CS 3354 Software Engineering

Final Project Deliverable I

# Tastey

## Group Members

Dan Nguyen
Cade Edwards
Luke Buchanan
Matthew Singsing
Rayven Gsell
Shoham Sanyal

URL: https://github.com/DanNguyenN/3354-Tastey

## Final Project Draft Description

Cooking is an essential part of life, everyone does it. So, you have likely experienced the pain of having to search for recipes online. Dodging countless ads, annoying backstories about the recipe, and the hunt to find the 'jump to recipe' button. And on top of all of that, you cannot easily filter recipes based on your dietary restrictions or allergies. This application offers a solution, an easy way to manage your recipe book, quickly filter through it, and then give you steps on how to cook your meal with no interruptions.

This design is very beneficial for real-world applications since companies could pay to have advertisements featured in between recommended recipes, culinary influencers could upload their own recipes for users to download, and grocery stores could pay to have recipes or ingredients featured at specific times of the year since they generate large amounts of revenue through seasonal food sales.

As you can likely tell, this is a passion project. One we hope to deliver on and legitimately use in the future outside of this class. It is similar in concept to the books app that was given in the original project document, but it will remove some niche features and add more specialized ones for recipes like adding genres to sort food by and having tabs in the recipes.

**Instructor Feedback:**

Great project topic with a fringe benefit, as it will potentially help everyone as we all need to eat, and will definitely promote healthy home cooking.

In the final report, please make sure to include comparison with similar applications -if any- and make sure that you differentiate your design from those and explicitly specify how.

Please share this feedback with your group members.

You are good to go. Have fun with the project and hope everyone enjoys the collaboration.

**Project Feedback Response:**

In response to our instructor's feedback, we did a little more research on the different kinds of recipe apps out there. The two recipe apps we researched are SuperCook and Mealime. Our app aims to serve a more niche population as it varies from each of these existing apps. SuperCook is an app that finds recipes based on the ingredients you have at home. While this is useful, Tastey implements a search using the dish's name. Mealime is an app that helps users meal prep for the week. This app is mainly focused on people that may not have the time to cook every day. In comparison, Tastey's recipes feed anywhere from 3-5 people and are meant to be shared with friends and family.

The aim of Tastey is for it to be a place where users can search and add recipes of their favorite meals. They can and allergen information and dietary restriction so that their results can be tailored to what they can and cannot eat. This means that since the community can upload creative recipes users can find new and creative ways to bypass their dietary restrictions while still enjoying their favorite foods. For example, an egg-free cake, gluten-free pancakes, and vegetarian fried chicken (chicken substitute), are all recipes that could be uploaded to Tastey.

# Project Scope

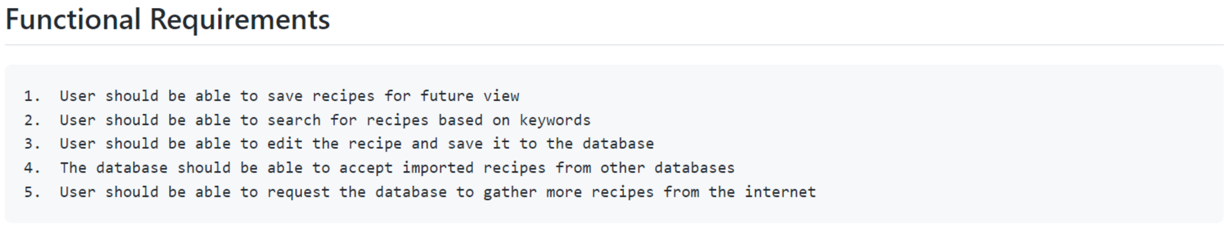
1. Tastey - A Recipe Library Software
   1.1. Recipe Management
      1.1.1. Load saved recipes
      1.1.2. Delete saved recipes
      1.1.3. Add your own recipes
      1.1.4. Search recipes by category
   1.2. Recipe Reading
      1.2.1. Ability to choose between tabs; ingredients, steps, stories, and photos
      1.2.2. Scroll to go between steps
      1.2.3. Adjust portion sizes
      1.2.4. Offer recipes/items that pair well with this one (Optional)
   1.3. Aesthetics/Accessibility
      1.3.1. Day and Night Mode (Optional)
      1.3.2. Change font and size of text (Optional)
      1.3.3. Adjust theme/color of the app (Optional)
      1.3.4. Choose allergies/dietary restrictions

## Software Process Model Choice

The software process model that is employed in our project is the waterfall model. This is because we felt that a more standard approach would be adequate for the project that we had in mind. From the beginning, we laid clear and well-thought-out ideas for what we wanted to achieve with our project. As a result, no iterations were necessary. Additionally, we created diagrams and documents that clearly state the specifications of our software.

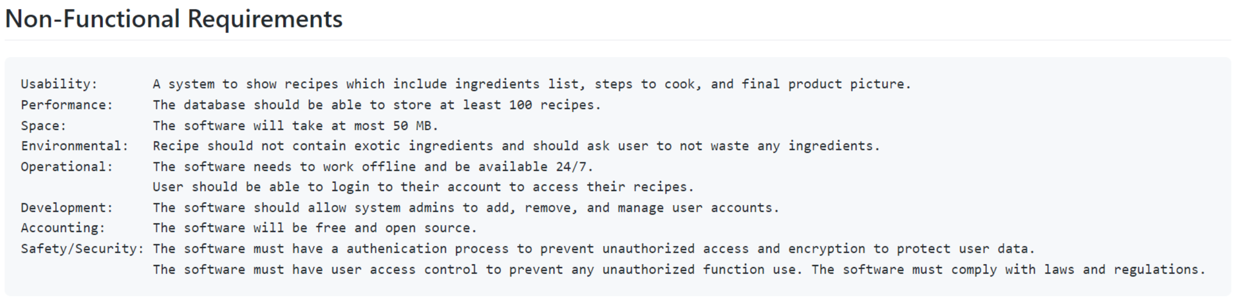
Other software models that we ended up not using were the incremental process model, the prototyping model, and the spiral model. We didn’t like the incremental process model because we felt that there were not enough parallel process flows that we could work on. Additionally, our time frame was rather short so only a few “increments” could be completed. The prototyping model wasn’t an option due to the fact that we were not at the stage of implementing our project. Therefore, there was no need to create multiple prototypes. Additionally, in order to build a working prototype sacrifices must be made and we as a team decided that we would not take those sacrifices to ensure that we delivered a premium product. The final software process model which we ruled out was the spiral model. The main reason for this is that we found it difficult to identify the framework activities that would represent the spiral path. Additionally, since no implementation was being done yet the spiral model seemed unnecessary for what we needed.

# Software Requirements



## Functional Requirements

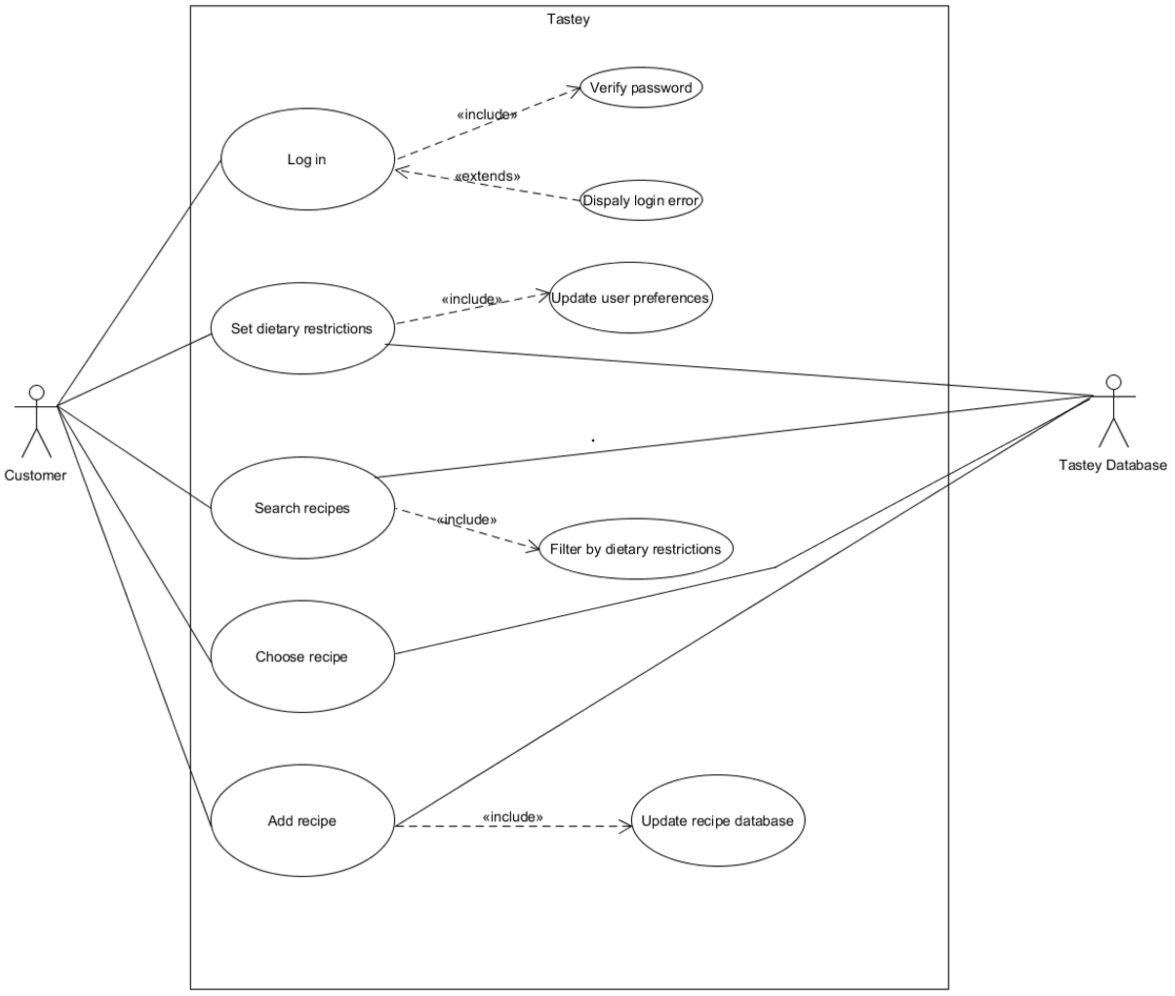
```
1.  User should be able to save recipes for future view
2.  User should be able to search for recipes based on keywords
3.  User should be able to edit the recipe and save it to the database
4.  The database should be able to accept imported recipes from other databases
5.  User should be able to request the database to gather more recipes from the internet
```



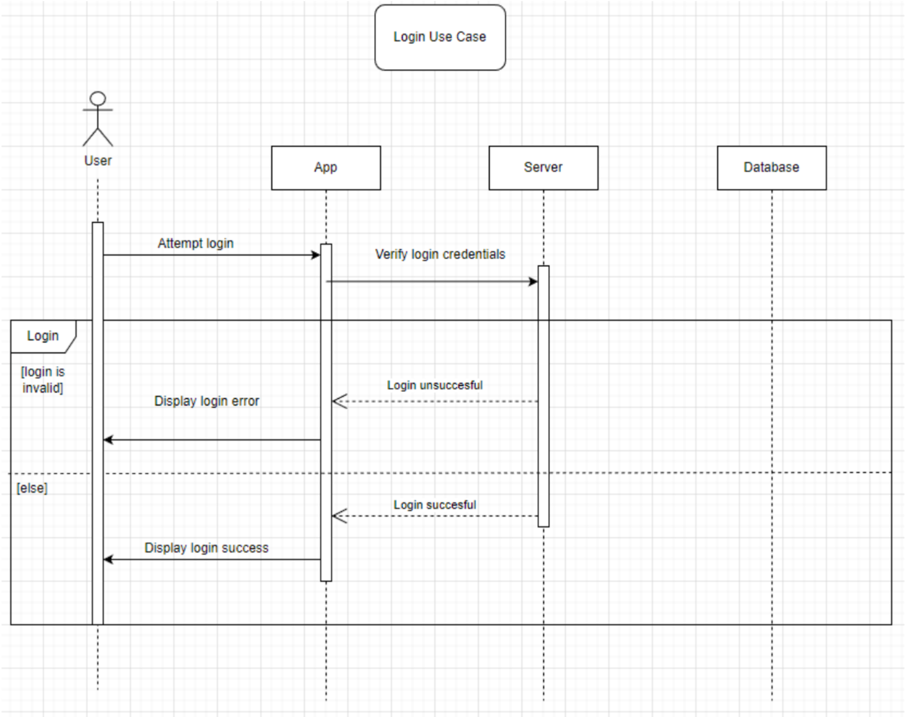
## Non-Functional Requirements

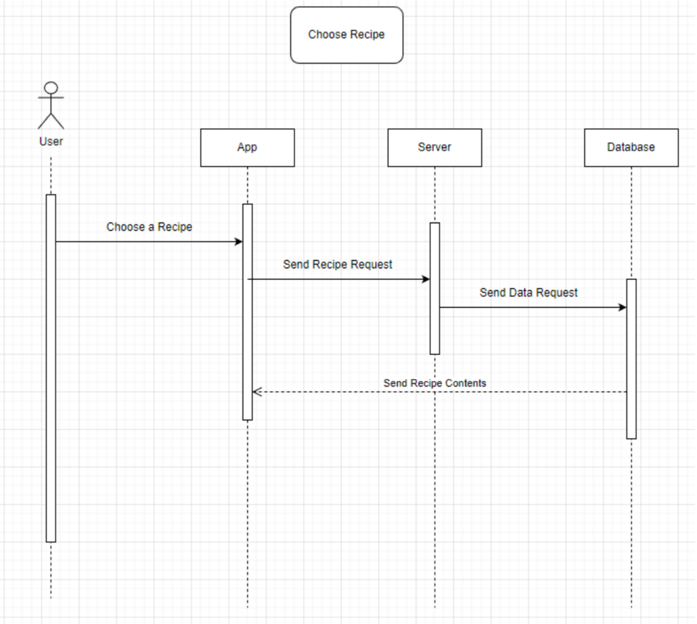
```
Usability:       A system to show recipes which include ingredients list, steps to cook, and final product picture.
Performance:     The database should be able to store at least 100 recipes.
Space:           The software will take at most 50 MB.
Environmental:   Recipe should not contain exotic ingredients and should ask user to not waste any ingredients.
Operational:     The software needs to work offline and be available 24/7.
                 User should be able to login to their account to access their recipes.
Development:     The software should allow system admins to add, remove, and manage user accounts.
Accounting:      The software will be free and open source.
Safety/Security: The software must have a authenication process to prevent unauthorized access and encryption to protect user data.
                 The software must have user access control to prevent any unauthorized function use. The software must comply with laws and regulations.
```

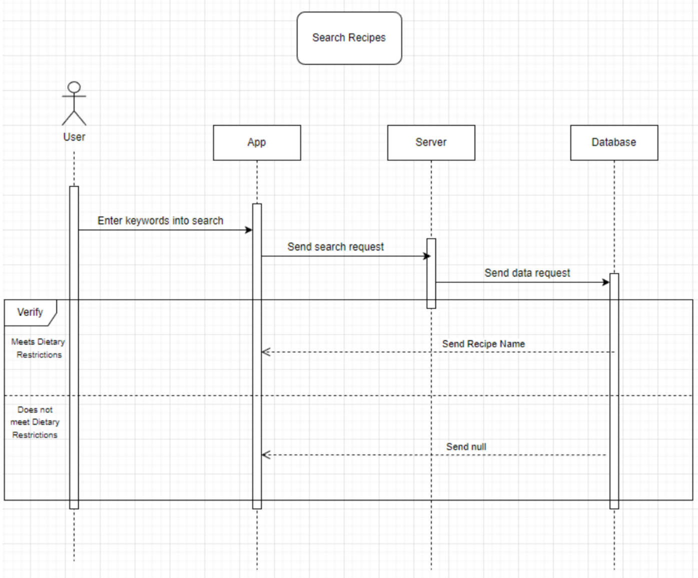
# Use Case Diagram

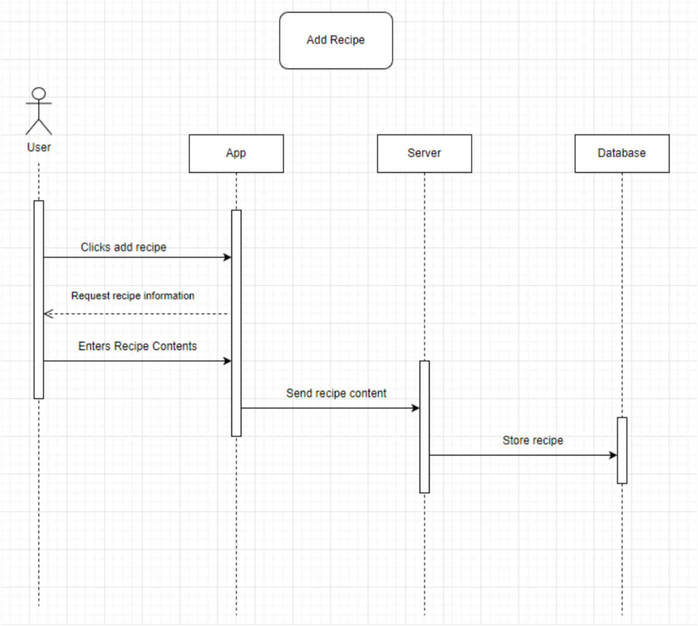


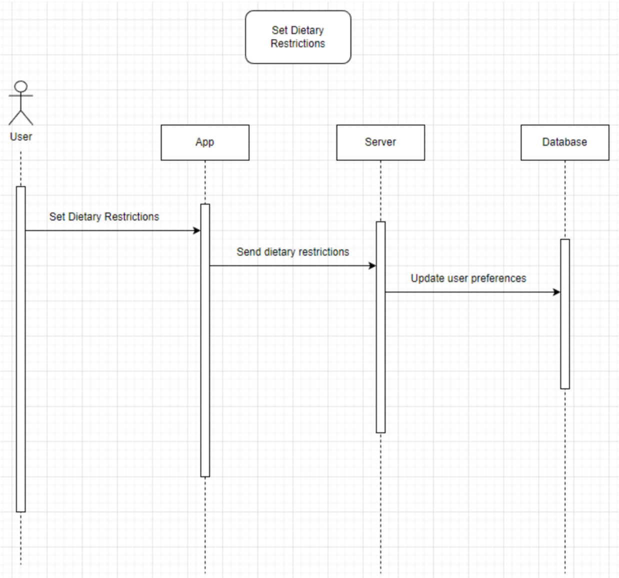
# Sequence Diagrams



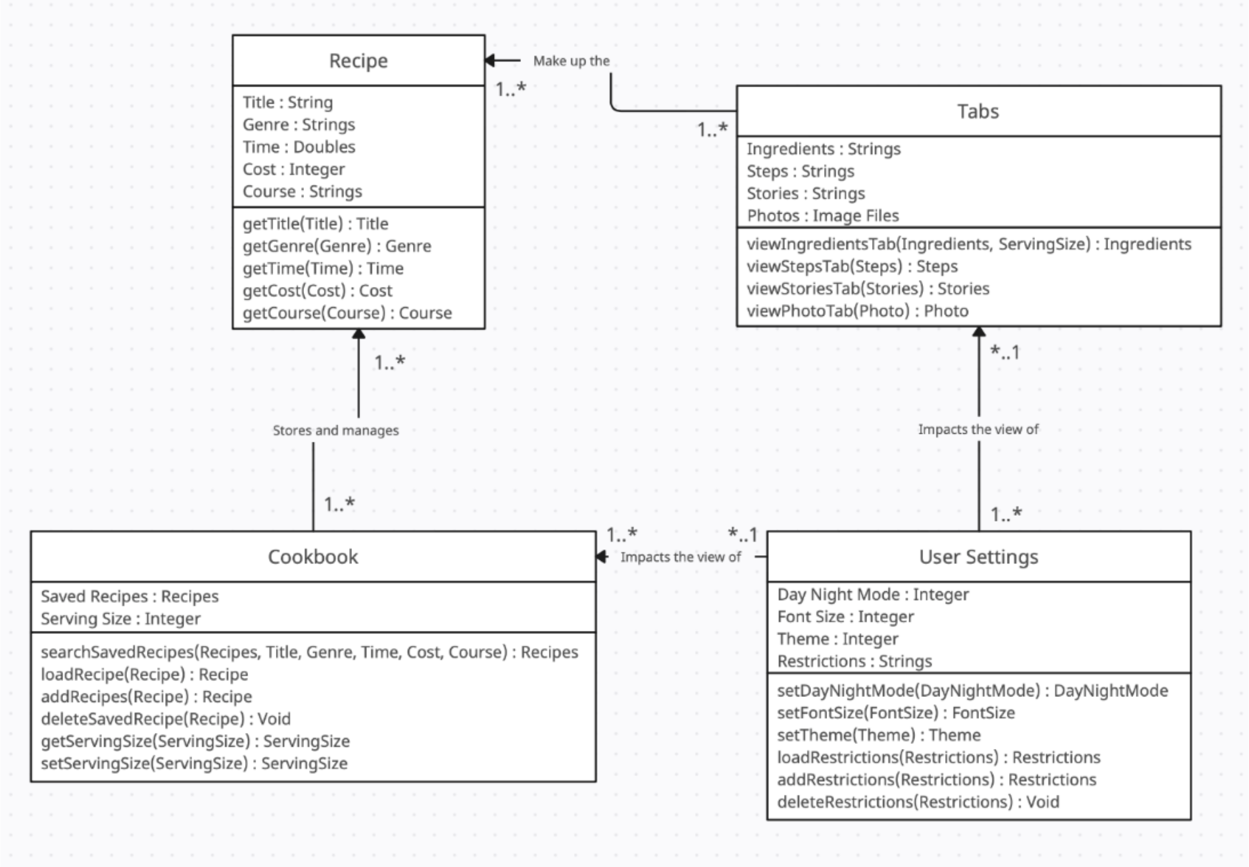



Search Recipes
User
App
Server
Database
Enter keywords into search
Send search request
Send data request
Verify
Meets Dietary Restrictions
Send Recipe Name
Does not meet Dietary Restrictions
Send null

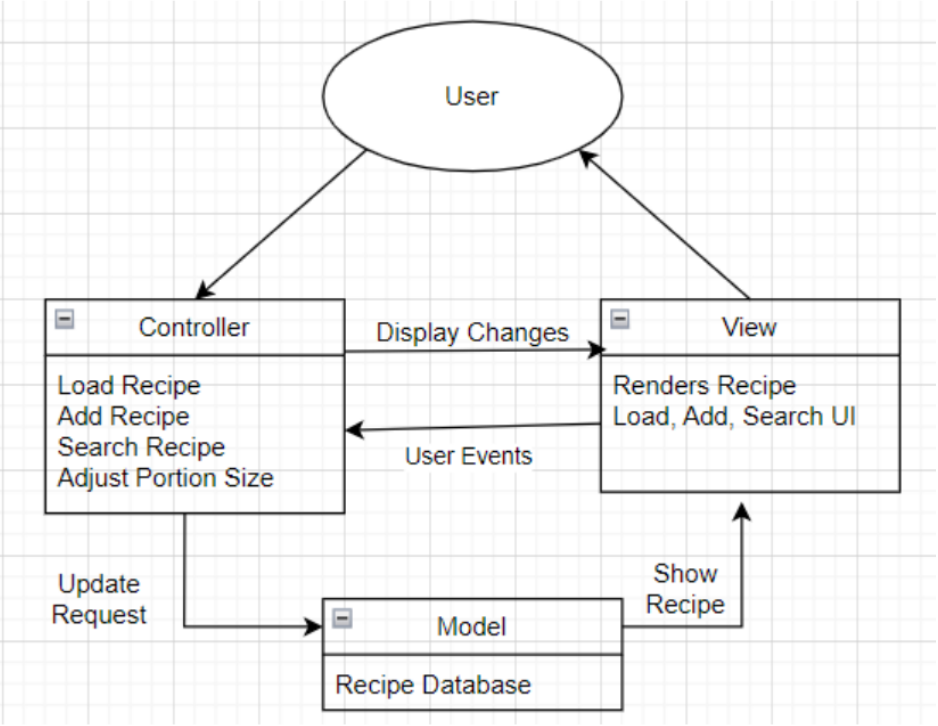

Add Recipe
User
App
Server
Database
Clicks add recipe
Request recipe information
Enters Recipe Contents
Send recipe content
Store recipe


Set Dietary Restrictions
User
App
Server
Database
Set Dietary Restrictions
Send dietary restrictions
Update user preferences

# Class Diagram


Recipe
Title : String
Genre : Strings
Time : Doubles
Cost : Integer
Course : Strings
getTitle(Title) : Title
getGenre(Genre) : Genre
getTime(Time) : Time
getCost(Cost) : Cost
getCourse(Course) : Course
Make up the
1..*
1..*
Tabs
Ingredients : Strings
Steps : Strings
Stories : Strings
Photos : Image Files
viewIngredientsTab(Ingredients, ServingSize) : Ingredients
viewStepsTab(Steps) : Steps
viewStoriesTab(Stories) : Stories
viewPhotoTab(Photo) : Photo
1..*
Stores and manages
1..*
*..1
Impacts the view of
1..*
Cookbook
Saved Recipes : Recipes
Serving Size : Integer
searchSavedRecipes(Recipes, Title, Genre, Time, Cost, Course) : Recipes
loadRecipe(Recipe) : Recipe
addRecipes(Recipe) : Recipe
deleteSavedRecipe(Recipe) : Void
getServingSize(ServingSize) : ServingSize
setServingSize(ServingSize) : ServingSize
1..*
Impacts the view of
*..1
User Settings
Day Night Mode : Integer
Font Size : Integer
Theme : Integer
Restrictions : Strings
setDayNightMode(DayNightMode) : DayNightMode
setFontSize(FontSize) : FontSize
setTheme(Theme) : Theme
loadRestrictions(Restrictions) : Restrictions
addRestrictions(Restrictions) : Restrictions
deleteRestrictions(Restrictions) : Void

## Architecture - MVC


User
Controller
Load Recipe
Add Recipe
Search Recipe
Adjust Portion Size
Display Changes
User Events
View
Renders Recipe
Load, Add, Search UI
Update Request
Show Recipe
Model
Recipe Database