Smart Ghr Wali Kitchen Assistant Project

Software Requirements Specification (SRS)

# 1. Introduction

## 1.1 Purpose

The purpose of this SRS document is to specify the requirements for the "Smart Ghr Wali Kitchen Assistant," an IoT-based kitchen management system integrated with AI. This system automates ingredient tracking, grocery management, and recipe generation, enhancing convenience and efficiency in the kitchen.

## 1.2 Scope

The Smart Ghr Wali Kitchen Assistant will consist of smart kitchen jars equipped with screens and weight sensors, and a mobile application that interacts with these jars. The system will allow users to manage ingredients, generate grocery lists, retrieve and modify recipes, and utilize AI to generate new recipes based on available ingredients.

## 1.3 Definitions, Acronyms, and Abbreviations

- **IoT**: Internet of Things

- **AI**: Artificial Intelligence

- **Smart Jar**: A kitchen jar equipped with a screen and weight sensor.

- **App**: The mobile application that interacts with the Smart Ghr Wali Kitchen Assistant.

## 1.4 References

- IEEE Standard 830-1998: IEEE Recommended Practice for Software Requirements Specifications

## 1.5 Overview

This document provides a comprehensive overview of the system's functionality, user interactions, system constraints, and non-functional requirements.

# 2. Overall Description

## 2.1 Product Perspective

The Smart Ghr Wali Kitchen Assistant is an innovative IoT application that integrates kitchen management with AI-driven recipe generation. The system comprises smart jars that monitor ingredient quantities and a mobile application that tracks these quantities, generates grocery lists, and suggests recipes.

## 2.2 Product Functions

### Ingredient Management:

Track quantities of ingredients in smart jars and update the "Current Ingredients" file accordingly.

### Grocery List Generation:

Generate grocery lists based on depleted or low ingredients.

### Recipe Management:

Retrieve recipes from the "My\_Recipe\_Book" file, adjust them based on the number of servings, and update ingredient quantities after cooking.

### AI Recipe Generation:

Create recipes using AI based on available ingredients when a recipe is not predefined.

### User Feedback:

Allow users to provide feedback on recipes, which updates the "My\_Recipe\_Book" file.

## 2.3 User Classes and Characteristics

### Home Cooks:

Primary users who manage kitchen ingredients, prepare meals, and generate grocery lists.

### Administrators:

Manage system settings, user permissions, and maintenance of the database.

## 2.4 Operating Environment

### Hardware:

Smart jars with embedded screens and weight sensors, mobile devices running the application.

### Software:

Android/iOS-based mobile application, cloud-based database for storing ingredient and recipe data.

### Communication:

The system relies on Wi-Fi/Bluetooth for communication between smart jars and the mobile application.

## 2.5 Design and Implementation Constraints

- The system must ensure real-time communication between the jars and the application.

- AI algorithms must generate recipes quickly and accurately based on available ingredients.

- Compliance with data privacy standards for user data storage.

## 2.6 Assumptions and Dependencies

- Users have access to a stable internet connection.

- The smart jars are correctly labeled and configured during initial setup.

# 3. System Features

## 3.1 Ingredient Management

### 3.1.1 Description and Priority

This feature allows the system to monitor and update the quantities of ingredients stored in the smart jars. High priority.

### 3.1.2 Functional Requirements

- The system shall detect the weight of the ingredients in each jar and update the "Current Ingredients" file.

- The system shall allow users to reassign jar labels (e.g., changing a jar labeled "Salt" to "Sugar").

- The system shall notify users when an ingredient quantity falls below a predefined threshold.

## 3.2 Grocery List Generation

### 3.2.1 Description and Priority

This feature automatically generates a grocery list based on the current inventory of ingredients. High priority.

### 3.2.2 Functional Requirements

- The system shall compare the "Current Ingredients" file against predefined thresholds and generate a list of items to purchase.

- The system shall allow users to view and modify the generated grocery list through the mobile application.

- The system shall support generating a grocery list remotely via the mobile app.

## 3.3 Recipe Management

### 3.3.1 Description and Priority

This feature manages user-defined recipes, allowing retrieval and adjustment based on available ingredients and serving sizes. High priority.

### 3.3.2 Functional Requirements

- The system shall allow users to retrieve a recipe from the "My\_Recipe\_Book" file based on the dish name.

- The system shall adjust ingredient quantities based on the number of servings requested.

- The system shall update the "Current Ingredients" file after a recipe is used, subtracting the quantities of ingredients used.

## 3.4 AI Recipe Generation

### 3.4.1 Description and Priority

This feature generates recipes using AI based on the ingredients available at home. Medium priority.

### 3.4.2 Functional Requirements

- The system shall generate a recipe using AI if the requested recipe is not available in the "My\_Recipe\_Book" file.

- The system shall ensure the generated recipe uses only ingredients that are available in the "Current Ingredients" file.

## 3.5 User Feedback

### 3.5.1 Description and Priority

This feature allows users to provide feedback on recipes, which the system uses to update the recipes. Medium priority.

### 3.5.2 Functional Requirements

- The system shall allow users to provide feedback on a recipe after it has been generated.

- The system shall update the "My\_Recipe\_Book" file based on user feedback (e.g., adjusting ingredient quantities).

- The system shall store feedback history for future reference.

# 4. External Interface Requirements

## 4.1 User Interfaces

- The mobile application shall provide a user-friendly interface for managing ingredients, generating grocery lists, and retrieving recipes.

- The smart jars shall display current ingredient quantities and allow label reassignment through their screens.

## 4.2 Hardware Interfaces

- The smart jars shall interface with the weight sensors and screens to accurately monitor and display ingredient data.

## 4.3 Software Interfaces

- The mobile application shall interface with a cloud-based database to store and retrieve ingredient and recipe data.

- The system shall interface with AI APIs for recipe generation.

## 4.4 Communications Interfaces

- The system shall use Wi-Fi/Bluetooth for communication between the smart jars and the mobile application.

# 5. Non-Functional Requirements

## 5.1 Performance Requirements

- The system shall update ingredient quantities in real-time as ingredients are added or removed from the jars.

- The system shall generate grocery lists and AI-based recipes within 5 seconds.

## 5.2 Security Requirements

- The system shall encrypt all user data stored in the cloud database.

- The system shall authenticate users before allowing access to the mobile application.

## 5.3 Usability Requirements

- The mobile application shall be intuitive, requiring minimal training for users.

- The smart jar interfaces shall be simple to use, with clear instructions displayed on the screens.

## 5.4 Reliability Requirements

- The system shall maintain a 99.9% uptime, ensuring that ingredient data is always accessible.

- The system shall automatically back up data to prevent loss in case of hardware failure.

# 6. Other Requirements

- The system shall include detailed user documentation for both the mobile application and smart jars.

- The system shall support future updates to incorporate additional AI features and enhanced user feedback mechanisms.