# Recipe Finder Analysis

Program name: “Recipe Reservoir”

### Problem Identification

In the modern day, there are not many existing mediums of ways to find recipes apart from your typical recipe book. These are not very reliable as they tend to be a labor to get through and are limited to whatever content is within the book.

This is a very common issue that most home cooks and chefs may face with, and as a result of this, they may turn to other solutions to find a better variety of recipes that are quick and easy to access.

A computational solution may be the best approach, as it offers people with the required technology a computerized solution that has ease of access and simple functionality.

### Stakeholders

The demographic for this software would be people and workers deeply associated with the cooking industry such as chefs, bakers, line cooks, home cooks, cooking influencers and celebrity chefs who are looking for new recipes to try out and are in need of ideas.

Food establishments like restaurants, bakeries, and supermarkets may find this program suitable for their businesses, providing both their customers and employers with an impressive array of recipes they can make in the kitchen.

### Why it is suited to a computational solution

By utilizing databases which can store up to millions of bytes of data, a computerized solution will lead to a quick and efficient way of finding recipes that a user can immediately access with ease. Compared to the process of scrolling through a recipe book aimlessly, this offers a much less tedious solution.

### Background

The idea for my project is a Recipe Recommendation System built entirely in Python. It will utilize a data sorting algorithm that will recognize ingredients being input by the user and suggest possible recipes that can be made with the available ingredients.

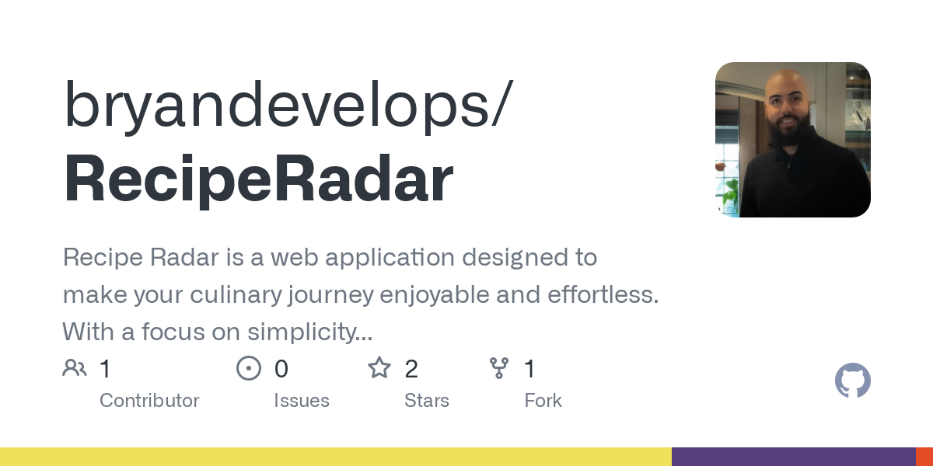
The user will be able to save recipes and keep a profile of their ingredients where they can add or remove them at any time and can be accessed by login.

### Research

### Existing Solutions

#### Recipe Radar

Website: [https://www.reciperadar.com](https://www.reciperadar.com/)

RecipeRadar helps the user to search for recipes by ingredients, it plans the user’s meals, and create food shopping lists. The service aims to be distraction-free, privacy-respecting, and is provided as free and open-source software so that users can inspect and modify the code to collaborate and make improvements.

RecipeRadar bears in mind factors such as shelf life, cost and availability, dietary requirements, flavor profiles, nutritional value, and seasonality of the produce in the user’s local area.

It uses a basic text box input that allows the user to type in a name of an ingredient and will show all the available recipes based on the ingredients entered. There are also options to filter out certain foods for dietary requirements and it is fully customizable for the user.

The program is a constraint satisfaction problem over a multi-dimensional domain of recipes, geography, time, and social networks.

RecipeRadar is strongly influenced by information retrieval concepts, and views recipe search as a relevance optimization problem. Their corpus of documents is a set of recipes, and their queries are ingredients and constraints. By carefully tokenizing, indexing and scoring the documents and queries, it can provide blazing-fast search results and improve recipe recall and result precision over time.

#### Strengths

* It has a very simple design, making it easy to use.
* It is designed to provide fast and immediate results based on a search.
* The program makes use of dietary filters, making it more accessible for people with specific dietary requirements.
* It contains a Meal Planner which allows the user to schedule their eating times.
* Shopping List feature to organize meal preparation.
* Once a recipe has been found, it provides a link to the website where the recipe was originally stored as well as the instructions and ingredients needed to make the recipe.

#### Weaknesses

* There isn’t a way to save the recipes you have picked.
* There is no user registration feature or way to save a recipe, making the program less reliable as there is no way to keep track of a recipe once it is found.
* Based on this, users will forget about a recipe later as there isn’t a quick and reliable way it can be saved.

#### Parts that I can apply to my solution:

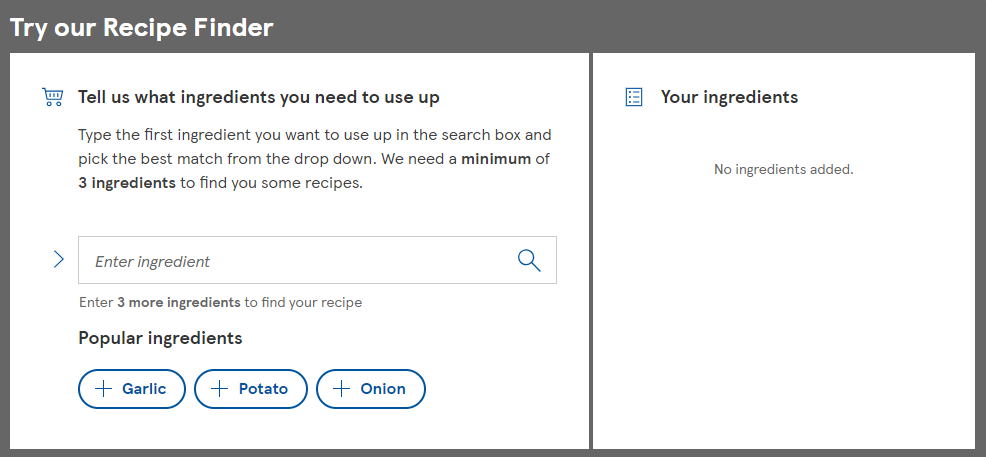
The program is still in its beta stage, and as a result, there is no built-in way to save a recipe once it has been found. This is what I plan to do in my rendition of this type of program which will differentiate it from RecipeRadar – it will keep a record of recipes a user can save and look back to whenever needed. In addition, I plan to add the Meal Planner, dietary filter, and a cooking calculator to provide the user with more interactivity.

#### Tesco Recipe Finder

Website: <https://realfood.tesco.com/what-can-i-make-with.html>



Tesco’s recipe finder shows you all the recipes you can make by inputting ingredients into their recipe finder tool on their website. The concept is like RecipeRadar, but is not as complex.

Tesco’s Recipe Finder is purposefully designed to be simple as it shows which recipes are available to the user based on ingredients and a quick search. When you click on a recipe, it redirects the user to a page of the chosen recipe right on the Tesco Real Food’s Website. In addition to a Recipe Finder, they also have programs such as: recipe binder, meal planner, and cooking calculator.

#### Strengths

* It is effective at showing quick results
* It redirects the user to a page of a recipe on the company’s website, which helps them gain traffic
* Tesco’s Website offers the user additional programs such as a binder, meal planner and cooking calculator. This provides more diversity for what the user can do with their recipes and goes into further detail on how you can prepare and schedule meals.
* There is a user registration feature which allows users to keep track of their saved recipes and create a profile for themselves.
* There is a feature that allows the user to shop for the ingredients when they are redirected to the page on the company’s website, which they will then profit from directly.

#### Weaknesses

* It uses an overly simplistic design and doesn’t offer the user many options or suggestions on what they should search.
* Due to its simplicity, users may not want to interact with it as it doesn’t look like it can provide much.
* There is a minimum input requirement of 3 ingredients which prevents the user from looking up recipes that only require 2 ingredients.

Overall, the Tesco Recipe Finder Tool is an optional program on the company’s even bigger site who the users may not even have to use, just as it suggests in the title, it says to: “Try” the Recipe Finder. While it is good at providing quick results – helping Tesco gain traffic as users are redirected to the company’s recipe pages, its seemingly bland design will draw users away from it as it doesn’t look intriguing enough.

The 3-ingredient minimum requirement also prevents people who are short of ingredients from using the program effectively, instead of providing them with an easy and effective 2-ingredient recipe.

#### Parts that I can apply to my solution:

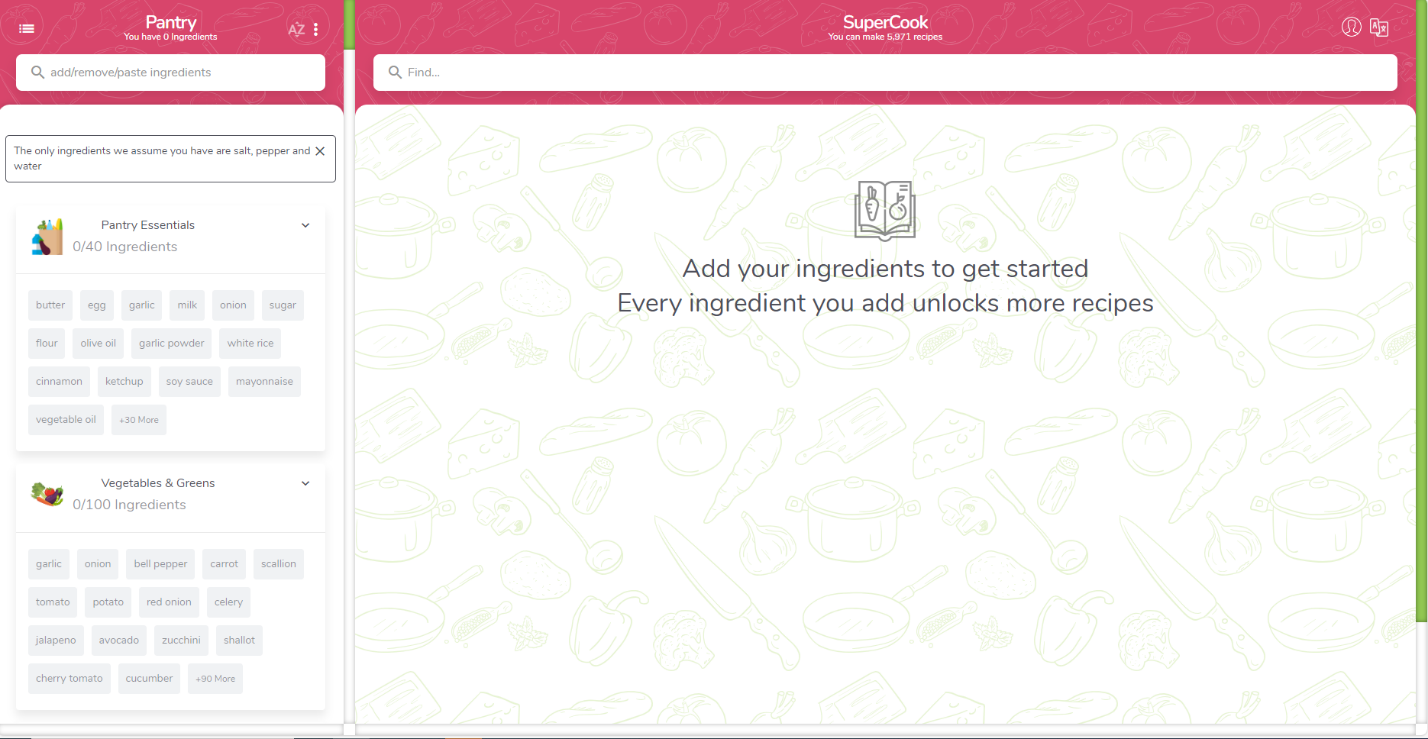
A clean, simplistic design will be straightforward to the user and will make it easy for them to use.

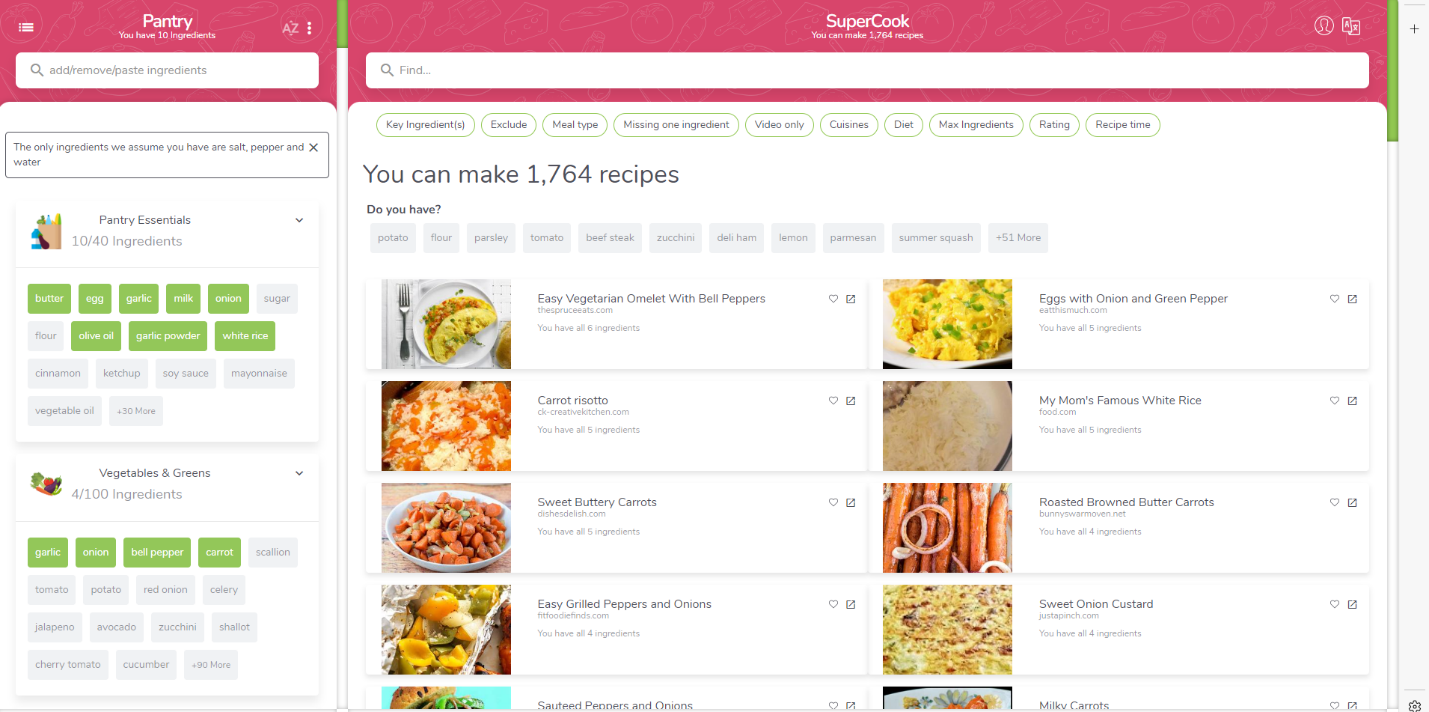
My program also aims to have blazing fast search results which will provide a link to the website where the recipe has originally been found. It will have a minimum input of two ingredients.

#### SuperCook

Website: <https://www.supercook.com>

SuperCook is an online recipe search engine that allows the user to select from a table of ingredients and update what recipes are available to them in real time.

The UI is very pleasant to the eyes as seen in the attached image above, and the process of selecting ingredients is made easy for the user, as ingredients are grouped into different categories in the left-hand side window, allowing the user to simply click on the ingredients that are readily available to them. The program also assumes that you already have access to water, salt, and pepper.

Once a user selects enough ingredients, the main window will be updated in real time with the recipes that are available to them based on the input, showing the number of available recipes as well as the link to each recipe’s page.

#### Strengths

* The UI is very pleasant and appealing, which draws in more users.
* It allows you to register an account and save recipes.
* The process of selecting recipes is made simple, easy and fun, as all the user needs to do is click on the already categorized ingredients instead of having to input them manually.
* It updates in real time based on the selected ingredients, which shows its efficiency and reliability.
* There is an ingredient suggestion tool in case an ingredient is not recognized by the program.

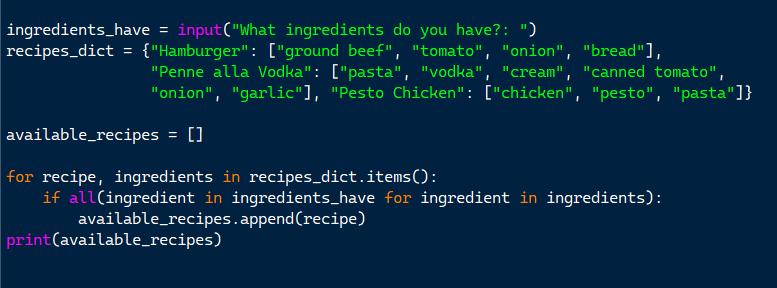
#### Weaknesses

* As the search tool relies on the selected keywords which doesn’t take literal input, the user is unable to input an ingredient that is not among the keywords of ingredients that are shown to the user, as it won’t be recognized.
* URLs are provided for the recipes that are queried by the search engine, meaning the user will have to leave the page to view the recipe, instead of viewing the instructions directly on the website.

#### Parts that I can apply to my solution:

I would like to make a similar looking UI to the one shown in SuperCook, where there is a separate window for the input or keywords and a main window for the queried results. Instead of a website, it would be its own application and the instructions for the recipes will be shown directly on the app instead of redirecting the user to the page it was originally found in. I would also like the main window to update in real time and be able to give the user the functionality of selecting ingredient keywords from a drop-down box or inputting it themselves manually.

### My Solution

Initial concept of the solution:

Attached above is a screenshot of the simplified code of this algorithm which was built in Python Idle.

The idea is to take the user input and based on a dictionary of recipes, (which contains a list of ingredients) display the available recipes that correspond to the user’s input.

The simplified algorithm makes use of a single input, a dictionary of recipes whose key is the string name of the recipe, and whose value is a list of ingredients.

It iterates through each key and value in the dictionary using a for loop and checks whether the conditional statement is true: If all ingredients exist within one of the recipe’s set of values, it will print it out to the user.

### Features of the proposed solution:

The complex solution to this problem will be making use of a database consisting of recipe names and their corresponding set of ingredients, storing hundreds upon thousands of different recipes. The program itself will have a GUI that allows the user to input, save, and even create personal recipes.

The database will be acquired via online sources. I will potentially use a Recipe API that contains existing databases filled with recipes, their ingredients, and the keys or URLs where they can be found (websites, etc.) This saves the time of me having to manually fill out a database filled with recipes from a website. While the second option is possible, this will lessen the time I spend on it as I can focus more on the front-end design.

### Finding a suitable search algorithm:

A hash search would be the most suitable search algorithm to use as these are known to provide fast, immediate results based on a query, which is what the whole program aims to do. The database could then be converted, or copy pasted into a hash table, and each recipe will then be provided with an individual unique identifier (hash key) and would then be capable of being traversed by a hash search.

A binary search could possibly be put into use to assort through the large database of recipes stored in memory; the search would divide the list into halves and compare each recipe’s ingredients with the ones inputted in by the user and return a full list of appropriate recipes that correspond with the user’s input.

### Programming language and libraries used:

My chosen programming language for this project will be Python, as I am very familiar with its syntax, and it also excels at the type of job this program is required to do. Python is a great language to use for searching and sorting algorithms, as well as storing advanced datasets such as the ones mentioned above.

Not only that, but Python is equipped with various libraries such as SQL, PySearch, Tkinter and more that will contribute to the completion of this project and simplify programming. External libraries like APIs are also able to be imported into Python.

### Extra features:

A user registration system may also be added to further this program's experience while providing security and ensuring that all data remains secure and kept safe for the user.

It will also allow the user to browse through the various profiles of people who use the software, seeing which recipes they have saved and have created for other people to use.

The GUI responsible for the program will be handled by the Tkinter module to create the windows necessary for the program to run. (i.e., text boxes, user profile screen, user interface etc.)

### Limitations of my solution

The main limitation I may encounter while programming the project is the lack of resources for databases, or Recipe APIs. I will need to find an open-source API or database, as the process of making a database myself would be wearisome, but it is still possible.

Finding a suitable sorting algorithm will also require me to go through the process of trial and error, until I decide on which algorithm is the most efficient at returning the recipes.

### Software and Hardware Requirements

#### Hardware:

* Modern Operating System (Windows 10 or later)
* x86 64-bit CPU (Intel / AMD architecture). ARM CPUs are not supported.
* 4 GB RAM.
* 5 GB free disk space.

#### Software:

* Windows, Linux or Mac operating system - You will need an operating system that supports Python.
* Python Interpreter – needed to translate the Python code.
* Python Modules – you will need to download the dedicated modules the program uses.

### Design

### Front-end Solutions

### Back-end Solutions

### Problem Research

### Objectives