
Architecture Design

for

Brew Day!

Version ~~2~~.0 approved

Deleted: 1

Prepared by Zhenghao Wu, Anqin Zha, Renjie Deng and Ruichao
Zhong

Dijkstra

Wednesday, May 22, 2019,

Deleted: Tuesday, April 2, 2019

Deleted: <Project>

Table of Contents

Table of Contents **ii**

Revision History **ii**

1. Overview **1**

 1.1 Project description 1

 1.2 References 1

 Design purpose 1

 1.3 1

2. Overall description **1**

 2.1 Use case diagram and class diagram 1

 2.2 Design model 3

 2.3 System architecture 3

3. System architecture **5**

 3.1 Brew Subsystem 5

 3.1.1 Description 6

 3.1.2 Database 6

 3.2 Ingredients Subsystem 7

 3.2.1 Description 7

 3.2.2 Database 7

 3.3 Note Subsystem 8

 3.3.1 Description 8

 3.3.2 Database 8

 3.4 Equipment Subsystem 9

 3.4.1 Description 9

 3.4.2 Database 9

4. Assessment **10**

 4.1 Stability 10

 4.2 Reusability 10

 4.3 Scalability 10

5. Alternative design (optional) **10**

6. More considerations **10**

7. Appendix **11**

Revision History

Name	Date	Reason For Changes	Version
Zhenghao Wu, Anqin Zha, Renjie Deng and Ruichao Zhong	2019-04-02	The first version of the Architecture Design	1.0
<u>Zhenghao Wu, Anqin Zha, Renjie Deng and Ruichao Zhong</u>	<u>2019-05-22</u>	<u>Update subsystem architecture; Update the class diagram to the latest one.</u>	<u>2.0</u>

Deleted: Table of Contents-**ii**

Revision History-**ii**

1. Overview-**1**

1.1 Project description-**1**

1.2 References-**1**

1.3 Design purpose-**1**

2. Overall description-**1**

2.1 Use case diagram and class diagram-**1**

2.2 Design model-**3**

2.3 System architecture-**3**

3. System architecture-**3**

3.1 Brew Subsystem-**3**

3.1.1 Description-**3**

3.1.2 Database-**3**

3.2 Ingredients Management Subsystem-**3**

3.2.1 Description-**3**

3.2.2 Database-**3**

3.3 Note Subsystem-**3**

3.3.1 Description-**3**

3.3.2 Database-**3**

3.4 Equipment Management Subsystem-**4**

3.4.1 Description-**4**

3.4.2 Database-**4**

4. Assessment-**4**

4.1 Stability-**4**

4.2 Reusability-**4**

4.3 Scalability-**4**

5. Alternative design (optional)-**4**

6. More considerations-**5**

7. Appendix-**5**

Deleted:

1. Overview

1.1 Project description

This project is to develop a desktop-base for home brewer to brew beers. Home brewer can use the product to maintain recipe, ingredients and equipment information, he can also receive recommendation from this software and take notes on each brewing.

1.2 References

Z. H. Wu, et al., "Software Requirements Specification for Brew Day! v6.0" May 22nd 2019.

1.3 Design purpose

This design is given to divide system into several subsystem with loose coupling and high cohesion, thus the large system can be separate into different task that each single programmer can do their jobs independently.

2. Overall description

2.1 Use case diagram and class diagram

The Use Case Diagram shown below:

Deleted: Apr

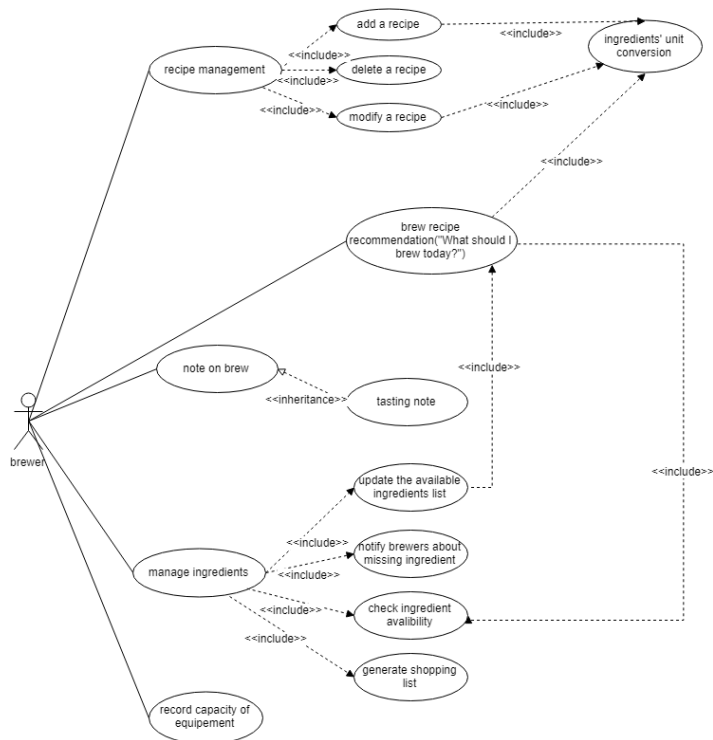
Deleted: 1

Deleted: st

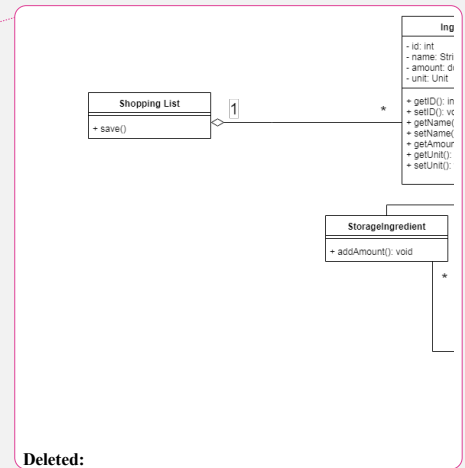
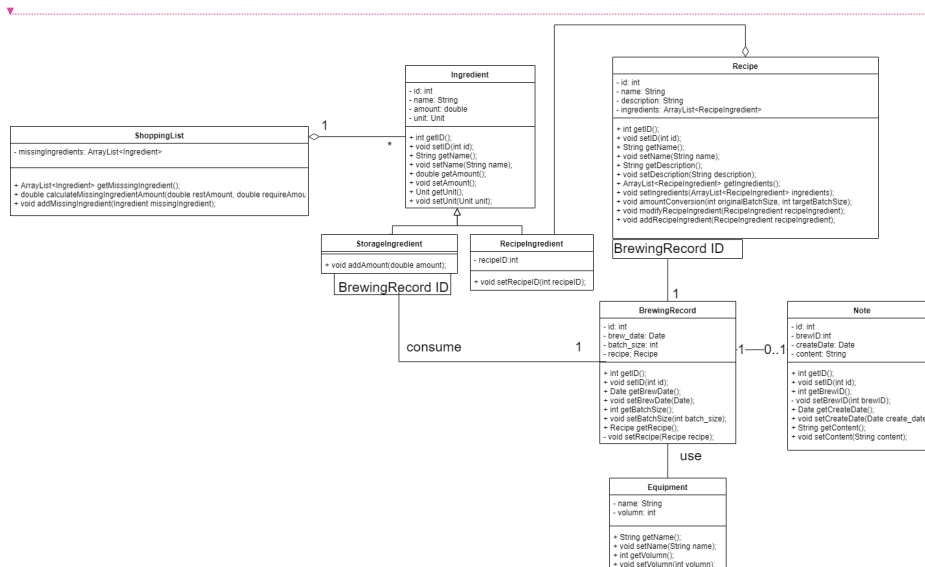
Formatted: Heading 2

Deleted: ¶
<Describe why this design is given, e.g., its purpose and advantages.>

Deleted: <Put the use case diagram and class diagram here>¶



The class diagram shown below:



Deleted:

2.2 Design model

This system is using MVC three tier architecture. This architecture model is used in there is because this model can help to divide the system in to different part (UI, application logic, storage) and help different developer to develop it separately.

Deleted: <Describe the system architecture model used in this design and why this model is chosen>

Deleted: ¶

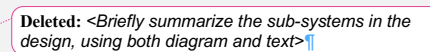
Formatted: Font: Not Italic

Deleted:

2.3 System architecture

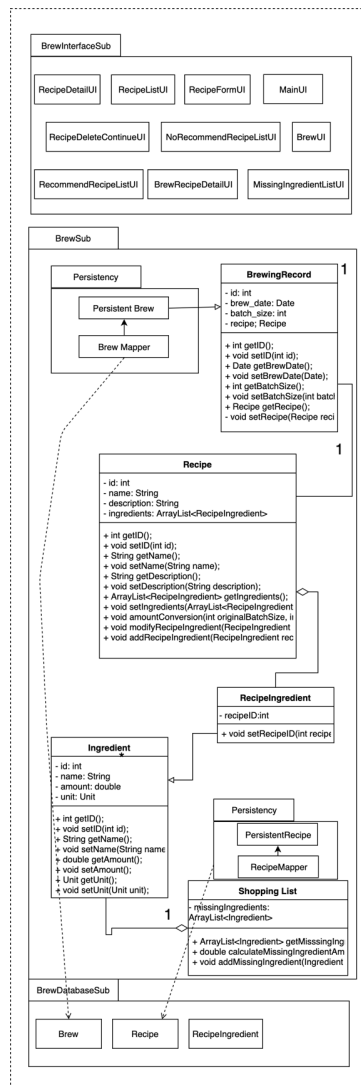
This system is divided into four subsystems: Brew Subsystem, Ingredient Subsystem, Note Subsystem, and Equipment Subsystem. Diagram that indicate the relationship between subsystems is shown below:

Formatted: template



3. System architecture

3.1 Brew Subsystem



Deleted: ¶

Deleted:

Deleted: <This section is used to describe each component in the system>¶

Formatted: Normal

3.1.1 Description

The brew subsystem is consisting of three part: BrewInterfaceSub is for UI, it consist of all boundary objects including: MainUI for the homepage of the software; RecipeDetailUI, RecipeListUI, RecipeFormUI, RecipeDeleteContinueUI for Recipe use case; BrewUI, RecommendRecipeListUI, NoRecommendRecipeListUI, BrewRecipeDetailUI, MissingIngredientListUI for recommend a brew use case. BrewSub is for application logic. It handle control and entity objects. Brew, Recipe, Ingredient, RecipeIngredient, ShoppingList are class in this part. And Brew and Recipe are the persistent data. BrewDatabaseSub is for storage, it storage, retrieval, and query of persistent data.

Deleted: <Briefly describe the components in this subsystem and put the architecture diagram here>¶

Deleted: <Describe each component in this subsystem >

Formatted: Font: 12 pt, Not Italic

Formatted: Font: 12 pt

Formatted: Font: 12 pt, Not Italic

3.1.2 Database

There are three database tables in the brew subsystem: Brew, Recipe, Ingredient in Recipe. Brew table is uses to record brew records. Recipe table is uses to record recipes. Ingredient in Recipe is records different kinds of and amount of ingredient in different recipe. The fields for each table are list below:

Formatted: Font: (Default) Arial

Brew				
Brew ID	Brew Date	Batch size	Recipe ID	Note ID

Recipe		
Recipe ID	Name	Description

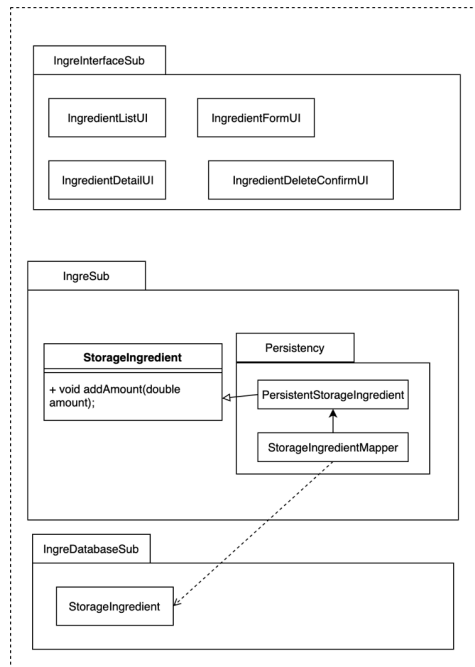
Formatted Table

Ingredient in Recipe			
Recipe ID	Ingredient ID	Amount	unit

Formatted Table

Formatted: Normal

3.2 Ingredients Subsystem



3.2.1 Description

The system contains 3 parts, Interface part, application logic part and database part. The interface part of the system contains Ingredient List, Ingredient Form, Ingredient Detail and confirm delete window. The ingredient list lists all the information about ingredients in the warehouse including the name of ingredients and amount of ingredients. Ingredient form is a form that you can add a new ingredient to the warehouse or modify the amount of ingredients in the warehouse. Ingredient detail page show you the detail information of an ingredient including name, amount and unit of the ingredient. IngredientConfirmDelete page is a page that ensure you would not delete the ingredient by accident. The data of StorageIngredient class is persistent, which will be store in database. The application logic part of the system contains StorageIngredient. With StorageIngredient part, you can manage the amount of ingredients in the warehouse. The database part contains a table called StorageIngredient, which store the data of the amount of ingredients in the warehouse.

3.2.2 Database

Deleted: <Describe the tables in database if exists>

Deleted: Management

Formatted: Normal

Deleted: <Briefly describe the components in this subsystem and put the architecture diagram here>

Formatted: Font: 12 pt, Not Italic

Formatted: Font: 12 pt, Not Italic

Formatted: Font: 12 pt, Not Italic

Formatted: Font: Not Italic

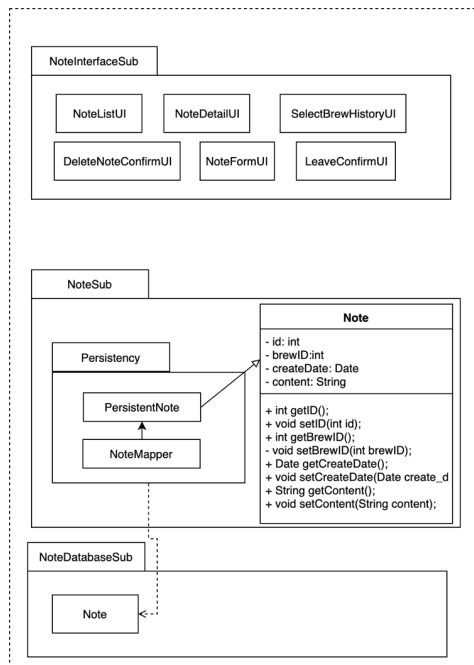
Deleted: <Describe each component in this subsystem >

Formatted: Font: 12 pt, Not Italic

There is one database table in the note subsystem: Note. Note table is uses to record brewer's note for a brewing record. The fields for the table is list below:

Ingredient			
Ingredient ID	Name	Amount	Unit

3.3 Note Subsystem



3.3.1 Description

The subsystem contains three part: Interface part, application logic and storage part. Among them, all component in interface part are for displaying the note list and brewing history list, the content of note, inputting interface and some confirm window to deal with interaction with users. For the application logic, the component "note" is mainly for setting and getting the ID, date and content of the notes, and the data of "note" is persistent, and then it will store its persistent data to the component "note" in storage part, which is the third part of this subsystem through the component "persistent Note". The "note" in storage database realize the storage, retrieval, and query of the data it received.

3.3.2 Database

Formatted: Font: (Default) Arial

Deleted: <Describe the tables in database if exists>
.....

Deleted:
<Briefly describe the components in this subsystem and put the architecture diagram here>

Formatted: Font: Not Bold

Deleted: <Describe each component in this subsystem

Formatted: Font: 12 pt, Not Italic

Deleted: >

Formatted: Font: 12 pt, Not Italic

There is one database table in the note subsystem: Note. Note table is uses to record brewer's note for a brewing record. The fields for the table is list below:

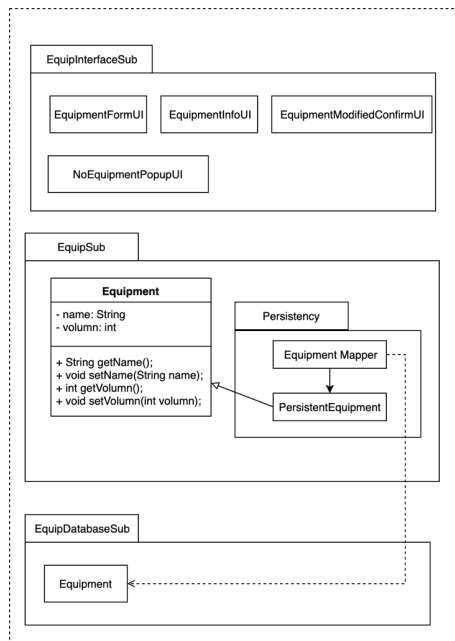
<u>Note</u>			
<u>Note ID</u>	<u>Create date</u>	<u>Content</u>	<u>Brew ID</u>

Formatted: Font: (Default) Arial

3.4 Equipment Subsystem

Deleted: <Describe the tables in database if exists>

Deleted: Management



3.4.1 Description

The subsystem contains three part: Interface, application logic and storage. Among them, all component in interface part are for displaying Equipment Form, Equipment information and some confirm information. For the application logic part, the data of the component "equipment" is persistent, the component itself is mainly used for setting and getting the name and volume of the equipment, once the setting is done, it will store data to "Equipment" in the storage part, which is the third part of this subsystem, through the component "persistent Equipment". The "Equipment" in storage database realize the storage, retrieval, and query of the data it received.

Deleted:

<Briefly describe the components in this subsystem and put the architecture diagram here>

Formatted: Font: Not Bold

Deleted: <Describe each component in this subsystem >

Formatted: Font: 12 pt, Not Italic

Formatted: Font: 12 pt, Not Italic

3.4.2 Database

There is one database table in the equipment subsystem: Equipment. Equipment table is used to record brewer's equipment for brewing. The fields for the table are listed below:

Equipment		
Equipment ID	Name	Volume

4. Assessment

4.1 Stability

The system based on this architecture is stable, because this system has minimized the associations which will cross subsystem boundaries, that is, the coupling of this system is loose and the cohesion of this system is high, therefore, if some changes are made to certain components, we need only to change very few objects' interface.

4.2 Reusability

The component of this system can easily be used in other places without changes, because three out of four subsystems can implement their jobs individually: taking notes, maintain equipment and maintain ingredients. They will only contact the brewing subsystem when they are requested data from it. What is more, the brewing subsystem needs only data instead of function of other subsystem, which means all components of this system have high cohesion.

4.3 Scalability

This system is easy to extend with this architecture because this system adapted divide and conquer strategy, which solves the complexity of the system. In other words, each component has the clear input and output requirement, thus it is easy to expand the architecture.

5. Alternative design (optional)

N/A

6. More considerations

For more detail about the UI design, please refer to the Software Requirement Specification v6.4.1 – User Interface

Formatted: Font: (Default) Arial

Deleted: <Describe the tables in database if exists>¶

Deleted: <This part assesses the current architecture from several perspectives>¶

Deleted: <Is the system based on this architecture stable? For example, if some changes are made to certain components, how many parts will be influenced? This is to check the coupling and cohesion>¶

Formatted: Font: 12 pt

Deleted: <Can the components of the system be used in other places without changes? Give the explanations>¶

Formatted: Font: 12 pt

Deleted: <Is the system easy to extend with this architecture? Give the explanations>¶

Formatted: Font: 12 pt

Deleted: <In case some special reasons that make the above design infeasible, an alternative is ready. Give the brief description and the reference design document here>¶

Formatted: Font: 12 pt

Formatted: Font: 12 pt

Deleted: <The designer can give readers considerations and suggestion here>¶
N/A

Formatted: Font: 12 pt, Not Italic

Formatted: Font: 12 pt

Formatted: Font: 12 pt

7. Appendix

N/A

Deleted: <Put any appendix here>¶
Formatted: Font: 12 pt