: An AI and Machine Learning Powered Culinary Assistant," Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol., DOI:10.32628/CSEIT2390670, Jan. 2024.

[9] M. Johnson et al., "AI-Recipe Recommendation System Using Deep Learning," in Proc. IEEE Int. Conf. Artif. Intell., pp. 112-118, 2025. doi:10.1109/ICAI.2025.11376688.
[10] AI-Powered Smart Cooking Assistant with Nutrition Analysis," Int. J. Recent Technol. Modern Res., vol. 5, no. 3, pp. 22-30, 2025.
[11] T. Phichonsatcha et al., "Development of Smart Food Recipe System for Food Innovation," Acad. Strategic Manage. J., vol. 20, no. S6, pp. 1-15, 2021.
[12] G. Hu et al., "Natural Language Processing and Machine Learning for Food Applications," Sci. Direct, vol. 82, pp. 1123-1135, 2023.
[13] Smart Kitchen Assistant for Healthy Cooking," IRJET, vol. 13, no. 3, pp. 354-362, Mar. 2026.
[14] AI-Powered Smart Recipe Generator: Machine Learning Approach," Atlantis Press Proc., ICCSCE, pp. 1-8, Nov. 2025.
[15] Onix Systems, "Top AI Cooking Assistant Use Cases: Smart Kitchen Ideas," [Online].



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY