Software Technical Requirements Document (TRD) for cookGPT Project

1. Introduction:

1.1 Project Overview:

cookGPT is an AI-powered Indian cooking assistant designed to generate recipes, offer personalized culinary suggestions, and enhance the overall cooking experience. This document outlines the technical requirements to achieve the objectives defined in the Business Requirement Document (BRD).

1.2 Scope:

The technical requirements encompass the development, implementation, and maintenance of the cookGPT software application, including its AI model, user interface, and integration capabilities.

2. System Architecture:

2.1 AI Model:

2.1.1 Requirement:

The system shall integrate a natural language processing AI model capable of understanding and generating Indian cooking recipes.

2.1.2 Specification:

- Utilize OpenAI's GPT-4 architecture for natural language understanding and generation.
- Implement fine-tuning on domain-specific Indian cooking datasets for improved recipe generation.

2.2 User Interface:

2.2.1 Requirement:

The system shall provide an intuitive and user-friendly interface accessible from desktop and mobile devices.

2.2.2 Specification:

- Develop a responsive web application using modern frontend frameworks (e.g., React).
- Incorporate visually appealing graphics, images, and interactive elements for recipes.

2.3 Software Components:

2.3.1 Requirement:

The system shall consist of modular software components for efficient development and maintainability.

2.3.2 Specification:

- Define separate components for frontend, backend, and AI model integration.
- Ensure modularity to facilitate future enhancements and updates.

3. Functional Requirements:

3.1 Recipe Generation:

3.1.1 Requirement:

The system shall generate clear and concise step-by-step recipes based on user requests.

3.1.2 Specification:

- Implement a recipe generation algorithm within the AI model.
- Ensure support for various Indian cuisines and regional specialties.

3.2 Personalization:

3.2.1 Requirement:

The system shall tailor recipe recommendations based on user preferences, dietary restrictions, and feedback.

3.2.2 Specification:

- Develop user profiles with relevant parameters for personalization.
- Integrate dynamic adjustment mechanisms based on user interactions and feedback.

3.3 User Input Handling:

3.3.1 Requirement:

The system shall implement robust error detection and correction mechanisms to handle ambiguous or unclear user inputs.

3.3.2 Specification:

- Integrate natural language understanding techniques to identify and rectify user input errors.
- Provide informative error messages for user guidance.

4. Integration:

4.1 External Platforms:

4.1.1 Requirement:

The system shall explore integration possibilities with external platforms and devices, considering compatibility with smart kitchen appliances.

4.1.2 Specification:

- Investigate APIs for potential collaborations with smart kitchen devices.
- Explore voice-activated assistant integration possibilities.

5. Sequence of Events:

5.1 Recipe Generation Flow:

5.1.1 Requirement:

Define the sequence of events for recipe generation, from user input to AI model output.

5.1.2 Specification:

- User inputs recipe request via the user interface.
- Frontend validates and formats the input before sending it to the backend.
- Backend processes the request, integrates the AI model, and generates a recipe.
- The recipe is presented to the user with relevant images and instructions.

6. Deployment:

6.1 Deployment Environment:

6.1.1 Requirement:

Define the environments for development, testing, and production deployment.

6.1.2 Specification:

- Development: Local development environments for individual developers.
- Testing: A staging environment for comprehensive testing and quality assurance.
- Production: Cloud-based infrastructure for scalable and reliable production deployment.

6.2 Deployment Automation:

6.2.1 Requirement:

Implement automated deployment processes to ensure consistency and efficiency.

6.2.2 Specification:

- Utilize containerization (e.g., Docker) for consistent deployment across environments.
- Implement continuous integration/continuous deployment (CI/CD) pipelines for automated testing and deployment.

7. Performance Metrics:

7.1 Success Criteria:

- Achieve positive user feedback and engagement metrics.
- Attain high accuracy in recipe generation and personalization.
- Successfully integrate with external platforms.

8. Assumptions and Constraints:

- Users will have basic internet connectivity for accessing recipe recommendations.
- The initial version will focus on vegetarian recipes; future updates may include non-vegetarian options.

9. Risks and Mitigation:

- Risk: Model may generate inaccurate or unsafe cooking instructions.
    - Mitigation: Implement a thorough validation process with human review for critical safety aspects.

- Risk: User preferences may vary significantly, impacting the effectiveness of personalization.
    - Mitigation: Regularly update the model with diverse user feedback to improve personalization accuracy.

10. Stakeholders:

- Project Owner
- Development Team
- QA Team
- Marketing Team
- End Users

11. Approval:

Project Owner:
[Your Name]

Date:

[Date]

This Software Technical Requirements Document provides detailed specifications for the development and implementation of the cookGPT project, ensuring alignment with the business objectives outlined in the Business Requirement Document. It includes specific details about system architecture, software components, sequence of events, deployment, and other technical aspects crucial for the successful execution of the project.