# Software Requirements Specification for

Recipe recommendation system

1. Introduction

1.1 Purpose

The purpose of this document is to build a recipe management system to identify ingredients and suggest recipe according to ingredients

1.2 Document Conventions

This document uses the following conventions.

|  |  |
| --- | --- |
| DB | Database |
| ER | Entity relationship |
| DFD | Data flow diagram |
|  |  |

1.3 Intended Audience and Reading Suggestions

This project is a prototype for the recipe recommendation system and it is restricted to small as well as large restaurant . This has been implemented under the guidance of college professors. This project is useful for the general users team and as well as to the restaurants.

1.4 Product Scope

Recommender systems are truly helpful when the recommended item is something that the user has not seen in the past. For example, popular movies of a preferred genre would rarely be novel to the user. Repeated recommendation of popular items can also lead to reduction in sales diversity

It is can be used in various small as well as large restaurants

We are providing registration and login credentials to users for this application. User can upload photos choose from gallery. User will upload image to server as input. Another web application is developed. Admin and user will be provided login to web application. User will select image from database. Ingredients will be identified. Recipe is suggested for the above identified ingredients. Admin will upload recipe. Admin can view user’s history. User can also add recipe and verified by admin to approve contents. User can reviews about recipe.

1.5 References

[1] G. Shani and A. Gunawardana, “Evaluating recommendation systems,” *Recommender systems handbook*, Springer, 2011, pp. 257–297.

[2] D.N. Chin, “Empirical evaluation of user models and user-adapted systems,” *User modeling and user-adapted interaction*, vol. 11, 2001, pp. 181–194.

[3] P. Brusilovsky and E. Millán, “User models for adaptive hypermedia and adaptive educational systems,” *The adaptive web*, 2007, pp. 3–53.

[4] S. Lam, D. Frankowski, and J. Riedl, “Do you trust your recommendations? An exploration of security and privacy issues in recommender systems,” *Emerging Trends in Information and Communication Security*, 2006, pp. 14–29.

[5] N. Good, J.B. Schafer, J.A. Konstan, A. Borchers, B. Sarwar, J. Herlocker, and J. Riedl, “Combining collaborative filtering with personal agents for better recommendations,” *Proceedings of the National Conference on Artificial Intelligence*, JOHN WILEY & SONS LTD, 1999, pp. 439–446.

[6] C.D. Manning, P. Raghavan, and H. Schütze, *An Introduction to Information Retrieval*, Cambridge University Press, Cambridge, England, 2009.

[7] J. Beel, S. Langer, and M. Genzmehr, “Sponsored vs. Organic (Research Paper) Recommendations and the Impact of Labeling,” *Proceedings of the 17th International Conference on Theory and Practice of Digital Libraries (TPDL 2013)*, T. Aalberg, M. Dobreva, C. Papatheodorou, G. Tsakonas, and C. Farrugia, eds., Valletta, Malta: 2013, pp. 395–399.

[8] J. Beel, S. Langer, A. Nürnberger, and M. Genzmehr, “The Impact of Demographics (Age and Gender) and Other User Characteristics on Evaluating Recommender Systems,” *Proceedings of the 17th International Conference on Theory and Practice of Digital Libraries (TPDL 2013)*, T. Aalberg, M. Dobreva, C. Papatheodorou, G. Tsakonas, and C. Farrugia, eds., Valletta, Malta: Springer, 2013, pp. 400–404.

[9] M. Ge, C. Delgado-Battenfeld, and D. Jannach, “Beyond accuracy: evaluating recommender systems by coverage and serendipity,” *Proceedings of the fourth ACM conference on Recommender systems*, ACM, 2010, pp. 257–260.

2. Overall Description

2.1 Product Perspective

A database system stores the following information.

* **Recipe details:**  
  It includes number of ingredients as well as suggested recipe.
* **Customer details:**  
  It includes customer id, name, address and phone number and so on. This information may be used for keeping the records of the customer for any other kind of information.
* **History:**  
  It includes recipe viewed by customer.
* **Review**

It include customer review for suggested recipe

2.2 Product Functions

Registration and login of user

Upload image

Identify ingredients

Suggest recipe

Review of users

2.3 User Classes and Characteristics

The Employee should have following management functionalities:

* CUSTOMER FUNCTIONS.  
  • Registration  
  • login to application.  
  • View details.  
  • View history
* ADMINISTRATIVE  
  • Add/Delete a recipe  
  • Update details.

2.4 Operating Environment

Front end:- JSP servlet, HTML, CSS

Backend :- My SQL

Android studio

Wamp server

Operating System : Windows 10

Technology : Java, J2EE

Web Technologies : Html, JavaScript, CSS

Web Server : Wamp server

Database : My SQL

Java Version : J2SDK 1.7 / 1.8

2.5 Design and Implementation Constraints

Number of recipe

There should be minimum 2 recipe should be available in app.

Image

Image should be cleared. While recognizing image, image should be noise less. Without image , system is not able to identify ingredients.

Database

Database should be properly maintained. As all details are stored in database, it should be maintained.

Response time

Response time for searching query should be minimum which near by micro seconds

Android is constrained by Little Memory for Storage**.** Data Connection also affect android system Android has large number of background process which runs in the background, which eats so much mobile data. And thus cost lots of money if you are not into unlimited data plan.

It is also constrained by battery problem while android has many process running in the background this increase the usage of RAM and decreasing Battery Performance. While many top notch device has good battery backup (Lithium Batteries) but still that doesn’t mean it solved the problem.

While developing web based application, internet connection should be maintained. Also, speed of internet should be proper so that application can run without any interruption.

A security constraint can be set up to force certain areas of your site (or your entire site) into SSL mode.

This is useful if those areas will be used for confidential information, such as login details.

Web resource collection

A web resource collection is a list of URL patterns (the part of a URL after the host name and port which you want to constrain) and HTTP operations (the methods within the files that match the URL pattern which you want to constrain (for example, POST, GET)) that describe a set of resources to be protected.

Authorization constraint

An authorization constraint establishes a requirement for authentication and names the roles authorized to access the URL patterns and HTTP methods declared by this security constraint. If there is no authorization constraint, the container must accept the request without requiring user authentication. If there is an authorization constraint, but no roles are specified within it, the container will not allow access to constrained requests under any circumstances

User data constraint

A user data constraint establishes a requirement that the constrained requests be received over a protected transport layer connection. This guarantees how the data will be transported between client and server. The choices for type of transport guarantee include NONE, INTEGRAL, and CONFIDENTIAL.

2.6 User Documentation

User manual is provided along with project. Both , hard copy and soft copy will be provided with the project

2.7 Assumptions and Dependencies

One assumption about the product is that it will always be used on mobile phones and emulator that have enough performance. If the phone does not have enough hardware resources available for the application, for example the users might have allocated them with other applications, there may be scenarios where the application does not work as intended or even at all. Another assumption is that it will store all data. It contain little memory storage so it cannot store all data

3. External Interface Requirements

3.1 User Interfaces

A first-time user of the mobile application should see the log-in page when he/she opens the application. If the user has not registered, he/she should be able to do that on the log-in page.

If the user is not a first-time user, he/she should be able to see the search page directly when the application is opened. Here the user chooses the type of search he/she wants to conduct. Every user should have a profile page where they can edit their e-mail address, phone number and password. Also, the user can set the mobile application to his/her preferred language. The “P” icon shows where the user can click to navigate to his/her profile page.

* Front-end software: Java
* Back-end software: SQL+

3.2 Hardware Interfaces

Since neither the mobile application nor the web portal have any designated hardware, it does not have any direct hardware interfaces.

* Windows.
* A browser which supports CGI, HTML & Javascript.

3.3 Software Interfaces

Application will used Java language to develop project. Android platform is used along with sdk tool. We used SQLite in backend

3.4 Communications Interfaces

The communication between the different parts of the system is important since they depend on each other. However, in what way the communication is achieved is not important for the system and is therefore handled by the underlying operating systems for both the mobile application and the web portal.

4. System Features.

User function

Registration and login

|  |  |
| --- | --- |
| Use case name | Registration and login |
| Priority | Essential |
| Precondition | Login id and password should be there. |
| Basic path | 1. User will register by filling all details |
|  | 1. Click on register button 2. After registration, user will get login id and password 3. User will enter login id and password 4. Click on login button 5. Enter recipe name and random number will be generated 6. Click on start recipe button 7. Home page will appear |
| Alternative Path | None |
| Post condition | Home page will be displayed which contain browse, captured and upload |
| Other | None |

Image upload

|  |  |
| --- | --- |
| Use case name | Image upload |
| Priority | Essential |
| Precondition | Image should be captured and browse |
| Basic path | 1. Click on upload button 2. Gallery will be open 3. Select image from gallery 4. Click on OK button 5. Multiple Image will be uploaded to database |
| Alternative Path | None |
| Post condition | Image will be displayed on web app |
| Other | None |

Image captures

|  |  |
| --- | --- |
| Use case name | Image capture |
| Priority | Essential |
| Precondition | Image should be ready which to be captured |
| Basic path | 1. Click on captured button on homepage 2. Open camera 3. Captured image of recipe 4. Click on OK button 5. Image will be uploaded |
| Alternative Path | None |
| Post condition | Upload image to database |
| Other | None |

Browse image

|  |  |
| --- | --- |
| Use case name | Browse image |
| Priority | Essential |
| Precondition | Image should be present in gallery |
| Basic path | 1. Click on browse button 2. Gallery will be open 3. Select image from gallery 4. Click on OK button 5. Image will be uploaded |
| Alternative Path | None |
| Post condition | Upload image to database |
| Other | None |

Display image

|  |  |
| --- | --- |
| Use case name | Display image |
| Priority | Essential |
| Precondition | None |
| Basic path | 1. Display image on web page 2. Click on button ”recommend recipe” |
| Alternative Path | None |
| Post condition | Recipe related to ingredients will be suggested |
| Other | None |

Filter content

|  |  |
| --- | --- |
| Use case name | Filter content |
| Priority | Essential |
| Precondition | Recipe relevant to ingredients should be identified |
| Basic path | 1. Content will be filtered in image to identify ingredients 2. Recommend name of ingredients with recipe 3. Recipe related to corresponding ingredients fetch from database 4. Recipe will be displayed to user |
| Alternative Path | None |
| Post condition | User will get recommended recipe |
| Other | None |

Post new recipe

|  |  |
| --- | --- |
| Use case name | Post new recipe |
| Priority | Essential |
| Precondition | Recipe should be present |
| Basic path | 1. Home page will display 2. Click on add new recipe button 3. Post new recipe 4. View history |
| Alternative Path | None |
| Post condition | User can view posted recipe |
| Other | None |

Review and rating

|  |  |
| --- | --- |
| Use case name | Review and rating |
| Priority | Essential |
| Precondition | User should be login to application |
| Basic path | 1. Display homepage 2. User view history 3. User rate to application 4. Click on button |
| Alternative Path | None |
| Post condition | List of rating get displayed |
| Other | None |

Admin function

Authentication

|  |  |
| --- | --- |
| Use case name | Authentication |
| Priority | Essential |
| Precondition | Login id and password should be correctly filled |
| Basic path | 1. Admin enter login id and password 2. Click on button 3. Admin will verify login id and password of user 4. If login credentials are correct then admin will authenticate user otherwise it will deny access |
| Alternative Path | None |
| Post condition | Homepage will be displayed |
| Other | None |

|  |  |
| --- | --- |
| Use case name | Add Ingredients |
| Priority | Essential |
| Precondition | Ingredients list should be present |
| Basic path | 1. Select add ingredients feature 2. Enter ingredients 3. Add nutrition value 4. Click on submit 5. Ingredients will be added to application |
| Alternative Path | None |
| Post condition | List of ingredients will be view |
| Other | None |

|  |  |
| --- | --- |
| Use case name | Add recipe |
| Priority | Essential |
| Precondition | Ingredients list should be present |
| Basic path | 1. Click on add recipe 2. Enter recipe name 3. Select ingredients list 4. Enter recipe procedure 5. Click on submit |
| Alternative Path | None |
| Post condition | List of list will be view |
| Other | None |

|  |  |
| --- | --- |
| Use case name | View registered user |
| Priority | Essential |
| Precondition | Registered user list should be present in database |
| Basic path | 1. Click on button 2. view registered user |
| Alternative Path | None |
| Post condition | List of user displayed |
| Other | None |

|  |  |
| --- | --- |
| Use case name | Verify and manage recipe |
| Priority | Essential |
| Precondition | Recipe should be present |
| Basic path | 1. Admin can view recipe |
| Alternative Path | None |
| Post condition | None |
| Other | None |

# 5. Other Nonfunctional Requirements

## 5.1Performance Requirements

5.1.1 TITLE: Prominent detect feature

DESC: The search feature should be prominent and easy to find for the user

The framework must be intuitive and the postpones included must be less .So in each activity reaction of the framework, there are no quick deferrals. In the event of opening windows shapes, of popping mistake messages and sparing the settings or sessions there is defer much underneath 2 seconds, if there should be an occurrence of opening databases, arranging inquiries and assessment there are no postponements and the operation is performed in under 2 seconds for opening ,arranging, registering, posting > 95% of the records. Additionally when interfacing with the server the postponement is construct altering in light of the separation of the 2 frameworks and the design between them so there is high likelihood that there will be or not a fruitful association in under 20 seconds for purpose of good correspondence.

5.1.2 TITLE: Usage of the detect feature

DESC: The detect feature should be evident, simple and easy to understand.

5.1.3 : TITLE: Response Time

DESC: The fastness of the search. Transmission time gets added to response time when your request and the resulting response has to travel over a network and it can be very significant.

## Safety Requirements

Android phone should be maintained. Data base should be properly maintained.Information transmission should be securely transmitted to server without any changes in information. The main security concern is for users account hence proper login mechanism should be used to avoid hacking. The tablet id registration is way to spam check for increasing the security. Hence, security is provided from unwanted use of recognition software.

## Security Requirements

User should contain login ID and password. Applications designed with security in mind are safer than those where security is an afterthought. Traditionally security issues are first considered during the Design phase of the Software Development Life Cycle (SDLC) once the Software Requirements Specification (SRS) has been frozen. Login credentials will provide security to application. Authentication is main feature for system.

## Software Quality Attributes

* Software functional quality reflects how well it complies with or conforms to a given design, based on [functional requirements](https://en.wikipedia.org/wiki/Functional_requirements) or specifications. That attribute can also be described as the fitness for purpose of a piece of software or how it compares to competitors in the marketplace as a worthwhile [product](https://en.wikipedia.org/wiki/Product_%28business%29);
* Software structural quality refers to how it meets [non-functional requirements](https://en.wikipedia.org/wiki/Non-functional_requirements) that support the delivery of the functional requirements, such as robustness or maintainability, the degree to which the software was produced correctly.

## Business Rules

User can use this app if and only if he registered. Or he\she should have ID and password.