**Software Engineering II Project**

**Digital Cookbook**

Group 4

Gu Mingzhe

Huang Mingde

Yang Rui

Chen Chang

**Content**

**[1.](#_gjdgxs)****Specification** 3

**[1.1.](#_30j0zll)****Description** 3

**[1.2.](#_1fob9te)****Product functions** 3

**[1.3.](#_3znysh7)****User characteristics** 3

**[1.4.](#_2et92p0)****Data** 3

**[1.5.](#_tyjcwt)****Functional requirements** 3

**[1.6.](#_3dy6vkm)****Non-functional requirements** 3

**[2.](#_1t3h5sf)****UML Specification** 4

**[2.1.](#_4d34og8)****Class Diagram** 4

**[3.](#_2s8eyo1)****GUI Design** 5

**[3.1.](#_17dp8vu)****Structure/MVC** 5

**[3.2.](#_3rdcrjn)****Screenshots** 5

**[4.](#_26in1rg)****Evaluation** 6

**[4.1.](#_lnxbz9)****Group Work** 6

**[4.2.](#_35nkun2)****Task Responsibilities** 6

# **Specification**

# **Description**

Welcome to the Customizable Recipes Cookbook! While our cookbook provides numerous recipes out of the box, we know that there are many more ideas out there. This application provides with some of the recipes we've collected from you! Please know that this is just a small sampling and we are always looking for more ideas. So, if you cooked up a great Customizable Recipe in your org, and you want to share, you can freely upload your own ideas to our cookbook!

# **Product functions**

Several recipes are provided with decent pictures and informative contents

* Search for recipes in favour by name
* Add your favourite recipes to the favourite list
* Rate recipes with the scale of 5

# **User characteristics**

* Novice Cookers
* Master Cookers who want to share

# **Data**

* Recipe Name
* Recipe Image
* Prep Time
* Cook Time
* Serve
* Ingredients
* Instructions
* Rating Score
* Favourite Recipe List

# **Functional requirements**

* Create/ Delete/ Edit Recipe
* Illustrate the selected recipe
* Synchronize Recipe to the database
* Search for a recipe by name
* User can rate recipes in the rating scale of 5
* User can add the desired recipe to their own favourite list
* The amount of ingredients will be automatically modified according to the changing of the serve number without modifying the row data in the database

# **Non-functional requirements**

* The system is required to be used by 100 people at the same time.
* The user interface should make a response within 2 seconds.
* The search result should be shown within 2 seconds.

# **UML Specification**

# **Class Diagram**

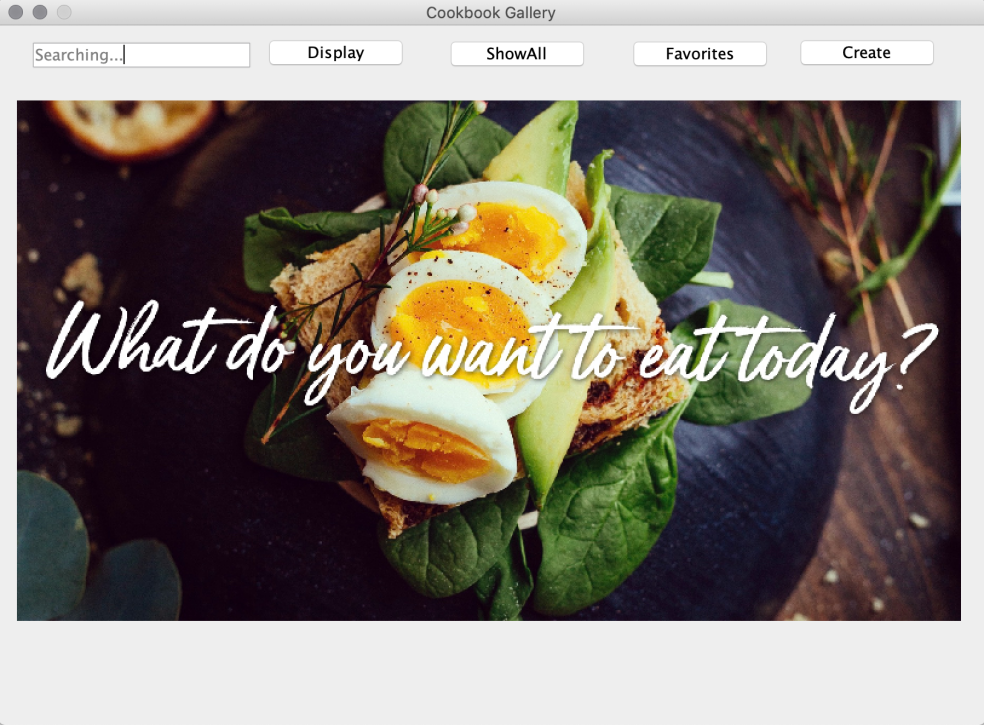
# **GUI Design**

# **Structure/MVC**

* Model: model is used to store entity classes in our project. The Recipe model and Ingredient model are corresponding to the recipe table and ingredients respectively. Each object of an entity maps one or several rows of data in database. The Recipes model stores a list of Recipe instances. Data can be transferred as a whole from the view to the database or vice verse.
* Controller: Each window or dialog has its own controller in our project, and these controllers all interact with database controller, including Gallery window controller, Detail dialog controller and Create dialog controller. Database controller directly operates with database. All the operations including create, delete, update and retrieve are encapsulated in database controller. When a request is sent by a view class, its own controller is first be activated and it then call the methods in database controller, after the corresponding operations are done in the database, as well as some data are transferred from database if required, its own controller then update the view interface, thus, to show the updating data to the user. In this process, data are transferred using entity class in model to reduce the trouble of transferring their attributes one by one.
* View: In our project, we design three view interfaces (Gallery window, Detail dialog, Create dialog). These three interfaces are used to show data to users and response to user’s request. Also, when some buttons are clicked or mice take some actions, view will send messages to controller to request updating.

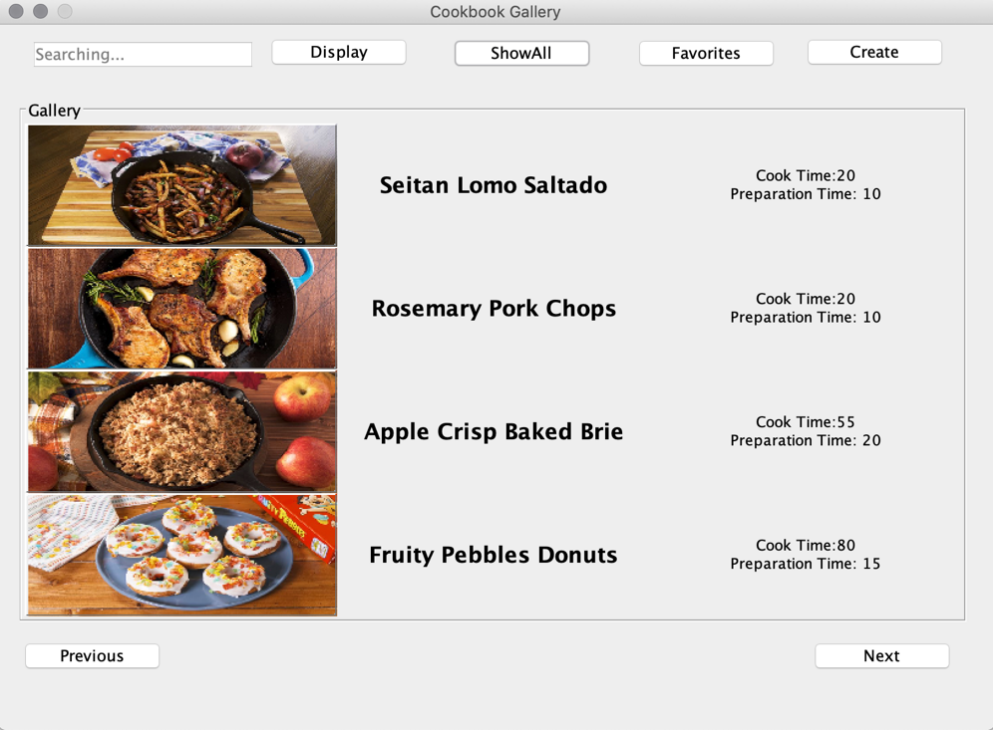
# **Screenshots**

**Welcome page**



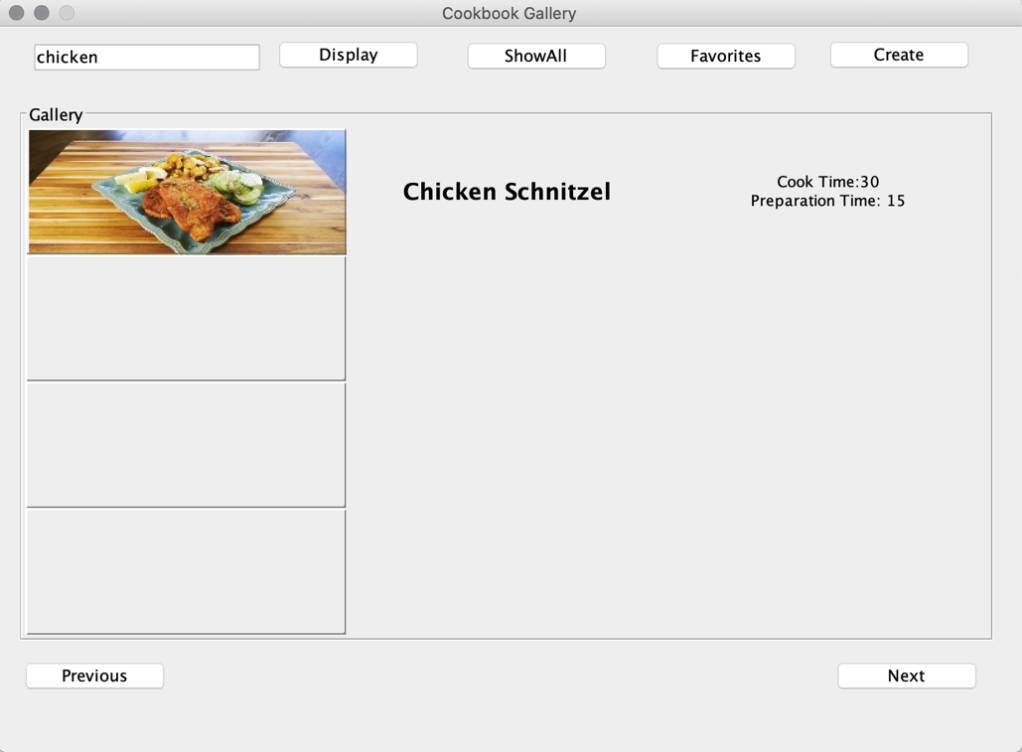
**Gallery page (triggered by “ShowAll” button)**

Illustrate all the recipes stored in the Cookbook.

****

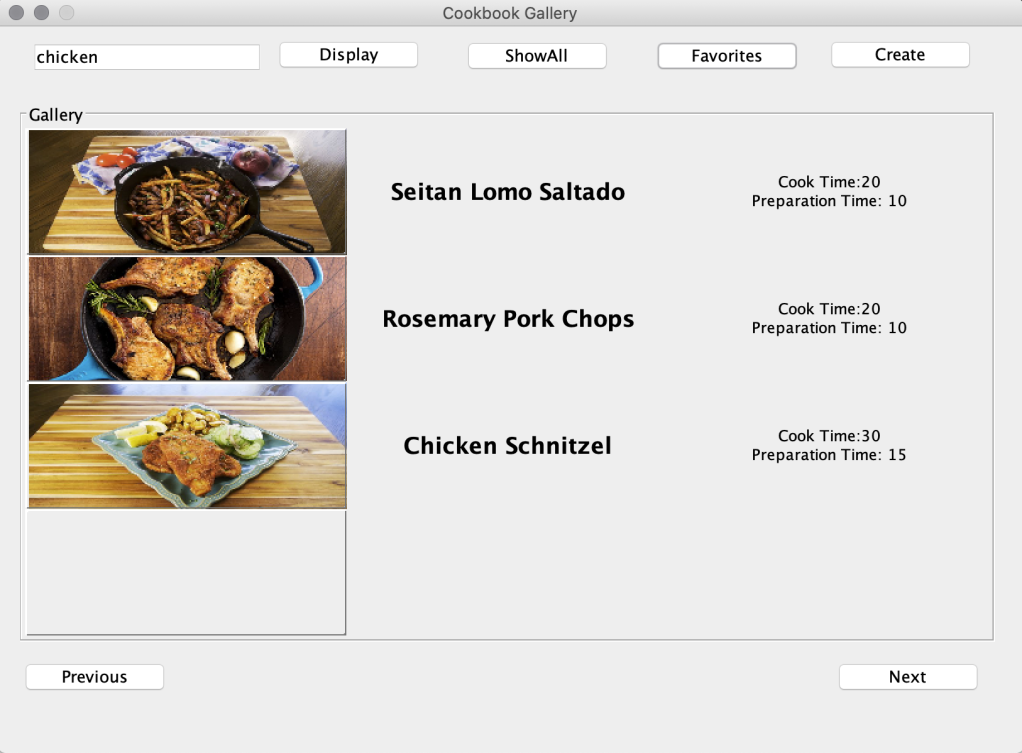
**Search result page (triggered by “Display” button or ENTER with keywords inserted)**

Show the result recipe of the searching.

****

**Favourite list page (triggered by “Favourites” button)**

Show all the recipes which is marked as “favourite”.

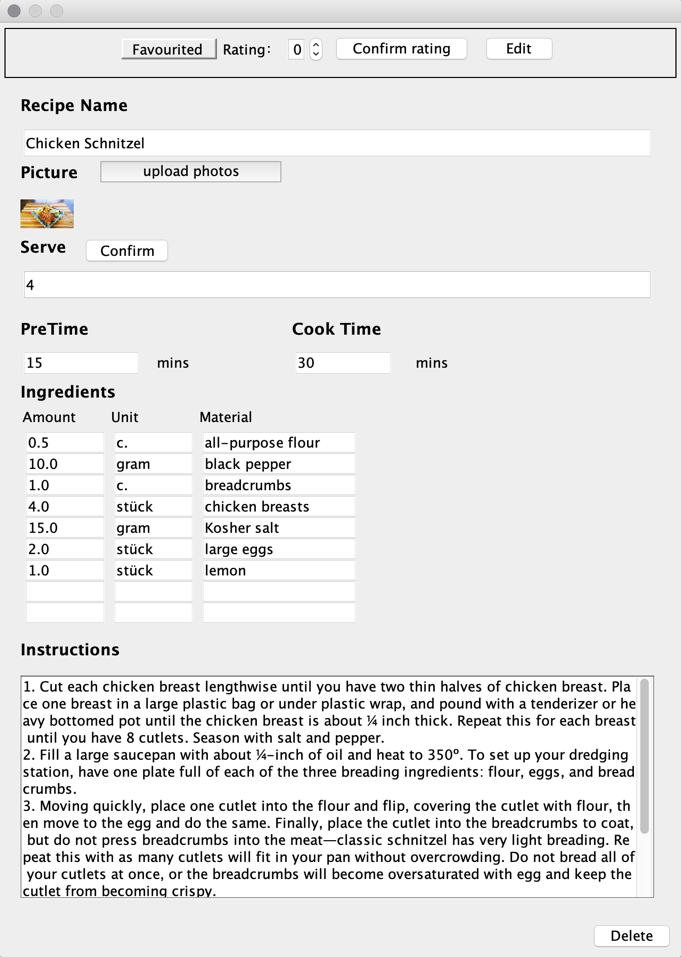
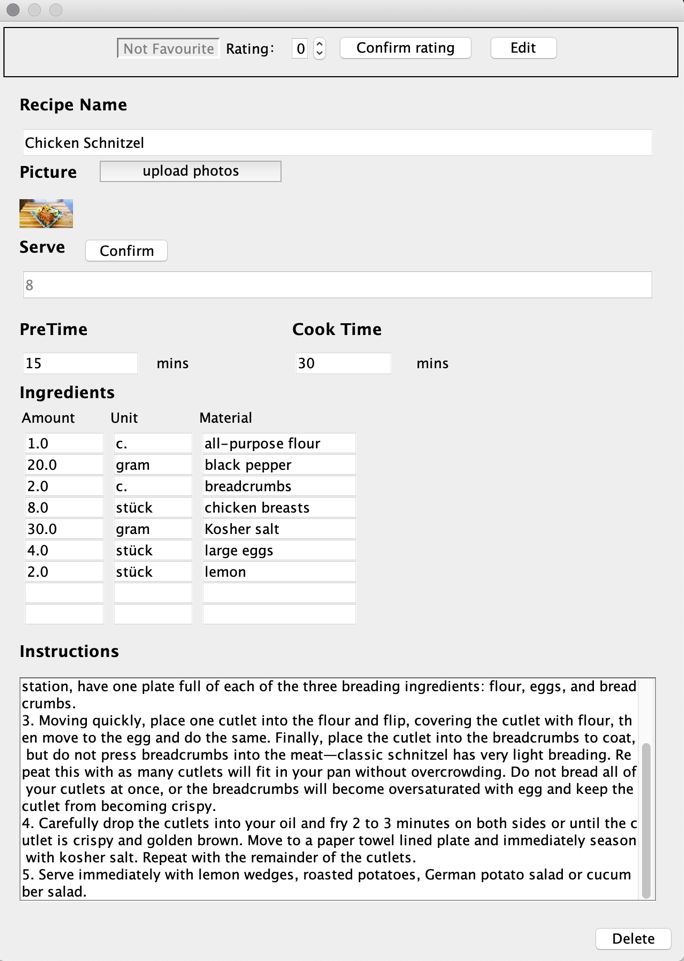
****

**Default detailed recipe page (triggered by clicking the image in the gallery page)**

Show the detailed information of the recipe.

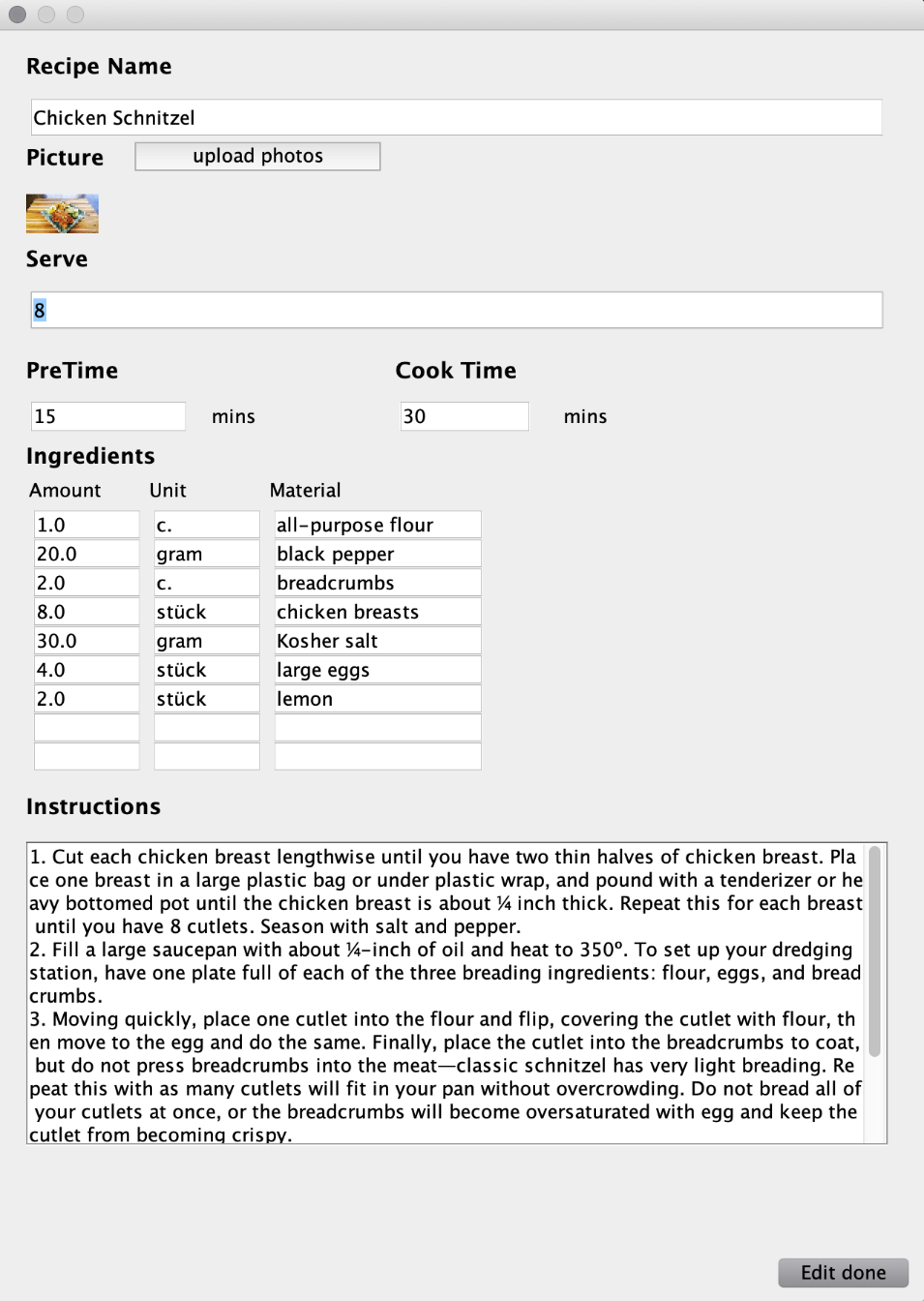
The default setting of this page is that all the literal information on this page is not editable with one exception “Serve”. The users can change the value of “Serve” and press the “Confirm” button. Then, all the amount of ingredients will change corresponding to the change of “Serve” value. It greatly helps users get the right amount of ingredients of the desired serve number without calculating by themselves.

The users can mark the recipe as Favourited and give rate in this page. The appearance of “Favourited” button will be automatically changed under two different conditions.

** **

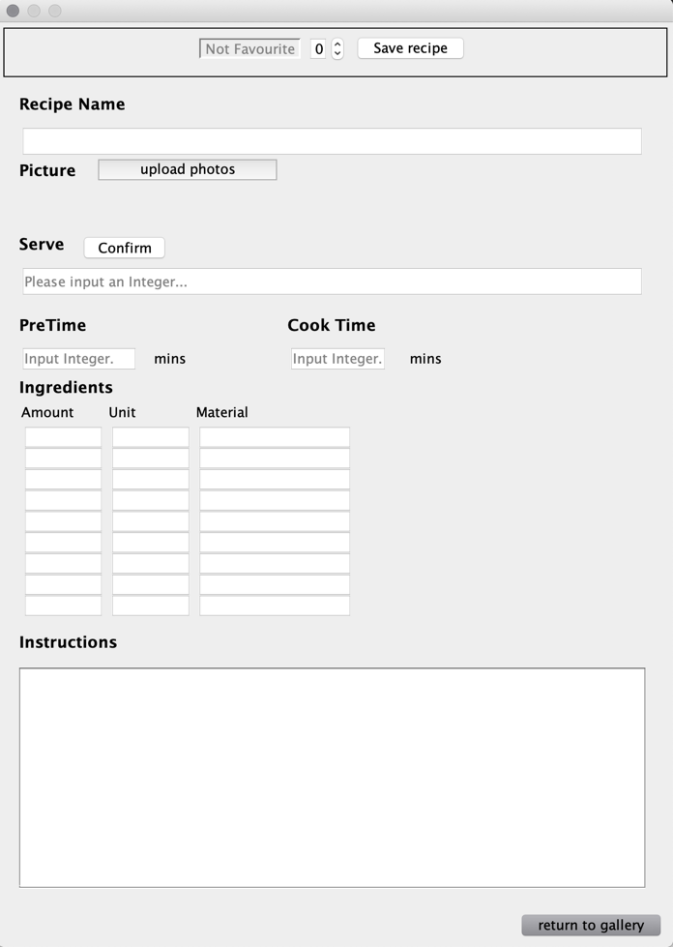
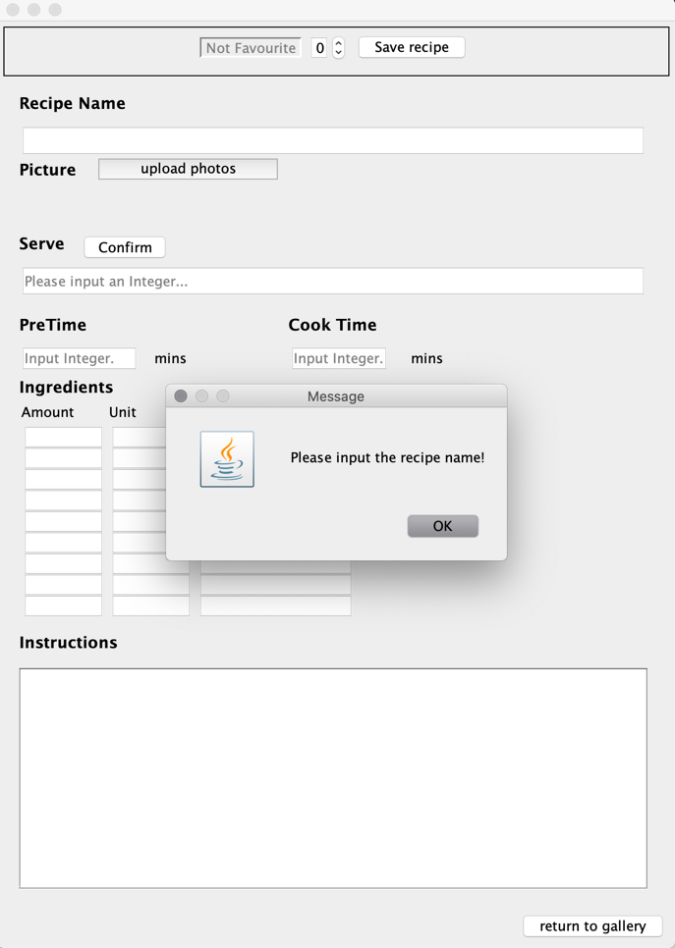
**Editable detailed recipe page (triggered by clicking the “Edit” button in the default detailed recipe page)**

All the information of the recipe is editable now. With clicking of “Edit done” button. The data will be updated in the database.

****

**Create recipe page (triggered by clicking the “Create” button in the gallery page)**

Users can create their own recipes with the given template. “Serve”, “PreTime”, “Cook Time” and “Amount” text areas are restricted to numbers only. All the information is necessary. Otherwise, the software will notice the user of the unfilled information. After clicking “Save recipe’ button, the recipe will be added to the database.

# **Evaluation**

# **Group Work**

* Sketch the draft of class diagram and determine the frame of the whole project.
* After the distributed work is done, connect each part to others and integrate as a whole.
* Do cross tests, debug, promote and optimize the details. Eliminate the bugs happened when the data was not correct by user’s input. Simplify the procedures to create/update/delete a recipe and add reminding windows for user. Improve the graphical appearance of the user interface.

# **Task Responsibilities**

UML Chen Chang, Yang Rui

GUI Design Chen Chang, Yang Rui

Model Design Gu Mingzhe

Database Design Huang Mingde

Testing Gu Mingzhe, Huang Mingde

Report Gu Mingzhe, Chen Chang, Huang Mingde, Yang Rui

User guide Gu Mingzhe