Software Requirements Specification

TasteBud Recipes Platform

Version: 1.0
 Date: June 22, 2025

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document describes the functional and non-functional requirements for the TasteBud Recipes platform, a web-based recipe sharing application developed for TasteBud Technologies.

1.2 Scope

The TasteBud Recipes platform is a web application that allows cooking enthusiasts to share, discover, and rate recipes. The system consists of:

- ASP.NET MVC web application for end-users
- ASP.NET Web API for future mobile applications
- User management system
- Recipe management system
- Search and rating functionality
- Basic administrative interface

1.3 Definitions and Acronyms

- **SRS**: Software Requirements Specification
- **MVC**: Model-View-Controller
- **API**: Application Programming Interface
- **CRUD**: Create, Read, Update, Delete
- **ERD**: Entity Relationship Diagram
- **UI/UX**: User Interface/User Experience

Overall Description

2.1 Product Perspective

TasteBud Recipes is a standalone web application designed to serve as a digital cookbook platform where users can share and discover recipes. The system will be hosted on standard web servers and designed for scalability.

2.2 Product Functions

The main functions of the system include:

- User registration and authentication
- Recipe creation, editing, and deletion
- Recipe browsing and searching
- Recipe rating system
- Administrative functions
- Responsive web interface
- RESTful API interface

2.3 User Classes and Characteristics

- **Regular Users**: Home cooks and cooking enthusiasts who create accounts to share and discover recipes
- **Anonymous Users**: Visitors who can browse and search recipes without registration
- **Administrators**: System managers who can moderate content and manage users

2.4 Operating Environment

- **Client Side**: Modern web browsers (Chrome, Firefox, Safari, Edge)
- **Server Side**: Standard web server environment supporting ASP.NET
- **Database**: SQL Server or compatible database system
- **Platforms**: Cross-platform web application accessible on desktop and mobile devices

2.5 Design and Implementation Constraints

- Limited budget for initial version
- Standard web server hosting
- Expected initial user base: 100-500 users in first month
- Must work on all modern browsers
- Mobile-responsive design mandatory
- Text-only recipes (no images in Phase 1)

2.6 Assumptions and Dependencies

- Users have access to modern web browsers
- Stable internet connection for users
- Database server availability
- Web server hosting environment configured for ASP.NET applications

2. System Overview

- **Tech Stack:** ASP.NET MVC, ASP.NET Web API, SQL Server, Bootstrap

- **Key Features:**

  - User registration and login

  - Recipe creation, browsing, editing, and deletion

  - Search and rating system

  - Admin controls

  - Responsive and mobile-friendly design

3. User Roles

- **Regular Users:** Share and explore recipes

- **Anonymous Users:** Browse and search only

- **Admins:** Manage users and moderate recipes

4. Functional Requirements

- **User Management:** Register, log in/out, update profile

- **Recipe Management:** Create, view, edit, delete recipes

- **Search & Discovery:** Browse by filters, search by title/ingredients

- **Rating System:** 1–5 stars, one rating per user per recipe

- **Admin Functions:** User deactivation, recipe moderation

5. Non-Functional Requirements

- **Performance:** Page load < 3s; support 100–500 users

- **Security:** Encrypted passwords, role-based access

- **Usability:** Intuitive UI, mobile support, accessible design

- **Reliability:** 99% uptime, daily backups, friendly error handling

- **Compatibility:** Works on all major browsers and devices

6. Architecture

- **Layers:** MVC + Web API + Entity Framework + SQL Server

- **API:** RESTful with JSON support

- **Database:** Normalized schema with users, recipes, ratings, categories

7. UI Pages

- **Public:** Home, browse, search, login/register

- **User:** Dashboard, create/edit recipes, profile

- **Admin:** Manage users and recipes

8. QA & Testing

- Unit, integration, UI, API, and cross-browser testing

- Full documentation: API, user manual, database schema

9. Constraints & Future Scope

- **Constraints:** 3-day build, text-only recipes, no images

- **Future Enhancements:** Images, comments, favorites, social sharing, notifications

10. Acceptance Criteria

- All key features work as expected

- Fast performance and smooth UI

- Tested for usability and compatibility

11. Approval

Document prepared by the Development Team
 Awaiting approval from CEO, Dev Lead, and QA Lead