International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 11 | Nov - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Calorie Intake Tracker

Dr Baby Kalphana , Harshini A

Professor, Sri Shakthi Institute of Engineering and Technology
Student, Sri Shakthi Institute of Engineering and Technology

Abstract : The **Calorie Intake Tracker** web application represents a modern approach to personal nutrition management through the integration of web technologies and interactive data visualization. Built using Django and supported by front-end technologies like HTML, CSS, Bootstrap, and JavaScript, this system provides users with an intuitive platform to monitor their daily food consumption and track nutritional intake, including carbohydrates, proteins, fats, and total calories.

The system operates by allowing users to select food items from a predefined database, automatically calculating and displaying nutrient values in real time. It dynamically updates a visual progress bar to reflect how much of the daily calorie goal has been met and presents a nutrient breakdown using a pie chart for better insight into dietary balance. Each user's entries are stored and managed efficiently using Django's ORM, ensuring personalized tracking and future extensibility for historical data review or authentication-based access.

This application supports healthier lifestyle choices by encouraging users to be more aware of their food intake and make informed nutritional decisions. Its potential applications extend beyond individual use to include dietary planning in wellness programs, fitness centers, and healthcare settings. While the system currently uses static food data, future enhancements could include real-time API integrations, user authentication, and mobile optimization.

By bridging user-friendly interfaces with meaningful dietary insights, the Calorie Intake Tracker demonstrates how digital tools can promote wellness and support the growing demand for self-managed health applications in an increasingly health-conscious world.

Index Terms – ***Calorie Tracker, Web-based Application, Django Framework, System Architecture, Database Design, User Interface, Module Implementation, Health Monitoring, Results and Discussion, Future Enhancement.***

I. INTRODUCTION

The rise of health-conscious lifestyles and increasing awareness of dietary habits have driven the need for effective tools to monitor food intake.

Obesity, malnutrition, and lifestyle diseases are on the rise due to improper dietary patterns.

The Calorie Intake Tracker aims to provide a digital solution that assists users in tracking and managing their daily nutrition in a simplified and interactive way.

This system allows users to select food items from a predefined list, automatically calculating and displaying nutritional values such as carbohydrates, proteins, fats, and total calories. Real-time updates, visual feedback, and clear summaries ensure that users are constantly aware of their nutritional consumption and can make better choices.

1.1 Project Overview

The Calorie Intake Tracker is a web-based application built using Django and front-end tools that provides users a platform to log their daily food intake. Each selected item updates the cumulative nutritional values and displays a progress bar and pie chart for visual insights. The platform is ideal for individuals aiming to maintain or improve their health by keeping their nutritional goals in check.

Key features include:

- Selection of food items from a database.
- Automatic calculation of carbs, proteins, fats, and calories.
- Real-time summary table of daily intake.
- Progress bar showing percentage of daily calorie goal.
- Pie chart representing nutrient distribution.

1.2 Objectives

- To develop a responsive and user-friendly web application for tracking food consumption.
- To calculate and display nutritional data in real-time.
- To support users in achieving dietary goals and maintaining health.

1.3 Project Scope

This project covers the implementation of a calorie tracking system that leverages Django for the backend, HTML/CSS/Bootstrap for the frontend, and JavaScript for interactivity. It includes predefined food items and nutrient values and is suitable for individual use.

Future scope includes:

- Integration with real-time nutrition APIs.
- User login and personalized dietary plans.
- Historical data analysis and reporting.
- Mobile application extension.

1.4 Project Background

With digital wellness tools becoming increasingly popular, many individuals look for ways to track and improve their eating habits.

Traditional methods like manual food logs are often inconvenient and inaccurate. This project stems from the need for an automated, visually engaging, and reliable solution that simplifies nutrition management. The Calorie Intake Tracker is a step toward that goal, making nutrition tracking accessible to everyone through a simple browser-based interface.

CHAPTER 2: LITERATURE REVIEW

This chapter presents an overview of existing systems and research work related to food intake tracking, dietary monitoring, and health management tools. The goal is to identify existing methodologies, their strengths and limitations, and how the proposed Calorie Intake Tracker improves upon them.

2.1 Review of Existing Nutrition Tracking Systems

Several mobile and web-based applications currently allow users to track calorie and nutrient intake. Notable examples include:

- MyFitnessPal: A comprehensive app that supports barcode scanning, extensive food databases, and integration with fitness trackers.
- HealthifyMe: Popular in India, offering personalized meal plans, fitness tracking, and nutritionist support.
- Lose It!: Focuses on calorie budgeting with intuitive visualizations and progress charts.

However, many of these tools require extensive configuration, suffer from UI complexity, or offer limited food customization.

2.2 Academic Research on Dietary Monitoring

- Research has shown that consistent logging of food intake significantly aids in weight loss and nutrition awareness.
- Machine learning models are increasingly being used to predict calorie content based on image recognition or food database lookups.

2.3 Limitations in Existing Systems

- Complex User Interfaces: Steep learning curves for new users.
- Over-dependence on Mobile Apps: Lack of accessible web-based platforms.
- Limited Data Visualization: Inadequate use of visual tools like charts and graphs to make nutrient distribution clear.

2.4 Motivation for the Proposed System

This project addresses the above limitations by building a simple, web- based calorie tracker that emphasizes real-time feedback, ease of use, and clear data visualization using modern web technologies like Django, Bootstrap, and Chart.js.

The proposed system does not attempt to replace advanced trackers but focuses on user-centric design, simplified interactions, and foundational dietary management tools. This makes it suitable for both beginners and casual health-conscious users.

CHAPTER 3: SYSTEM DESIGN AND ARCHITECTURE

The Calorie Intake Tracker is structured using a clear and efficient architecture that separates concerns across various modules including models, views, templates, and static files. The system is designed based on Django's MVT (Model-View-Template) pattern.

3.1 Key Features

- **Food Selection**: Users select food items from a dropdown populated with pre-defined items.
- **Nutrition Calculation**: Upon selection, nutrients such as carbs, proteins, fats, and calories are displayed.
- **Real-Time Feedback**: The application updates a progress bar and pie chart immediately as data is added.
- **Delete Option**: Users can remove previously added food entries.

3.2 System Flow and Architecture

- The frontend interface collects user input through HTML forms.
- Views in Django handle HTTP requests and manage logic like adding/removing consumed food.
- Models define the data schema for storing food and consumption data.
- Templates render the final web page using dynamic data from views.

3.3 Database Design

- **Food Model**: Stores food name, carbohydrates, proteins, fats, and calories.
- **Consume Model**: Links the user to the food items they have consumed.

CHAPTER 4: TECHNOLOGIES AND TOOLS USED

Technology	Description
Django	Python-based web framework used for backend development and ORM
Python	Core programming language for business logic
HTML/CSS	Used for structuring and styling the web interface
Bootstrap	Framework for responsive design and prebuilt UI components
JavaScript	Enables interactivity such as dynamic chart updates
Chart.js	JavaScript library used to create the pie chart for nutrient distribution
SQLite	Lightweight database used in development stage

CHAPTER 5: IMPLEMENTATION AND MODULES

5.1 Module Descriptions

- **User Input Module**: Accepts food selection via a dropdown.
- **Computation Module**: Aggregates nutrient totals and calculates calorie consumption.
- **Visualization Module**: Updates chart and progress bar.
- **Data Management Module**: Interacts with Django ORM for data storage and retrieval.

5.2 Sample Code Snippet (Views)

```
from django.shortcuts import render, redirect from .models import Food, Consume

def index(request):
```

```
if request.method == "POST":
food_consumed = request.POST['food_consumed'] consume = Food.objects.get(name=food_consumed) user = request.user
consume = Consume(user=user, food_consumed=consume) consume.save()
foods = Food.objects.all() else:
foods = Food.objects.all()
consumed_food = Consume.objects.filter(user=request.user)
return render(request, 'myapp/index.html', { 'foods': foods,
    'consumed_food': consumed_food
})
def delete_consume(request, id): consumed_food = Consume.objects.get(id=id) if request.method == 'POST':
consumed_food.delete() return redirect('/')
return render(request, 'myapp/delete.html')
```

5.3 Model Code (Django Models)

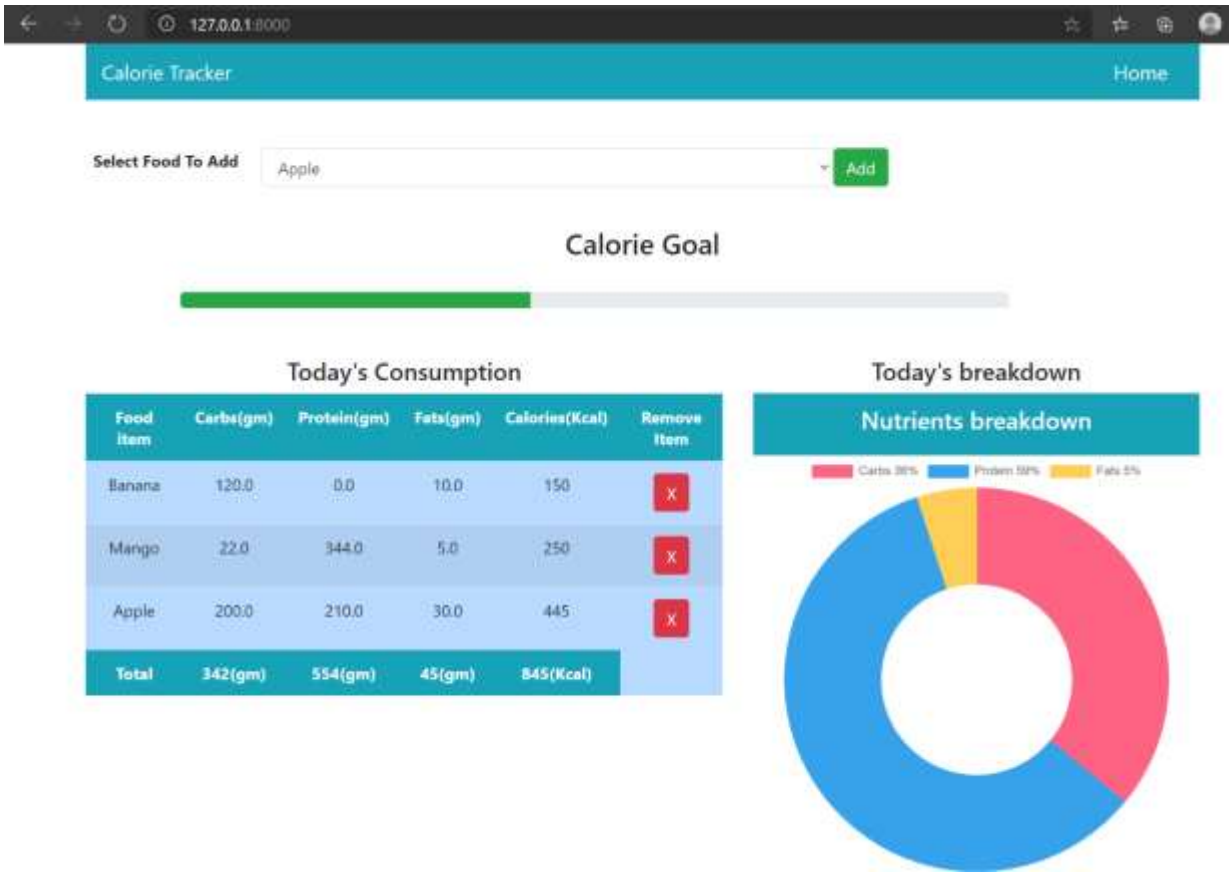
```
from django.db import models
from django.contrib.auth.models import User
class Food(models.Model):
name = models.CharField(max_length=100) carbs = models.FloatField()
protein = models.FloatField() fats = models.FloatField() calories = models.IntegerField()
def ___str____(self): return self.name

class Consume(models.Model):
user = models.ForeignKey(User, on_delete=models.CASCADE) food_consumed = models.ForeignKey(Food,
on_delete=models.CASCADE)
```

CHAPTER 6: RESULTS AND DISCUSSION

The Calorie Intake Tracker application was successfully implemented and tested. It enables users to track their food intake, view nutrient summaries, and analyze their calorie goals in real time.

- The dropdown allows food selection with quick feedback.
- The summary table dynamically updates with each addition.
- The calorie goal progress bar visually represents goal achievement.
- The pie chart clearly shows the distribution of nutrients.
- Users can delete entries, allowing flexible adjustments.

Screenshot: Calorie Tracker Dashboard



This image demonstrates the layout of the dashboard including food entries, nutritional values, and visual indicators.

CHAPTER 7: CONCLUSION AND FUTURE WORK

Conclusion:

The Calorie Intake Tracker successfully delivers an intuitive and interactive web-based solution for monitoring dietary intake. By leveraging Django's powerful backend capabilities and incorporating dynamic front-end technologies like Bootstrap and Chart.js, the application provides a reliable and user-friendly platform for real-time nutrition management. Users can track carbohydrates, proteins, fats, and calories effectively and gain meaningful insights through data visualization tools such as progress bars and pie charts.

This project demonstrates how modern web technologies can be combined to solve real-life health and wellness challenges. It encourages users to adopt healthier eating habits by offering immediate feedback and clarity about their food consumption patterns.

Future Work:

While the current system offers robust features for personal calorie tracking, future enhancements could make it even more powerful and versatile. Potential improvements include:

- Implementation of a user authentication system for storing individual histories.
- Integration with nutrition databases through APIs for broader food data.
- Support for user-defined goals and recommendations based on dietary preferences.

- AI-driven food recognition using image uploads.
- Exportable reports and analytical tools for nutritionists and healthcare professionals.

CHAPTER 8: REFERENCES

1. Achananuparp, Palakorn, and Ee-Peng Lim. "Online Food Diary: A Tool for Studying Personal Food Consumption." *ACM Web Science Conference*, 2011.

2. Illner, A. K., Freisling, H., Boeing, H., Huybrechts, I., Crispim, S. P., & Slimani, N. (2012). "Review and evaluation of innovative technologies for measuring diet in nutritional epidemiology." *International Journal of Epidemiology*, 41(4), 1187–1203.

3. König, L. M., et al. (2018). "Describing the participant experience in app-based dietary assessment studies." *JMIR mHealth and uHealth*, 6(5), e121.

4. Lu, Y., et al. (2020). "Food calorie estimation using deep learning and computer vision." *Pattern Recognition Letters*, 131, 223–231.

5. Sun, Mingui, et al. "Eating Activity Recognition for Health Monitoring." *IEEE Pervasive Computing*, vol. 13, no. 3, 2014, pp. 42–49.