Software Requirements Specification

for

<Fivers Food Booking App>

Version 1.0 approved

Prepared by <Kyle>

<NTU SCSE SC2006>

<8 Feb 2024>

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The Fivers Food Application is a web application or mobile application in development phase by the NTU SCSE SC2006 Fivers Team. This document outlines the structure for the software design.

1.2 Document Conventions

In this document, each point will have its own priority. Each point is taken as an elaboration of its parent point.

1.3 Intended Audience and Reading Suggestions

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers. Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>

This document is intended for software developers seeking to further develop this application. Start by having a read through this document before working on the other design models to simplify the whole design process.

1.4 Product Scope

<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals. Relate the software to corporate goals or business strategies. If a separate vision and scope document is available, refer to it rather than duplicating its contents here.>

The Fivers Food Application serves to aid Singaporeans find food that fit their dietary needs. This is a community collaborative application, where users will assist to update, if any, information of eateries around Singapore.

1.5 References

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>

2. Overall Description

2.1 Product Perspective

<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>

2.2 Product Functions

<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary (such as a bullet list) is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, is often effective.>

The Fivers Food application allows users to locate and/or reserve a seat at a restaurant near them. Users are also able to search for restaurants near them based on selected filters provided.

2.3 User Classes and Characteristics

<Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. Distinguish the most important user classes for this product from those who are less important to satisfy.>

2.4 Operating Environment

<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.>

2.5 Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software).>

Using the Google Maps API comes with technology constraints including usage limits, rate limiting, and dependency on Google's services. Data availability and accuracy may vary, while integration complexity demands technical expertise. Adapting to changes in the API and planning for scalability are crucial aspects of effective implementation.

2.6 User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>

2.7 Assumptions and Dependencies

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>

1) Assume that the user gives honest reviews and ratings for the restaurants as they are listed based on these factors.
2) Depends on google map api accuracy to provide an accurate location for the user.

3. External Interface Requirements

3.1 User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>

3.2 Hardware Interfaces

<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.>

3.3 Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>

3.4 Communications Interfaces

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>

4. System Features

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

4.1 Authentication by Fivers Start-Up Page

The “Fivers” food application must at the **Start-up Page**, provide the options for users to “Register for New Account” or “Login with Existing Account” to proceed with any interactions before the user is allowed to proceed with any interactions.

4.1.1 The Start-up Page must redirect the user to the Registration Page if the user chooses the “Register for New Account” option.

4.1.1.1 When the user chooses to sign up for a new account, he must provide a valid Singapore mobile phone number, with a valid 8 integers long number, and must start with ‘8’ or ‘9’. When the user submits his request the Registration System must send a Verification email to the submitted email within five minutes of request submission.

4.1.1.2 The Verification email contains a “Verify” which should bring the user to the Verification page.

4.1.1.2.1 The Verification page must allow the user to complete registration by providing a strong password.

4.1.1.2.2 If a strong password is not provided within 15 minutes, the user will have to conduct the registration from start again.

4.1.1.3 The Registration Page must prompt the user with “Invalid email” if the provided email has an invalid format.

4.1.1.4 The Registration Page should prompt the user with “Email already registered” should the account be found in the database.

4.1.1.4.1 The Registration System should provide an option for the user to navigate to the Login page to proceed with login.

4.1.1.4.2 The Registration System should also provide an option for the user to reattempt registration at the Registration page.

4.1.2 The Start-up Page must redirect the user to the Registration Page if the user chooses the “Login to Existing Account” option.

4.1.2.1 The Login Page must provide a secure login environment.

4.1.2.2 The Login Page must check for validity of account before navigating user to the Home Page

4.1.2.2.1 The Login Page must prompt the user with “Invalid phone number” if the provided number has an invalid format.

4.1.2.2.2 The Login Page must provide the user the option to “Register for a New Account” or “Try Again” if the account is not found in the existing database.

4.1.2.3 The Login Page should redirect the user to the Home Page once the Account field and Password field matches existing database data.

4.2 Home Page of the Application

The Home Page must display at the bottom of the page, four tabs available for user interaction, [1] Search Page, [2] Favourites Page, [3] My Booking Page, [4] My Profile Page.

4.2.1 The Home Page should display the Search Page upon successful Login by default.

4.2.2 The Home Page must redirect the user to the Page the user has selected.

4.3 Search Page of the Application

Users must be able to search for eateries on the Search Page, based on the combination of one or more search filter(s) which includes (i) Name, (ii) Dietary Restrictions, (iii) Proximity (iv) Waiting Time, (v) Affordability, and (vi) Ratings.

4.3.1 The system must return a list of restaurants that contain the text input by the user when the Name field is not empty. (method to check not empty)

4.3.2 The system must only return the list of eateries that meet the user's selected dietary restriction. This includes but is not limited to: (i) Halal food, (ii) Vegetarian food, (iii) Gluten-free, (iv) certain meat restrictions.

4.3.3 The system must only return the list of eateries that are within the user's selected distance radius.

4.3.4 The system must only return the list of eateries that have a waiting time shorter than or equal to the user's selected waiting time.

4.3.5 The system must only return the list of eateries that meet the price range selected by the user.

4.3.6 The system must only return the list of restaurants that meet the selected ratings by the user.

4.3.7 When the user submits his/her query request, with or without filters, a list of restaurants that matches with the query name will be displayed on the results page.

4.3.8 If a query request returns no results, the Search Page should either return the top restaurants near the user's location if GPS is activated, or return the user's favourited restaurants.

4.4 Reservation of a restaurant in the Fivers Application

The user must be able to submit a reservation for the eatery/restaurant he/she has selected after clicking on “Reserve”.

4.4.1 There should be a pop-up page to indicate to the user of a successful reservation

4.4.1.1 A successful reservation requires the user to complete the following required fields: (i) Date of reservation, (ii) Time of reservation, (iii) Number of dine-in pax.

4.4.2 The Reservation System should produce a pop-up page to inform the user of an unsuccessful reservation.

4.4.2.1 An unsuccessful reservation can be due to incomplete fields, in this instance, the Reservation System will have to highlight the incomplete field name in red, with an error prompt on top of the field name that says “*Field is required”

4.4.3 The user is not required to complete optional fields including but not limited to: (i) requires a high stool, (ii) request for indoor, (iii) request for non-smoking dining area.

4.5 My Bookings Page in the Fivers Application

The user must be able to view all of his successful reservations on the My Bookings Page.

4.5.1 The user should be allowed to amend his booking at any time before the scheduled date and time of reservation.

4.5.1.1 The allowed amendments are as follows: (i) Date of reservation, (ii) Time of reservation, (iii) Number of dine-in pax.

4.5.1.2 Upon successful amendments of booking the Reservation System must return the user to the My Booking Page

4.5.2 The user should be allowed to cancel his booking at any time before the scheduled date and time of reservation.

4.5.3 Upon a successful cancellation of reservation, the Reservation System must delete the booking from the user’s My Bookings Page

4.6 Favourites Page of the Fivers Application

The user should be able to select a restaurant and add it to “Favourites”.

4.6.1 The user must be able to select a restaurant from the Favourites Page and interact with the restaurant directly without going through the Search Page.

4.6.2 The user must be able to remove the restaurant from “Favourites” by selecting the restaurant and clicking on “Remove from Favourites”.

4.7 Reviews System of the Fivers Application

The user should be able to submit a review for the restaurant he has patronised.

4.7.1 The user must be able to select the number of stars for the ratings. This should range from 0 to 5 stars, inclusive, with only integers allowed.

4.7.1.1 The star review is a compulsory field.

4.7.1.2 The Review System should prompt the user if the user did not select a star.

4.7.1.3 The Review System must award the user two points for a star review.

4.7.2 The user should be able to post a text review for the restaurant he has patronised.

4.7.2.1 This text review should be limited to only 250 characters.

4.7.2.2 The Review System should prompt the user if the text exceeds 250 characters with “Review too long”.

4.7.2.3 The text review is an optional field.

4.7.2.4 The Review System must award the user three points for a text review.

4.7.3 Upon successful rating provided, the Review System should return the user to the Home Page.

4.8 My Profile Page of the Fivers Application

The user should be able to update his profile in the My Profile Page

4.8.1 The user should be able to update his contact number to his profile.

4.8.1.1 The Registration System must send a verification WhatsApp message with 4-digit verification code to the contact number within five minutes.

4.8.1.2 The user must verify the verification code before accepting the update to phone number.

4.8.2 The user should be able to update his profile name, which is the default name used for bookings.

4.8.3 The user must be able to update his dietary preferences to his profile.

4.8.3.1 The list of dietary preferences are limited to: (i) Halal food, (ii) Vegetarian food.

4.8.4 The user must be able to view the restaurants that he has provided a rating for.

4.8.5 The user should be able to view the number of points he has accumulated.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>

5.2 Safety Requirements

The developers of this application caution users to interact with restaurants in the physical world with enough research and reference external of the application. The developers will not be responsible for any harm, whether physical, psychological, caused to users in the usage of the application. This includes but is not limited to: food poisoning, wrong consumption outside of dietary restrictions.

5.3 Security Requirements

The developers of this application ensures a secure environment for users to interact with restaurants. The developers will not ask for any details for the user's credit or debit cards, nor solicit in any forms, financial assets of the user.

5.4 Software Quality Attributes

The developers ensure a user-friendly interface where users can easily search for the functions they wish to interact with. The developers will also ensure tutorials for new every new updates.

5.5 Business Rules

The developers of this application does not have any plans to sell this application.

6. Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>

Source: http://www.frontiernet.net/~kwiegers/process_assets/srs_template.doc