Code Academy
أكاديمية البرمجة

مكين
Makeen

سلطنة عُمان
وزارة النقل والاتصالات وتقنية المعلومات
Sultanate of Oman
Ministry of Transport, Communications and Information Technology

TASTE BUD

Recipe Sharing Platform (SRS)

Kauthar alhashmi Dev

1. Introduction
1.1 Purpose
The purpose of this document is to define the requirements for the TasteBud Recipe Sharing Platform — a web application that allows users to create, browse, and rate recipes.

1.2 Scope
The application will support:

- User registration and login
- Recipe creation, editing, and deletion
- Recipe browsing and search
- Recipe rating system
- Basic admin features
- An API and web-based MVC interface

1.3 Definitions

- CRUD: Create, Read, Update, Delete
- MVC: Model-View-Controller
- API: Application Programming Interface
- EF: Entity Framework
- ASP.NET Identity: Authentication and user management system

2. Overall Description
2.1 Product Perspective
TasteBud is a brand-new system, built from scratch using a 3-tier architecture:

- Presentation Layer: ASP.NET MVC & HTML/CSS/JS
- Business Logic Layer: .NET Core services
- Data Access Layer: Entity Framework Code First

2.2 User Classes

Guest → Can browse and search recipes
Registered User → Can add, edit, delete own recipes, and rate others
Admin → Can manage users and remove any recipe

2.3 Operating Environment

- Web browser (desktop & mobile)
- Hosted on standard Windows web server
- Responsive design (mobile-friendly)

2.4 Assumptions & Dependencies

- No image uploads in v1
- Users are trusted to post appropriate content
- API and MVC share the same business logic
- Email-based registration

3. Functional Requirements

3.1 User Management

- Register using email & password
- Login and logout
- View and update profile

3.2 Recipe Management

- Add a recipe with:
  - Title, Description, Ingredients, Instructions
  - Prep & Cook time, Servings, Difficulty, Category
- Edit and delete own recipes
- Browse all recipes
- Search by title or ingredients

3.3 Rating System

- Authenticated users can rate any recipe (1–5 stars)
- A recipe’s average rating is displayed

3.4 Admin Features

- View all users and recipes
- Delete any recipe
- Disable/remove user accounts

4. Non-Functional Requirements

Performance: Support 100–500 users initially
Security: Use ASP.NET Identity with hashed passwords
Scalability: Architecture should allow future features (e.g. images, comments)
Reliability: Minimal downtime, graceful error handling
Usability: Simple UI, large buttons, mobile support

5. Use Cases

Register User: User creates an account
Login User: User logs in to the system
Create Recipe: Authenticated user adds a recipe
Edit/Delete Recipe: User modifies their own recipes
Browse/Search Recipes: Any user can search recipes by keyword
Rate Recipe: Authenticated user rates a recipe
Admin Manage Users: Admin views and manages all users
Admin Manage Recipes: Admin removes inappropriate recipes

6. User Stories

- As a user, I want to register with email and password so I can share my own recipes.
- As a user, I want to search recipes by ingredients so I can use what I already have.
- As a user, I want to rate recipes so I can help others find the best ones.
- As an admin, I want to delete inappropriate recipes to maintain platform quality.

7. Assumptions & Constraints

- No recipe image uploads in v1
- All recipes are published immediately
- Authentication required for adding or rating recipes
- Admins are responsible for content moderation
- Future mobile app will use the same API