

Expanded Software Requirements Specification (SRS)

Project: Recipe Sharing Platform

1. Introduction

1.1 Purpose

This document provides a detailed description of the functional and non-functional requirements for the "Recipe Sharing Platform". The system will serve as a web-based solution that allows users to create, manage, and share cooking recipes. It will also include administrative tools for monitoring and content moderation.

1.2 Scope

The application will:

- Allow users to register and log in.
- Enable users to create, view, edit, and delete recipes.
- Allow users to rate other users' recipes.
- Provide advanced search functionality.
- Include an admin interface to manage users and content.
- Be built using ASP.NET MVC, Web API, and Entity Framework.

1.3 Definitions and Acronyms

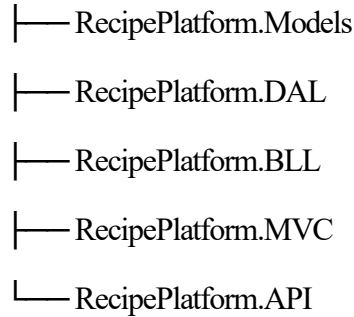
- **SRS**: Software Requirements Specification
 - **CRUD**: Create, Read, Update, Delete
 - **MVC**: Model View Controller
 - **API**: Application Programming Interface
 - **DAL**: Data Access Layer
 - **BLL**: Business Logic Layer
-

2. Overall Description

2.1 Product Perspective

This platform is a modular, scalable web application developed using a multi-layered architecture. It is designed to integrate easily with mobile apps or third-party services through RESTful APIs. The architecture will follow:

Solution:



2.2 User Classes and Characteristics

User Role	Description
Guest	Can browse and search recipes.
Registered User	Can create, edit, delete, and rate recipes. Can update profile.
Admin	Full access to manage content, users, and site-wide settings.

2.3 Operating Environment

- ASP.NET MVC (Frontend)
- ASP.NET Web API (Backend API Layer)
- SQL Server (Database)
- Visual Studio 2022
- IIS / Azure Hosting
- Modern Browsers (Chrome, Firefox, Edge)

2.4 Assumptions and Dependencies

- Users have reliable internet access.
 - All client-side scripts run on supported browsers.
 - Admin roles are manually assigned by database seeding or super admin.
-

3. Functional Requirements

3.1 User Management

- Register new users with email/password.
- Login/Logout with session or token.
- Reset password via email.
- Edit profile (username, bio, profile picture).

3.2 Recipe Management

- Add new recipes with:
 - Title
 - Description
 - Ingredients (list)
 - Instructions (step-by-step)
 - Optional images
 - Category (e.g., Dessert, Main Dish)
- Edit/Delete your own recipes.
- View recipe details.

3.3 Search Functionality

- Basic keyword search.
- Advanced filtering:
 - By category
 - By rating
 - By ingredient
 - By author

3.4 Rating System

- Logged-in users can rate each recipe 1–5 stars.
- Rating is averaged and shown publicly.
- Users can only rate a recipe once.

3.5 Admin Features

- View/manage all users.
 - Delete inappropriate recipes.
 - Ban/unban users.
 - Monitor reports or abuse flags.
-

4. Non-Functional Requirements

4.1 Performance

- The system must handle up to 10,000 active users.
- Response time for pages < 3s.
- API latency < 500ms.

4.2 Security

- All passwords are hashed using industry-standard algorithms (e.g., SHA256 + Salt).
- Preventing XSS and SQL Injection attacks.
- Secure API using authentication tokens (JWT or cookies).

4.3 Usability

- Responsive design for desktops and mobile.
- User-friendly forms with validation and feedback.
- Accessible for users with disabilities.

4.4 Reliability

- 99.9% system uptime.
 - Daily database backups.
 - Logging and error reporting system.
-

5. Use Cases

5.1 Register New User

Actors: Guest

Description: A user provides email, password, and optionally a profile name to create an account.

Precondition: Email not already used.

Postcondition: Account is created, and user is redirected to dashboard.

5.2 Login

Actors: Registered User

Description: A user provides valid credentials and logs into the system.

Precondition: User must be registered.

Postcondition: Session starts, and user accesses their profile.

5.3 Create Recipe

Actors: Registered User

Description: User enters recipe details and submits it.

Precondition: User must be logged in.

Postcondition: Recipe is saved and visible to public.

5.4 Search Recipes

Actors: Any User

Description: User searches using keywords or filters.

Precondition: Recipes exist in the database.

Postcondition: Search results displayed.

5.5 Rate Recipe

Actors: Registered User

Description: User rates a recipe from 1 to 5.

Precondition: Recipe must be published.

Postcondition: Average rating is updated.

5.6 Admin Manage Content

Actors: Admin

Description: Admin deletes content, bans users, or handles abuse reports.

Precondition: Admin is logged in.

Postcondition: Content/user changes are reflected in the system.