IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁒ Peer-reviewed & Refereed journal ⁒ Vol. 13, Issue 4, April 2025

DOI: 10.17148/IJIREEICE.2025.13465

FOOD CALORIES ESTIMATION

Sabarish M[1], Mrs. M. Sowndharya M.Sc., M.Phil., (Ph. D.)[2]

B.Sc. Computer Science with Cognitive Systems, Dr. N. G. P. Arts and Science College, Coimbatore, India[1]

B.Sc. Computer Science with Cognitive Systems, Dr. N. G. P Arts and Science College, Coimbatore, India[2]

Abstract: The Food Calories Estimation System provides an efficient and user-friendly approach to tracking daily calorie intake using food image recognition and manual weight input. The system integrates a Flask-based web application with the Gemini API to identify food items from uploaded images and estimate their caloric value per 100g. Users can register, log in, set target calorie goals, and track their remaining calorie intake dynamically. The system calculates Total Daily Energy Expenditure (TDEE) based on user inputs such as weight, height, and activity level. The real-time calorie tracking module updates the consumed and remaining calories after each meal entry. The web application features a responsive frontend using HTML, CSS, and JavaScript, ensuring an interactive user experience. This solution provides a practical tool for individuals to monitor and manage their dietary intake effectively, promoting healthier eating habits**.**

Keywords: Food calorie estimation, Flask web application, real-time calorie tracking, food image recognition, nutrition monitoring, AI-based food identification.

I. INTRODUCTION

In today's fast-paced world, maintaining a balanced diet and tracking daily calorie intake is essential for achieving personal health and fitness goals. Many individuals struggle with estimating their calorie consumption accurately, leading to challenges in weight management and overall well-being. To address this issue, we present an AI-powered Food Calories Estimation System, which simplifies calorie tracking using image-based food recognition and real-time caloric analysis. This system allows users to upload images of their meals, leveraging the Gemini API to identify food items and estimate their calorie content per 100g. By integrating Flask for backend development and a user-friendly web interface using HTML, CSS, and JavaScript, the application provides seamless calorie tracking. Additionally, users can manually input their target daily calorie intake and receive dynamic updates on consumed and remaining calories, enabling better dietary management. The methodology incorporates user authentication, caloric needs calculation using BMR formula, and real-time calorie tracking based on image processing. Unlike traditional calorie-tracking applications that rely on manual entry, our system automates food recognition and calorie estimation, reducing effort while enhancing accuracy. This paper discusses the system's architecture, implementation, and evaluation, highlighting its efficiency in calorie estimation, ease of use, and real-time tracking capabilities. While the system effectively assists users in monitoring their intake, potential limitations include dependency on API accuracy and manual weight input. Future improvements could involve automated portion estimation and historical tracking for enhanced functionality.

II. LITERATURE REVIEW

The development of food calorie estimation systems has been significantly influenced by advancements in artificial intelligence (AI), deep learning, and food recognition technologies. Various studies highlight the role of AI in food quality assessment, calorie estimation, and dietary tracking, which serve as the foundation for this project.

Khan et al. (2022) [1] discuss AI applications in agriculture and food quality improvement, emphasizing how machine learning models can be leveraged for food classification and nutritional analysis. Similarly, Huang & Rust (2018) [2] explore AI-driven services, demonstrating its transformative impact on industries, including food and healthcare.

Kalivaraprasad et al. (2021) [3] focus on food recognition and calorie estimation using AI, showcasing deep learning methods for identifying food items from images and estimating their caloric content. This aligns with the Gemini API-based approach used in this project. Emlen (1966) [4] provides a theoretical foundation by analyzing how time and energy influence food preference, which can be integrated into AI-based dietary recommendation systems.

The impact of digital platforms on food choices is discussed by Freeman et al. (2014) [5], who analyze food marketing on social media, highlighting the growing influence of digital tools in dietary habits. Shi (2011) [6] explores advanced AI techniques, offering insights into deep learning methods applicable to food identification tasks.

Molnar (2020) [7] emphasizes the importance of interpretable machine learning, which is crucial for making AI-based calorie estimation systems transparent and user-friendly. Additionally, Hodnett & Wiley (2018) [8] provide a step-by-step guide to deep learning models, which are fundamental to training and deploying AI-powered food recognition systems.

In summary, the literature review highlights the integration of AI, machine learning, and deep learning in food recognition and calorie estimation. The Gemini API-based approach used in this project aligns with existing research, providing an efficient, AI-driven solution for personalized nutrition tracking. Future work can explore enhancing accuracy with deep learning models and improving interpretability to make AI-based calorie tracking more reliable and accessible.

III. METHODOLOGY

A. User Authentication and Profile Setup

User Registration

Users must register by providing their email ID, username, and password through the web interface.
The system securely stores user credentials, allowing them to log in on future visits.
Passwords are hashed for security, preventing unauthorized access.

User Login

Users enter their credentials to access the system.
The session remains active until the user manually logs out, ensuring seamless interaction.

B. Food Image Processing and Calorie Estimation

Image Upload

Users upload an **image of their food** through the web interface.
The image is sent to the Gemini API, which processes the food item and returns the predicted name and caloric value per 100g.

Food Weight Input

Since calorie estimation depends on portion size, users manually enter the weight of the food (in grams).

Real-Time Calorie Display

The system instantly displays the calculated calorie value based on the entered weight.
Users receive feedback on how much they have consumed from their daily target.

C. Target Calorie Tracking

Setting the Target Calories

Users manually input their target calorie intake based on their dietary goals (weight loss, maintenance, or gain).
This value is used as a benchmark to track calorie consumption throughout the day.

Real-Time Consumption Tracking

After every food scan, the system updates the consumed and remaining calorie values dynamically:

For example:

If the user's target is 2000 kcal and they consume 400 kcal, the system shows 1600 kcal remaining.

D. Web Application Development

Backend Development (Flask + Python)

The backend is developed using Flask, handling user authentication, API requests, and calorie calculations.
The system includes routes for login, registration, calorie tracking, and image upload.
The Gemini API is integrated into the backend to process food images and fetch calorie information.

Frontend Development (HTML, CSS, JavaScript)

The user interface is designed using **HTML, CSS,** and JavaScript, providing an interactive experience.
Features include:

i. A login/register page
ii. A dashboard to enter body metrics and target calories
iii. An image upload section for food calorie estimation
iv. A real-time calorie consumption display

E. System Workflow

Step 1: User Authentication

User logs in/registers using their email and password.

DOI: 10.17148/IJIREEICE.2025.13465

Profile Setup
User enters body weight, height, and activity level to calculate their caloric needs.
The system provides recommended calorie intake for maintenance, gain, or loss.
Setting Target Calories
User **sets their desired daily calorie target** based on fitness goals.
Food Image Upload
User uploads an image of their meal.
The image is sent to the Gemini API, which predicts the food item and its caloric value per 100g.
Food Weight Input & Calorie Calculation
User enters the weight of the food item in grams.
The system calculates and displays the calorie intake.
Tracking Remaining Calories
The system updates calories consumed and remaining calories dynamically.
The user can continue scanning meals until they reach their daily limit.

IV. ARCHITECTURE

A. Frontend (User Interface)

The frontend is developed using HTML, CSS, and JavaScript, ensuring an interactive and responsive experience.
Login & Registration Pages – Secure authentication for users.
Dashboard – Displays user profile, target calories, and calorie tracking.
Food Upload Module – Allows users to upload images of food items.
Calorie Display Section – Shows consumed and remaining calories after each food scan.

B. Backend (Flask Server)

The backend is built using Flask (Python**)** and manages the core functionalities:
User Authentication – Handles login, registration, and session management.
Calorie Calculation Module – Computes daily caloric needs and tracks consumption.
API Integration – Sends the uploaded food image to the Gemini API and retrieves food details.
Real-Time Calorie Updates – Dynamically calculates and updates calorie consumption.

C. External API (Gemini API)

The Gemini API is used for food recognition and calorie estimation. It performs the following:
Receives the uploaded food image.
Identifies the food item.
Returns the calorie content per 100g.

D. Data Flow and Processing

User Authentication
User logs in or registers through the web interface.
Flask validates credentials and creates a session.

Profile Setup
User enters height, weight, and activity level.
System calculates daily caloric needs using BMR and TDEE formulas.

Setting Target Calories
User sets a daily calorie target based on personal fitness goals.

Image Upload & API Processing
User uploads a food image.
Flask sends the image to the Gemini API.
API identifies the food and returns its caloric value per 100g.

Calorie Calculation
User enters the food weight (in grams).
The system calculates total calories consumed

Display of Results
Users can see their calories consumed and remaining calories for the day in real time.
The process repeats for every new food item uploaded.

DOI: 10.17148/IJIREEICE.2025.13465

E. Key Features of the Architecture

Modular Design: Separates frontend, backend, and API integration for maintainability.
Scalability: Can be expanded with additional features like database storage or history tracking.
Real-Time Updates: Provides instant calorie tracking without requiring persistent storage.
Security: Uses Flask session management for user authentication and protection.

V. IMPLEMENTATION

A.Development Environment & Tools

Backend: Flask (Python), Requests (for API calls)
Frontend: HTML, CSS, JavaScript, Bootstrap
API: Gemini API for food recognition and calorie estimation
Database: Stores user credentials and calorie targets (optional)

B. System Workflow

1. User Authentication: Login and register system with session management.
2. Profile Setup: Users enter height & weight to calculate calorie needs.
3. Set Target Calories**:** Users manually input a calorie goal.
4. Image Upload & API Processing: Users upload food images, which are analyzed by the Gemini API.
5. Calorie Calculation: Based on API results, the system computes total calories
6. Calorie Tracking: Displays total consumed and remaining calories dynamically.
7. Results Display: Shows food name, calories per 100g, total consumed, and remaining calories.

C. User Interface

Login/Register Pages: Secure authentication system.
Dashboard: Displays calorie goals, intake, and remaining calories.
Image Upload Page: Allows users to upload food images.
Results Page: Shows calorie details and remaining calories dynamically.

D. Testing & Deployment

Unit & Integration Testing: Ensured seamless interaction between modules.
Edge Case Handling: Tested incorrect images and missing inputs.
Deployment: Hosted using Flask

E. Key Features

Real-time calorie tracking
Secure user authentication
Gemini API-based food recognition
User-friendly, responsive web interface
Scalable architecture for future enhancements

VI. RESULT

A. Accuracy of Food Identification

The Gemini API correctly identifies food items in most cases, providing reliable calorie estimates.
Some misclassifications occur with unclear images or multiple food items in a single image.

B. Calorie Estimation Performance

The system calculates calories based on the formula
Results are displayed in real-time, allowing users to track their intake dynamically.

C. User Experience and Functionality

The login and registration **system** ensures secure user authentication.
The dashboard provides a clear summary of total calories consumed and remaining calories.
The image upload and result display are smooth and responsive.

D. Limitations & Observations

The system does not store food history, so users cannot track past meals.
The accuracy depends on the API's recognition capability, requiring clear food images.
Users must manually enter food weight, which may lead to estimation errors.

E. Summary of Results

Accurate food identification and calorie estimation using Gemini API.
Efficient calorie tracking without complex manual entry.
Simple and intuitive UI for better user experience.
Real-time updates on remaining calories based on the user’s goal.

VII. DISCUSSION

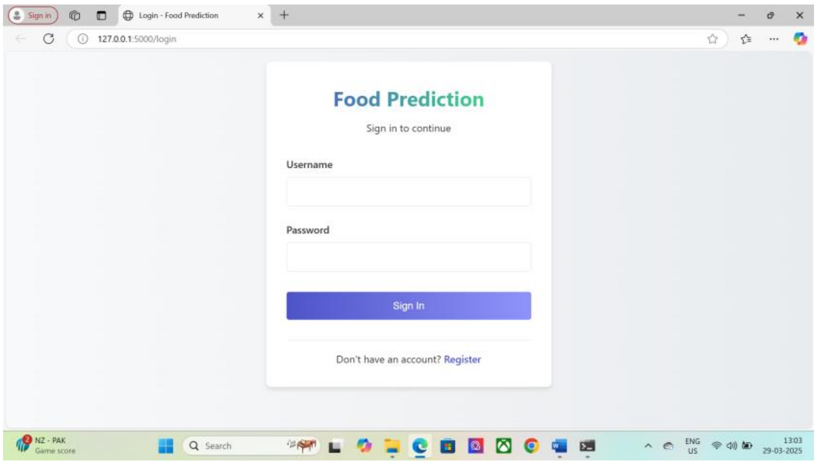
A. Login Module

Purpose:
The login module ensures secure access to the system, allowing users to track their calorie intake in a personalized manner.

Functionality:
Users enter their email and password to authenticate.
The system validates credentials against stored user data.
If authentication succeeds, the user is redirected to the dashboard.
If the login fails, an error message is displayed, prompting re-entry of credentials.

Security Measures:
Password encryption to protect user credentials.
Session management to maintain logged-in state.

User Experience:
A simple and user-friendly UI with fields for email and password.
A “Forgot Password” option can be added for account recovery.



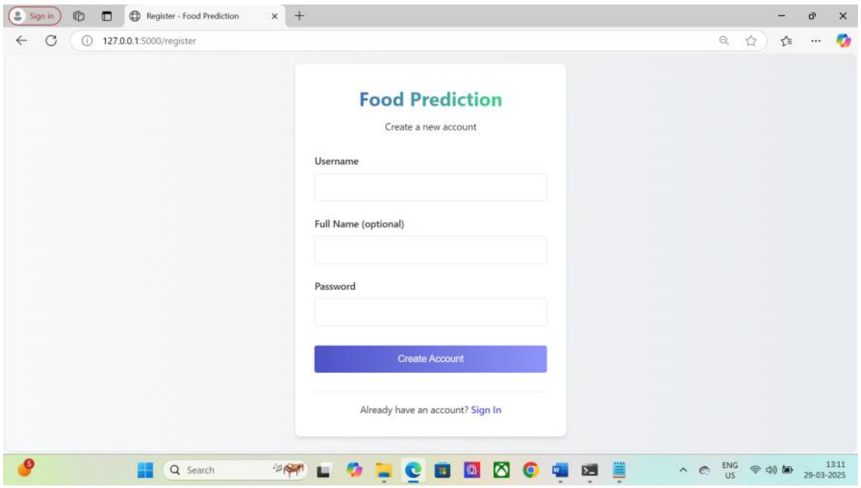
B. Register Module

Purpose:
The register module allows new users to create an account, enabling them to access the calorie tracking system.

Functionality:
Users enter their name, email, and password to register.
The system checks if the email already exists to prevent duplicate accounts.
Once registered, users are redirected to the login page to access their profile.

Data Validation:
Ensures password strength for security.
Prevents duplicate accounts by validating email uniqueness.

User Experience:
Simple and intuitive form-based registration.
A confirmation message after successful registration.



C. Calories Set, Image Upload, and Weight Entry Module

Purpose:

This module enables users to **set their target calories**, upload an image of their food, and enter the food weight for calorie calculation.

Functionality:

Set Target Calories:

Users input their target calorie intake for the day.
This value is used to calculate the remaining calories after each food entry.

Image Upload:

Users upload an image of the food they are about to consume.
The image is sent to the Gemini API for food identification and calorie estimation.

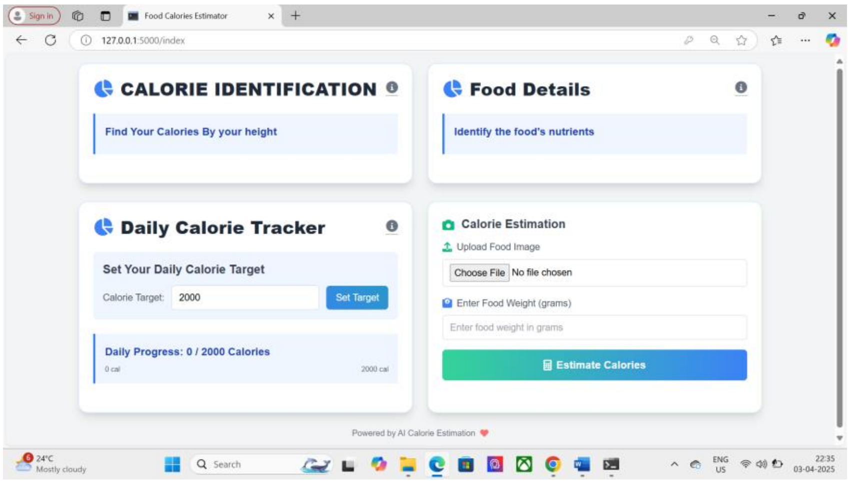
Food Weight Entry:

Users manually enter the weight of the food (in grams).
The system calculates calories based on weight

User Experience:

Simple interface for uploading images and entering food weight.

Real-time calorie calculation after input submission.



Calorie Target: 2000

No file chosen

Daily Progress: 0 / 2000 Calories

D. Calorie Identification by Height and Weight Module

Purpose:

• This module calculates a user's daily calorie needs based on their height and weight, providing personalized calorie recommendations.

DOI: 10.17148/IJIREEICE.2025.13465

Functionality:

1. **User Data Input:**

Users enter their height (in cm) and weight (in kg).
The system calculates their Basal Metabolic Rate (BMR) using standard formulas.

2. **Calorie Requirement Calculation:**

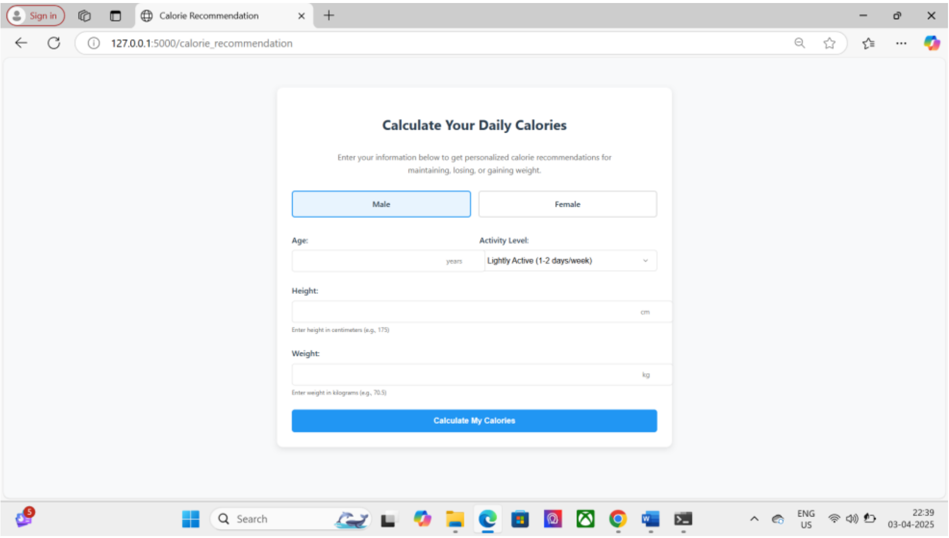
Based on the BMR, the system determines daily calorie needs for maintenance, weight loss, and weight gain.
Users receive recommendations on how many calories they should consume daily.

3. **Integration with Calorie Tracking:**

The calculated calorie requirement is used to track food intake.
After each meal entry, the system updates the remaining calorie count for the day.

User Experience:

Simple form-based interface for entering height and weight.
Instant calculation and display of daily calorie needs.
Dynamic tracking of calorie consumption based on user input.



Calculate Your Daily Calories

Enter your information below to get personalized calorie recommendations for maintaining, losing, or gaining weight.

Male | Female

Age: years

Activity Level: Lightly Active (1-2 days/week)

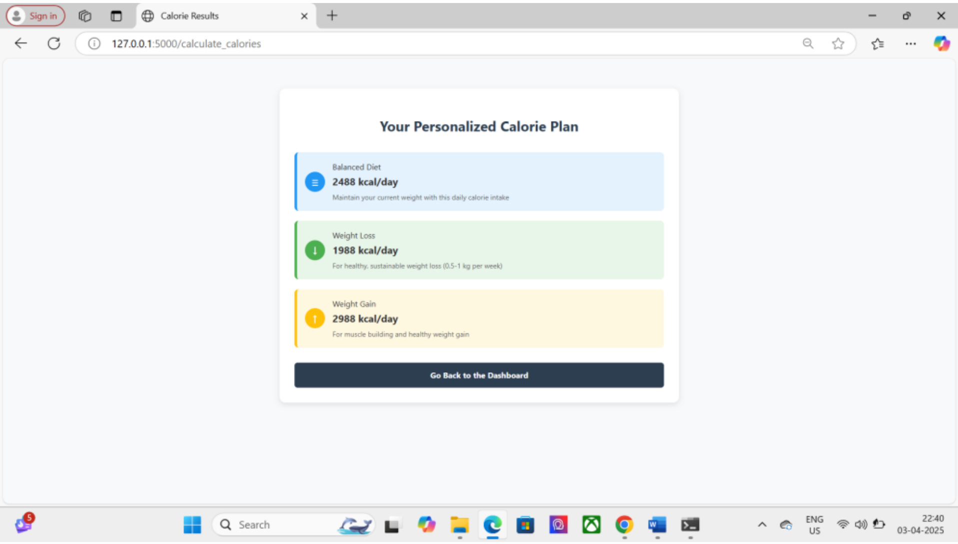
Height: cm

Enter height in centimeters (e.g., 175)

Weight: kg

Enter weight in kilograms (e.g., 70.5)

Calculate My Calories



Your Personalized Calorie Plan

Balanced Diet
2488 kcal/day
Maintain your current weight with this daily calorie intake

Weight Loss
1988 kcal/day
For healthy, sustainable weight loss (0.5-1 kg per week)

Weight Gain
2988 kcal/day
For muscle building and healthy weight gain

Go Back to the Dashboard

E. Food Details Module

Purpose:

This module provides detailed nutritional information about the identified food item, helping users make informed dietary choices.

Functionality:

1. **Food Identification:**

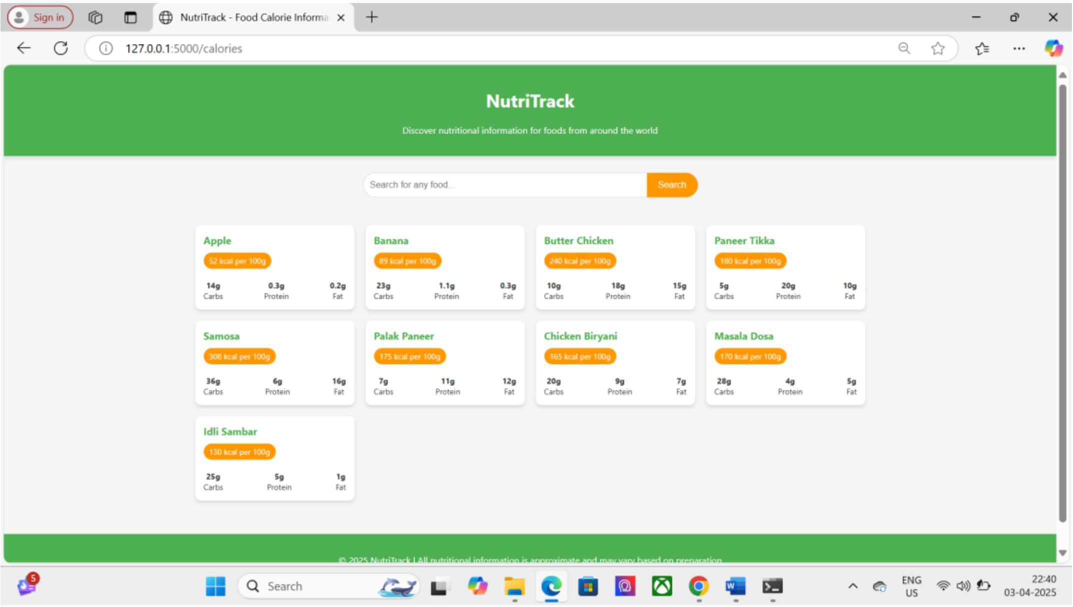
The uploaded image is processed using the Gemini API.
The API returns the food name along with estimated calorie values.

2. **Nutritional Information Display:**

The system retrieves key nutritional data, including:
Food Name: Name of the identified food item.
Calories: Estimated calorie content per 100g.
Protein: Amount of protein present in the food.
Carbohydrates: Carbohydrate content per serving.
Fats: Fat content per serving.

User Interaction:
Users can view the nutritional breakdown before confirming food intake.
The system updates their calorie tracking based on the selected food and entered weight.

User Experience:
Clear and detailed nutritional breakdown of food items.
Real-time feedback on calorie and macronutrient intake.
Helps users track and manage their diet effectively.



NutriTrack

Discover nutritional information for foods from around the world

Apple
52 kcal per 100g
14g Carbs | 0.3g Protein | 0.2g Fat

Banana
89 kcal per 100g
23g Carbs | 1.1g Protein | 0.3g Fat

Butter Chicken
240 kcal per 100g
10g Carbs | 18g Protein | 15g Fat

Paneer Tikka
180 kcal per 100g
5g Carbs | 20g Protein | 10g Fat

Samosa
308 kcal per 100g
36g Carbs | 6g Protein | 16g Fat

Palak Paneer
175 kcal per 100g
7g Carbs | 11g Protein | 12g Fat

Chicken Biryani
165 kcal per 100g
20g Carbs | 9g Protein | 7g Fat

Masala Dosa
170 kcal per 100g
28g Carbs | 4g Protein | 5g Fat

Idli Sambar
130 kcal per 100g
25g Carbs | 5g Protein | 1g Fat

© 2025 NutriTrack | All nutritional information is approximate and may vary based on preparation

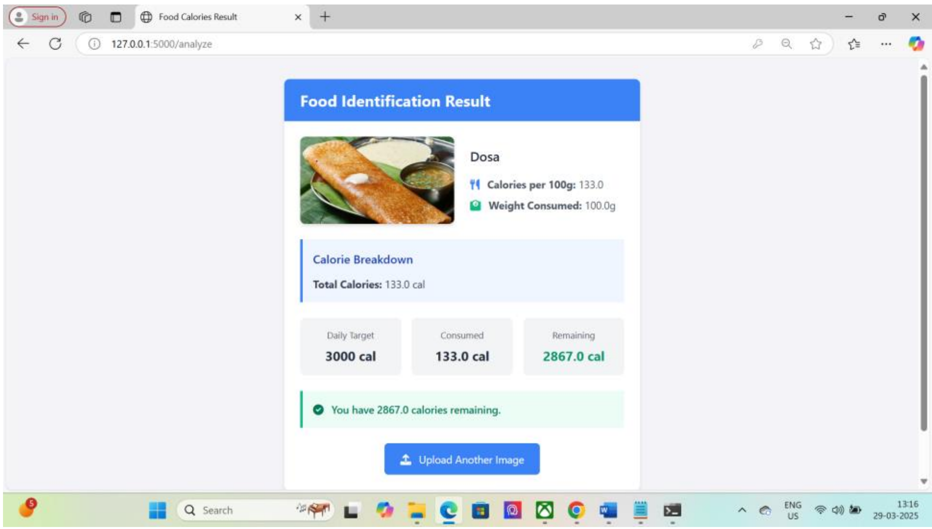
F. Result and Remaining Calories Module

Purpose:
Displays the food recognition result, estimated calories, and remaining calories after consumption.

Functionality:
The system retrieves food details from the Gemini API and displays:
Food Name (identified by the API).
Calories per 100g (fetched from API data).
Total Calories Consumed (calculated using food weight).
Remaining Calories (based on user's target).
The updated remaining calorie count helps users track their intake.

User Experience:
Clear and visually organized results.
Users can adjust weight inputs to refine calorie estimates.

DOI: 10.17148/IJIREEICE.2025.13465



Food Identification Result

Dosa

Calories per 100g: 133.0

Weight Consumed: 100.0g

Calorie Breakdown

Total Calories: 133.0 cal

Daily Target	Consumed	Remaining
3000 cal	133.0 cal	2867.0 cal

You have 2867.0 calories remaining.

Upload Another Image

VIII. CONCLUSION

The Food Calories Estimation System successfully provides users with an efficient and user-friendly way to track their daily calorie intake. By integrating Flask, Gemini API, and a responsive web interface, the system enables users to upload food images, estimate calorie content, and track remaining calories based on their dietary goals. The implementation of BMR calculations ensures that users receive personalized recommendations for their caloric needs. The system offers real-time calorie tracking, allowing users to monitor their intake dynamically. While the Gemini API provides accurate food identification and calorie estimation, minor limitations exist, such as occasional misclassification of food items and reliance on manual weight input. Despite these challenges, the system successfully simplifies calorie tracking without requiring extensive manual data entry. Future enhancements may include automated portion size estimation, meal history tracking, and AI-driven dietary recommendations to further improve user experience and accuracy. Overall, the project demonstrates a scalable and effective approach to personalized nutrition tracking, empowering users to make informed dietary choices.

REFERENCES

[1]. Khan, M. A., Khan, R., & Praveen, P. (2022). Artificial intelligence applications in agriculture and food quality improvement. IGI Global.
[2]. Huang, M., & Rust, R. T. (2018). Artificial intelligence in service. Journal of Service Research, 21(2), 155–172. https://doi.org/10.1177/1094670517752459
[3]. Kalivaraprasad, B., Prasad, M., Vamsi, R., Tejasri, U., Santhoshi, M. N., & PramodKumar, A. (2021). Analysis of food recognition and calorie estimation using AI. AIP Conference Proceedings, 2407, 020020. https://doi.org/10.1063/5.0074506
[4]. Emlen, J. M. (1966). The role of time and energy in food preference. The American Naturalist, 100(916), 611–617. https://doi.org/10.1086/282455
[5]. Freeman, B., Kelly, B., Baur, L., Chapman, K., Chapman, S., Gill, T., & King, L. (2014). Digital junk: food and beverage marketing on Facebook. American Journal of Public Health, 104(12), e56–e64. https://doi.org/10.2105/ajph.2014.302167
[6]. Shi, Z. (2011). Advanced artificial Intelligence. World Scientific.
[7]. Molnar, C. (2020). Interpretable Machine learning. Lulu.com.
[8]. Hodnett, M., & Wiley, J. F. (2018). R Deep Learning Essentials: A step-by-step guide to building deep learning models using TensorFlow, Keras, and MXNet, 2nd Edition. Packt Publishing Ltd.