



# SHRIDEVI HACKATHON 2024

## TITLE PAGE

- **Theme - Healthcare**
- **Team Members Name –**

**A Pradhyumna, H M Ayush Kumar, K Akhil Kumar, Prajwal Subramanya S N**

- **College Name – Siddaganga Institute Technology**



M61

# PROBLEM STATEMENT



- Individuals with chronic diseases often struggle to find dietary guidance tailored to their specific health conditions and personal preferences.
- Current solutions lack personalization, making it challenging for users to adhere to recommended nutritional plans that could aid in managing symptoms effectively.

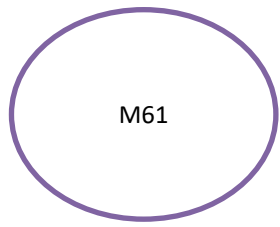


M61

# PROBLEM STATEMENT



- **Objective:** Develop an AI-powered nutritional advisor to support individuals with chronic diseases (e.g., diabetes, hypertension) in managing their conditions through tailored dietary recommendations.
- **Personalized Recommendations:** Develop an AI model to provide dietary suggestions based on individual health conditions, preferences, and lifestyle, ensuring recommendations are easy to follow and sustainable.
- **Data-Driven Insights:** Use machine learning algorithms to analyze large datasets of nutritional information and health outcomes, offering scientifically-backed, condition-specific dietary guidance.



# TECHNICAL APPROACH

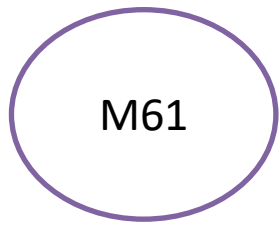


## 1. Data Handling & Model Training

- Libraries: Pandas, Scikit-learn (train\_test\_split, RandomForestClassifier), Joblib.
- Preprocessing: Converts preferences and health\_conditions to numeric codes.
- Model: Trained RandomForestClassifier to predict suggestions, saved as nutrition\_model.pkl.

## 2. Database Management

- Database: SQLite.
- Functions:
  - init\_db(): Initializes database and creates suggestion table.
  - insert\_user(preferences, health\_conditions): Inserts user data.
  - insert\_multiple\_users(num\_users): Adds random user data.



# TECHNICAL APPROACH



## 3. API Development

- Framework: FastAPI.
- Endpoint: /suggestions:
- Accepts POST requests for user preferences and health\_conditions, returns nutrition suggestions.
- Input Model: UserInput Class for data validation.

## 4. Frontend Development

- Technologies: HTML, CSS, JavaScript.

## 5. Server Execution

- Command: FastAPI app runs on Uvicorn at localhost:8000

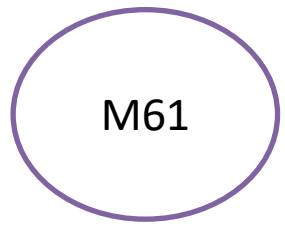


M61

# TECHNICAL APPROACH



- **Data Collection and Preprocessing:**
  - Gather data on dietary preferences, chronic disease conditions, and relevant nutritional information.
  - Preprocess the data by encoding categorical variables (e.g., preferences and health conditions) into numerical formats suitable for model training.
- **Model Training:**
  - Use a machine learning model, such as a RandomForestClassifier, to predict personalized dietary suggestions based on user inputs.
  - Split the dataset into training and testing sets to evaluate the model's performance.



# TECHNICAL APPROACH

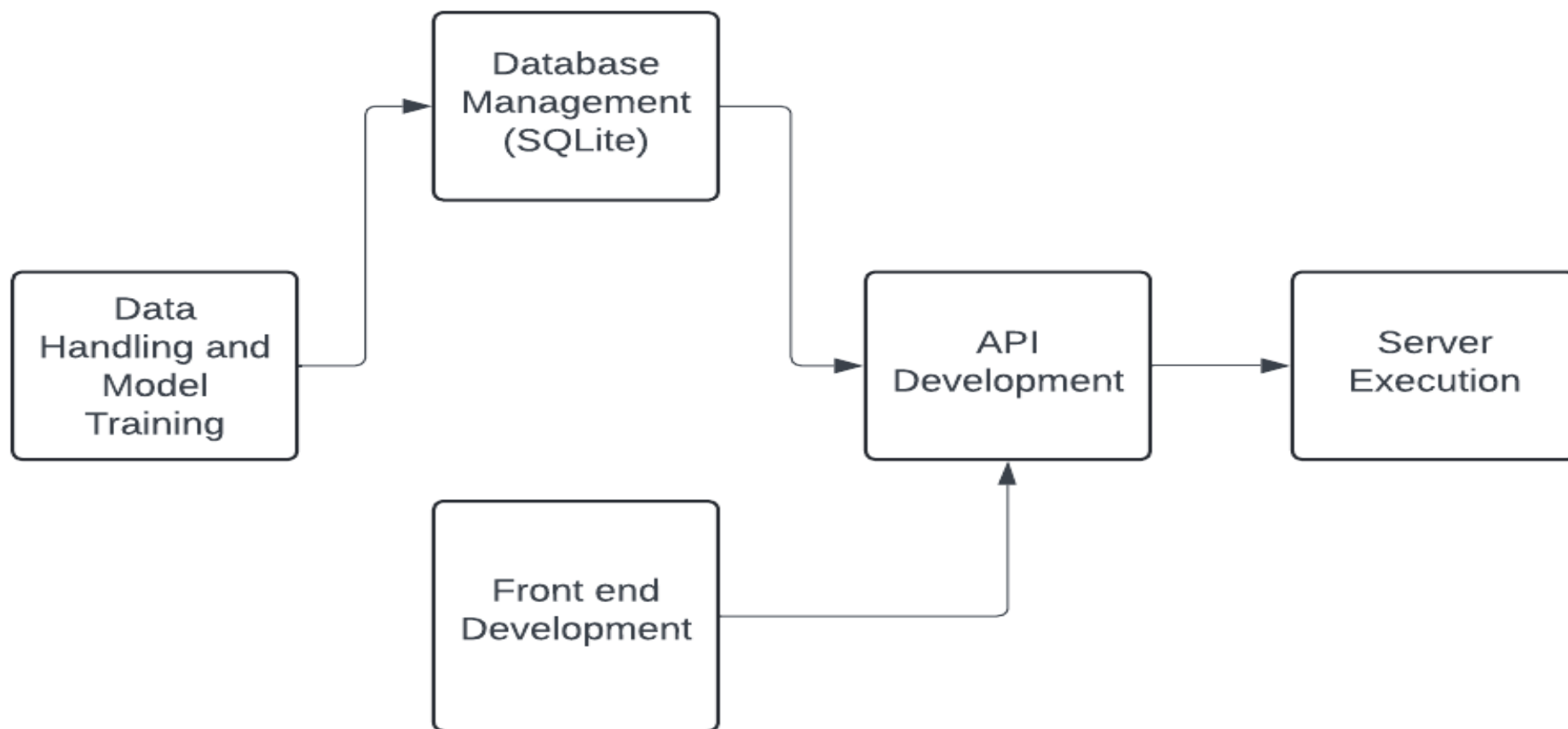


- **Database Integration:**
  - Initialize an SQLite database to store user dietary preferences and health conditions.
  - Populate the database with initial sample data and set up methods for adding new user data.
- **Backend API Development:**
  - Build a backend API using FastAPI to handle user requests and serve dietary recommendations.
  - Set up endpoints to process user input, make model predictions, and return results to the front end.
- **Frontend and User Interface:**
  - Design a user-friendly web interface for entering dietary preferences, health conditions, and meal preferences.
  - Connect the frontend to the backend API to retrieve and display personalized suggestions.

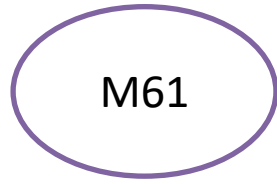


M61

# TECHNICAL APPROACH



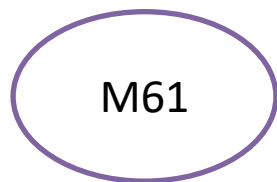




# FEATURES



- **User Profiling:**
  - Create detailed user profiles by collecting dietary preferences, health conditions, and lifestyle habits. This allows for personalized nutrition recommendations that cater to each individual's health needs and goals.
- **Dynamic Meal Planning:**
  - Offer personalized meal suggestions based on user inputs and health conditions, ensuring that dietary recommendations are not only relevant but also nutritious and aligned with user preferences.



# FEATURES



- **Health Condition Alerts:**

- Implement a feature to alert users about dietary constraints associated with their health conditions. The system can suggest safer food alternatives to minimize health risks, enhancing overall well-being.



M61

# CHALLENGES FACED



- **Challenge:** Model Evaluation and Hyperparameter Tuning
  - It is difficult to evaluate the performance of the model and tune the hyperparameters, which limits the ability to increase the accuracy of recommendations.
- **Solution:** Provide a controlled evaluation procedure using cross-validation together with appropriate performance measures (precision, recall, etc.) for the optimization of hyperparameters.
- **Challenge:** Integration of Frontend and Backend
  - Users may fail to properly integrate the backend API and the frontend, leading to communication failures between components.
- **Solution:** Proper API documentation along with examples of firing off asynchronous requests in the frontend and then parsing the responses.



M61

# CHALLENGES FACED



- **Challenge:** Algorithm Bias
  - Machine learning models may inherit biases from training data, leading to unfair or ineffective recommendations.
- **Solution:** Ensure diverse and representative training data. Regularly audit model outputs and implement fairness metrics to identify and mitigate bias in recommendations.
- **Challenge:** User Engagement and Retention:
  - Keeping users engaged with the platform over time can be difficult, especially if they do not see immediate results.
- **Solution:** Incorporate gamification elements, such as progress tracking, rewards for achieving goals, and personalized feedback. Regularly update content and features based on user feedback to maintain interest.