BrewersBuddy

Software Requirements Specification

Date

SWENG 500 Team 3

Geoff Blogreff

Gregg Ideus

Jonathon Parise

John Pistorius

Steven Platz

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| <date> | <Version 1> | <Your Name> | <First Revision> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Scope 1

1.3 Definitions, Acronyms, and Abbreviations 1

1.4 References 1

1.5 Overview 1

2. General Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Characteristics 2

2.4 General Constraints 2

2.5 Assumptions and Dependencies 2

3. Specific Requirements 2

3.1 External Interface Requirements 3

3.1.1 User Interfaces 3

3.1.2 Hardware Interfaces 3

3.1.3 Software Interfaces 3

3.1.4 Communications Interfaces 3

3.2 Functional Requirements 3

3.2.1 <Functional Requirement or Feature #1> 3

3.2.2 <Functional Requirement or Feature #2> 3

3.3 Use Cases 3

3.3.1 Use Case #1 3

3.3.2 Use Case #2 3

3.4 Classes / Objects 3

3.4.1 <Class / Object #1> 3

3.4.2 <Class / Object #2> 3

3.5 Non-Functional Requirements 4

3.5.1 Performance 4

3.5.2 Reliability 4

3.5.3 Availability 4

3.5.4 Security 4

3.5.5 Maintainability 4

3.5.6 Portability 4

3.6 Inverse Requirements 4

3.7 Design Constraints 4

3.8 Logical Database Requirements 4

3.9 Other Requirements 4

4. Analysis Models 4

4.1 Sequence Diagrams 5

4.3 Data Flow Diagrams (DFD) 5

4.2 State-Transition Diagrams (STD) 5

5. Change Management Process 5

A. Appendices 5

A.1 Appendix 1 5

A.2 Appendix 2 5

# 1. Introduction

The introduction to the Software Requirement Specification (SRS) document should provide an overview of the complete SRS document. While writing this document please remember that this document should contain all of the information needed by a software engineer to adequately design and implement the software product described by the requirements listed in this document. (Note: the following subsection annotates are largely taken from the IEEE Guide to SRS).

## 1.1 Purpose

BrewerBuddy will be an online application that allows home brewers to enter and track batch information for beer and wine.

## 1.2 Scope

*This subsection should:*

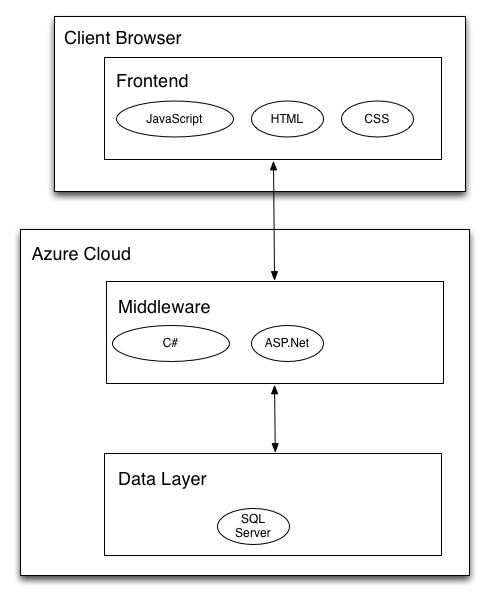
*(1) Identify the software product(s) to be produced by name; for example, Host DBMS, Report Generator, etc*

*(2) Explain what the software product(s) will, and, if necessary, will not do*

*(3) Describe the application of the software being specified. As a portion of this, it should:*

*(a) Describe all relevant benefits, objectives, and goals as precisely as possible. For example, to say that one goal is to provide effective reporting capabilities is not as good as saying parameter-driven, user-definable reports with a 2 h turnaround and on-line entry of user parameters.*

*(b) Be consistent with similar statements in higher-level specifications (for example, the System Requirement Specification) , if they exist.What is the scope of this software product.*



**Figure 1: High level System Diagram**

## 1.3 Definitions, Acronyms, and Abbreviations

*This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the SRS. This information may be provided by reference to one or more appendixes in the SRS or by reference to other documents.*

## 1.4 References

*This subsection should:*

*(1) Provide a complete list of all documents referenced elsewhere in the SRS, or in a separate, specified document.*

*(2) Identify each document by title, report number - if applicable - date, and publishing organization.*

*(3) Specify the sources from which the references can be obtained.*

*This information may be provided by reference to an appendix or to another document.*

## 1.5 Overview

*This subsection should:*

*(1) Describe what the rest of the SRS contains*

*(2) Explain how the SRS is organized.*

# 2. General Description

*This section of the SRS should describe the general factors that affect 'the product and its requirements. It should be made clear that this section does not state specific requirements; it only makes those requirements easier to understand.*

## 2.1 Product Perspective

*This subsection of the SRS puts the product into perspective with other related products or*

*projects. (See the IEEE Guide to SRS for more details).*

## 2.2 Product Functions

This subsection of the SRS should provide a summary of the functions that the software will perform.

## 2.3 User Characteristics

This subsection of the SRS should describe those general characteristics of the eventual users of the product that will affect the specific requirements. (See the IEEE Guide to SRS for more details).

## 2.4 General Constraints

*This subsection of the SRS should provide a general description of any other items that will*

*limit the developer’s options for designing the system. (See the IEEE Guide to SRS for a partial list of possible general constraints).*

## 2.5 Assumptions and Dependencies

This subsection of the SRS should list each of the factors that affect the requirements stated in the SRS. These factors are not design constraints on the software but are, rather, any changes to them that can affect the requirements in the SRS. For example, an assumption might be that a specific operating system will be available on the hardware designated for the software product. If, in fact, the operating system is not available, the SRS would then have to change accordingly.

# 3. Specific Requirements

This will be the largest and most important section of the SRS. The customer requirements will be embodied within Section 2, but this section will give the D-requirements that are used to guide the project’s software design, implementation, and testing.

Each requirement in this section should be:

* Correct
* Traceable (both forward and backward to prior/future artifacts)
* Unambiguous
* Verifiable (i.e., testable)
* Prioritized (with respect to importance and/or stability)
* Complete
* Consistent
* Uniquely identifiable (usually via numbering like 3.4.5.6)

Attention should be paid to the carefuly organize the requirements presented in this section so that they may easily accessed and understood. Furthermore, this SRS is not the software design document, therefore one should avoid the tendency to over-constrain (and therefore design) the software project within this SRS.

## 3.1 External Interface Requirements

### 3.1.1 User Interfaces

### 3.1.2 Hardware Interfaces

### 3.1.3 Software Interfaces

### 3.1.4 Communications Interfaces

## 3.2 Functional Requirements

This section describes specific features of the software project. If desired, some requirements may be specified in the use-case format and listed in the Use Cases Section.

### 3.2.1 <Functional Requirement or Feature #1>

3.2.1.1 Introduction

3.2.1.2 Inputs

3.2.1.3 Processing

3.2.1.4 Outputs

3.2.1.5 Error Handling

### 3.2.2 <Functional Requirement or Feature #2>

…

## 3.3 Use Cases

### Use Case #1 – User can create an account

User needs to indicate they want to create an account by clicking the create account button/link. The required data needed to create an account is their username, email, and password. The username needs to be unique. The user can also fill in their first name, last name, city, state, and zip at this time.

### Use Case #2 – Login

Users need to supply their user name and password to log into the system. If the input is invalid or incorrect they will be prompted to enter it again. If after three tries they still can't login, a forgot password dialog will be shown. Clicking the link will send an e-mail to them that they can use to change their password.

### Use Case #3 – Logout

Users that want to logout of their session will click the logout link. When this is done the system will log them out and take them back to the login screen. From here the user can log back in or leave the page.

### Use Case #4 – User can recover forgotten password

Users that want to recover their password will need to click the recover password link. The user will need to provide their username. The system will then send an email to the email address associated to that user name with their password in it.

### Use Case #5 – User can edit account information (Email, first and last name, password, username, city, state, zip)

Users will need to go to their account and click the edit account information link. From here users will be able to update their information. When they are done updating they will need to hit save to save the changes. Hitting cancel will disregard the changes.

### Use Case #6 – User can view their account information

Users will need to have an account and be logged in. Users will need to click the “my account information” link to be view their current information. Users will be able to see their first name, last name, email, username, city, state and zip.

### Use Case #7 – User can enter zip to find brewers around them

Users will need to click the search link. From there the need to enter their zip code. The user will then be able to see the list of brewers that match the entered zip code. The system will show first name, last name, username, and email in this list.

### Use Case #8 – Users can look up friends

Users will need to click the search link. From there they need to enter the username or first name or last name of the person they are looking for. The system will show the first name, last name, username, and email matching the information provided.

### Use Case #9 – Share a batch with a friend

The user navigates to the batch they would like to share and selects the “share batch” link. The system then prompts the user to enter the username of the friend they want to share the batch with. The system then sends the batch information to the friend user.

### Use Case #10 – User can delete their account

Users will need to have an account and be logged in. Users will need to click the “delete account” link to delete their account. The user will be prompted to verify they wish to permanently delete their account. Immediately after the user verifies they wish to delete their account they will be logged out of the system and their account removed.

### Use Case #11 – User can create a new batch

A user can create a new batch by entering the following information:

* + name batch
  + enter type of batch (Beer, wine, etc…)
  + enter description of batch (optional)
  + notes (optional)

At any time during the “create new batch” process the user can edit the name, type, or description of batch.

### Use Case #12 – Update an existing batch

Users will need to have an account and be logged in. Users will need to click the “edit batch” link to edit an existing batch. The user will be able to edit the following:

* User can edit type of batch
* User can edit description

### Use Case #13 – User can enter notes about batch

Users will need to have an account and be logged in. Users will need to click the “edit batch” link for an existing batch. Users can then click the “add notes” link to add a note to that batch.

### Use Case #14 – User can delete notes

Users will need to have an account and be logged in. Users will need to click the “edit batch” link for an existing batch. Users can then click the “delete notes” link to delete a note to that batch.

### Use Case #15 – User can view all their batches

Users will need to have an account and be logged in. Users can then select the “view all batches” link to view a list of the following information for all saved batches:

* + Date batch was entered
  + batch name
  + batch type (Beer, wine, etc…)
  + batch description

### Use Case #16 – User can enter bottled data

When a bath has been completed, the user will create bottle information before adding it to their inventory. This bottle data includes the number of bottles, volume of the bottle and bottling date. The use can also set an alert to indicate when the bottles are ready for consumption.

### Use Case #17 – User can delete bottle from inventory

As a the user consumes bottles(or perhaps breaks them), they will need to remove them from their inventory. They will simply select which bottle they consumed. The consumption date will be record.

### Use Case #18 – Calculate alcohol percentage

While working on a batch, the user may want to calculate the current alcohol by volume. They will use their hydrometer to take a reading of the specific gravity and then add this measurement. Based on the original gravity they input at the start of the batch, the ABV will be calculated and stored with the date.

### Use Case #19 – Calculate sulfites in wine

While working on a batch, the user may want to update the sulfite(SO2) information for the batch. They will take a sulfite reading using measurement equipment and enter the read into they system. The reading will be time stamped.

### Use Case #20 – Calculate specific gravity based on temperature

While updating a batch, the user may take a specific gravity reading and need to adjust it based on temperature of the batch. They will enter the reading from their hydrometer and also the temperature of the batch. The system will then adjust the reading based on the temperature and store the adjusted value with a timestamp.

### Use Case #21 – Update a batch with new measurements and reading, SG, acidity, ext

The user has taken new measurements and wants to add them to their batch. They will select the batch in the system and then choose update. From here, they will choose to add a measurement. They will select the type of measurement and enter the value. The system will record the measurement with a timestamp.

### Use Case #22 – View trends of measurements over time, ex: how is acidity changing?

The user may want to see how various measurements have been changing over time. They will select a batch and then select view. From here, they will choose measurements and will be shown a table or graph of how the measurements have been changing over time. This will allow them to see how the batch is progressing and spot any negative trends.

### Use Case #23 – Add actions to a batch, for example when they racked, bottled, filtered ect.

While working on a batch, the user will need to record what actions they have performed. They will select the batch and then choose update. From here, they will choose the action performed and add notes. The action will be record by the system along with a timestamp.

### Use Case #24 – View actions performed on a batch

The user may want to look up what actions they have performed on a batch and when. They will select the batch and then choose view. From here, they will choose to view the action history of the batch. Each action will be displayed in chronological order along with any notes.

### Use Case #25 – User can delete batch

When a user is done with a batch and no longer wants the batch to exist, the user will select to delete the batch. Next the system will confirm that the user does in fact want to delete the batch, via a pop-up confirmation box, and the user will confirm.

### Use Case #26 – User can add any ingredients to a batch

A user will add many ingredients to a batch. For every ingredient, there will be specific attributes the users will add. Examples of what would be required for each ingredient include:

* Name
* Type
* Quantity
* Supplier
* Notes

### Once the ingredient attributes have been input, the user saves the ingredient and its attributes

### Use Case #27 – User can remove ingredients from a batch

When a user no longer wants certain ingredients in a batch, they have the ability to delete specific ingredients from the batch. They check the ingredients they want to remove and then click the remove ingredients button. Next the system displays a confirmation box, via a pop-up, asking the user if they are sure they want to remove the ingredients. If the user agrees, they will confirm their selection by clicking the OK button.

### Use Case #28 – User can view their inventory

User can inventory amount and type (ex. 12 bottles, 4 gallons)

A user selects to view their inventory. Upon the selection of the view inventory page, the inventory displays and the user can see the various items in their inventory. This will help the user decide whether or not they have too many or too few ingredients.

### Use Case #29 – Add recipes

At a certain point in time, the user will add a new recipe. Once he figures out he wants to add a recipe, they complete the associated attributes of the recipe. Once the recipe and its applicable attributes have been added, the user saves the new recipe

### Use Case #30 – View Recipes

A user clicks the view recipes button. The recipes which have already been created are then displayed on the screen.

### Use Case #31 – Delete recipe

When a user no longer wants certain recipes, they have the ability to delete specific recipes, by selecting remove recipe. Next the system, via a pop-up box will ask if the user does in fact want to delete the recipe. If the user agrees, they will confirm their selection by clicking the OK button.

### Use Case #32 – Start a new batch from a recipe

A user wants to start a new batch. One way he or she can start a batch is by using other recipes to help create the new batch. She selects previously created recipes to incorporate to a new batch. Once the new recipe is in the new batch, the user clicks the Save Batch button.

### Use Case #33 – Add reminder

* [if reminder on batch]

The user will select the reminders option. The reminders form will be presented to the user with the batch option automatically selected. The user will enter the reminder date and time along with a description and optional notes. When complete, the user will submit their data to the server. The reminder will then be associated with the batch and show on the general reminders screen. If the reminder is cancelled, any entered data will not be saved.

* [if general reminder]

The user will select the reminders screen. They will enter the reminder date and time along with a reminder description. They may optionally add notes to the reminder and associate a batch. When complete, the user will submit their data to the server. If the reminder is cancelled, any entered data will not be saved.

### Use Case #34 – Add batch to cellar inventory

The user navigates to the inventory screen and selects the Add Batch option. The system then prompts the user to enter the bottling date, quantity, and volume. When complete, the user submits the information and the system prompts them to let them know their information has been recorded.

* Bottling date
* Bottle quantity
* Bottle volume

### Use Case #35 – Search inventory

The user navigates to the inventory screen and selects the search option. They then enter their search criteria, such as bottling date range, quantity available, or volume available. The user then selects the search action, which searches the available inventory using the criteria specified. If no search criteria has been specified, then the search action will not be active.

### Use Case #36 – Search ingredients

The user navigates to the ingredients screen and selects the search option. They then enter their search criteria, such as hop or grape type, malt type, etc. For instance, “malt” would show all malts while “Belgian pilsner malt” would only show Belgian pilsner malts. When submitted, the system shows the user a list of all matching ingredients found. If no search criteria has been entered, then the search action will not be active.

### Use Case #37 – Share batch information

The user navigates to the batch they would like to share and selects the share action. The system then asks the user for the email address or user name of the user they would like to share with. If a user name is selected, then the system will send an email to the user the batch has been shared with notifying them of the shared batch. If an email address is used, then an email will be sent to the specified email address notifying the user that the batch has been shared with them. If the user is already a user of the system, then they will see the batch the next time they log in. Otherwise, the user must sign up before they can see the shared batch.

### Use Case #38 – Rate batch

The user navigates to the batch they would like to rate. The user selects the rate option and then rates the batch on a scale of 0 to 100. When finished, they submit their rating, which records their value and associated it with the selected batch.

### Use Case #39 – Comment on batch

The user navigates to the batch they would like to comment on. The user selects the comment option and enters their comment, which must be between 1 and 256 characters in length. When complete, the user submits their comment, which associates it with the selected batch.

## 3.4 Classes / Objects

### 3.4.1 <Class / Object #1>

3.4.1.1 Attributes

3.4.1.2 Functions

<Reference to functional requirements and/or use cases>

### 3.4.2 <Class / Object #2>

…

## 3.5 Non-Functional Requirements

Non-functional requirements may exist for the following attributes. Often these requirements must be achieved at a system-wide level rather than at a unit level. State the requirements in the following sections in measurable terms (e.g., 95% of transaction shall be processed in less than a second, system downtime may not exceed 1 minute per day, > 30 day MTBF value, etc).

### 3.5.1 Performance

### 3.5.2 Reliability

### 3.5.3 Availability

### 3.5.4 Security

### 3.5.5 Maintainability

### 3.5.6 Portability

## 3.6 Inverse Requirements

State any \*useful\* inverse requirements.

## 3.7 Design Constraints

Specify design constrains imposed by other standards, company policies, hardware limitation, etc. that will impact this software project.

## 3.8 Logical Database Requirements

Will a database be used? If so, what logical requirements exist for data formats, storage capabilities, data retention, data integrity, etc.

## 3.9 Other Requirements

Catchall section for any additional requirements.

# 4. Analysis Models

List all analysis models used in developing specific requirements previously given in this SRS. Each model should include an introduction and a narrative description. Furthermore, each model should be traceable the SRS’s requirements.

## 4.1 Sequence Diagrams

## 4.3 Data Flow Diagrams (DFD)

## 4.2 State-Transition Diagrams (STD)

# 5. Change Management Process

Identify and describe the process that will be used to update the SRS, as needed, when project scope or requirements change. Who can submit changes and by what means, and how will these changes be approved.

# A. Appendices

Appendices may be used to provide additional (and hopefully helpful) information. If present, the SRS should explicitly state whether the information contained within an appendix is to be considered as a part of the SRS’s overall set of requirements.

*Example Appendices could include (initial) conceptual documents for the software project, marketing materials, minutes of meetings with the customer(s), etc.*

## A.1 Appendix 1

## A.2 Appendix 2