Software Requirements Specification (SRS)

Scent Sense: AI-Powered Perfume Recommendation System

Prepared by:
Mohammed Showharwade Neshad
ID: 2010398

Department of Computer Science and Engineering (CSE)
Independent University, Bangladesh (IUB)

May 13, 2025

Contents

1 Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document outlines the complete software requirements for "Scent Sense" - an AI-powered web application that assists users in discovering perfumes based on their preferences, personality, and emotional state. The app combines user input with AI-generated recommendations, styled in a sleek and modern UI.

The primary objective is to develop a smart, aesthetic, and interactive perfume recommendation tool that:

- Suggests perfumes using keyword, quiz, and mood input.
- Leverages artificial intelligence and external APIs for images and data.
- Provides a personalized user experience with modern design principles.

1.2 Intended Audience

This document is intended for:

- Project evaluators and faculty members
- Project supervisor/mentor
- Any stakeholder involved in assessing project scope and deliverables

1.3 Scope

The application will be delivered as a responsive web platform. It includes:

- Frontend interface for interaction and display
- AI-based backend requests to Groq API
- Dynamic perfume suggestions
- Image fetching via Pexels
- Local storage for user preferences (favorites)

2 Requirements Elicitation

2.1 Techniques

Multiple elicitation techniques were utilized to gather detailed insights from stakeholders:

- **Interviews**: One-on-one sessions with potential users (students, friends, and faculty members)
- **Questionnaires**: Online form shared among peers with questions about perfume preferences

- **Observation**: Competitive analysis of similar websites and apps
- **Prototyping**: Basic low-fidelity wireframe for early feedback
- **Brainstorming**: Internal sessions for creative ideas like mood-based recommendations

2.2 Preparation

Before conducting elicitation sessions, several preparations were made:

- Stakeholders were identified (users, instructors, technical reviewers)
- A goal-driven questionnaire was drafted
- Prototypes were prepared for visual demonstration
- Pexels and Groq APIs were reviewed to confirm feasibility

2.3 Conducting the Elicitation

2.3.1 Opening/Introduction

The elicitation began with an overview of the system's objective - to create a beautiful, minimal, and intuitive perfume recommendation platform with AI.

2.3.2 Body

Stakeholders provided insights on required features, such as:

- A quiz-based personality detector
- Keyword search interface
- Mood-based suggestions
- Image support and real-time links to product purchases
- Visual appeal and theme preferences

2.3.3 Closing

Sessions ended by confirming interpretations and validating early requirements.

2.3.4 Follow-up

Additional emails and messages were exchanged to clarify inputs and validate assumptions.

2.4 Analyzing Elicitation Results

The data from interviews and forms were grouped into three main types:

- **Core Functionalities**: Keyword suggestions, quiz module, mood matcher
- **Enhanced UX**: Favorites list, responsive layout, soft background
- **System Constraints**: Use of AI APIs (Groq), image APIs (Pexels), and local storage

2.5 Difficulties of Requirements Elicitation

- **Ambiguity**: Users sometimes struggled to articulate technical expectations
- **Incompleteness**: Early responses lacked clarity on preferred features
- **Time Constraints**: Limited availability from stakeholders
- **Scope Drift**: Frequent new ideas risked feature creep

3 Requirements Analysis and Negotiation

3.1 SMART Objectives Analysis

All core requirements were assessed using the SMART framework:

- **Specific**: Requirements are clearly defined
- **Measurable**: Success can be measured by feature presence and performance
- **Achievable**: Technically feasible within timeline
- **Relevant**: Each feature contributes to personalized fragrance recommendation
- **Time-bound**: Core features to be implemented within available time

3.2 MoSCoW Prioritization

Requirements were prioritized as follows:

Must Have:

- Keyword-based perfume search
- Quiz-based recommendation
- Mood-based suggestion
- AI integration via Groq API
- Responsive and aesthetic UI

Should Have:

- Save to Favorites (via localStorage)

- Real-time image fetching using Pexels API

Could Have:

- Responsive animations
- Animated transitions

Won't Have (for now):

- User authentication
- Admin backend or user database

4 System Features & Requirements

4.1 Functional Requirements

FR Code	Description
FR1	The system shall allow users to search perfumes by keyword.
FR2	The system shall suggest perfumes based on quiz inputs (season, color, occasion).
FR3	The system shall recommend perfumes based on selected mood.
FR4	The system shall display each perfume suggestion with name, description, image, and Amazon link.
FR5	The system shall allow users to save perfumes to a "Favorites" list.
FR6	The system shall persist favorites using localStorage.
FR7	The system shall allow users to remove items from Favorites.

Table 1: Functional Requirements

4.2 Non-Functional Requirements

5 Monitoring Strategy

- **Performance Monitoring**: Chrome DevTools and browser console used to monitor load times
- **Error Logging**: JavaScript error logs for failed image loads
- **Feedback Loop**: Informal user testing after each major update
- **Manual QA**: Each feature tested manually on Chrome and Safari

NFR Code	Description
NFR1	The UI shall be visually appealing, soft-themed, and intuitive.
NFR2	AI and image results should return within 5 seconds.
NFR3	Works on all modern browsers and mobile devices.
NFR4	No user data is stored on servers.
NFR5	Modular architecture allows for backend integration in future.
NFR6	Separated code files for clarity and future edits.

Table 2: Non-Functional Requirements

6 Risk Management

- **API Downtime**: Mitigated by using fallback descriptions
- **Design Delays**: Solved by locking CSS early
- **Response Inconsistency**: Prompt tuning and regex-based parsing
- **Short Timeline**: Features prioritized using MoSCoW

7 Change Management Process

The following change management process was followed:

1. **Capture**: All user suggestions documented
2. **Assess**: Reviewed for feasibility and effort
3. **Prioritize**: Categorized as must-have or optional
4. **Implement**: High-priority changes applied immediately
5. **Document**: All changes logged in SRS

8 Conclusion

This SRS provides a complete specification for the development of the "Scent Sense" perfume recommendation system. It ensures all stakeholders are aligned with the scope, features, risks, and non-functional targets of the application. Through this structured requirement documentation, the project was successfully delivered within time and aesthetic expectations.