A responsive web application to encourage home-cooking and reduce food waste

Contents

[Problem Description 2](#_Toc19295475)

[Account of related literature 2](#_Toc19295476)

[Account of project work and its outcome 4](#_Toc19295477)

[Legal, Social, Ethical and Professional Issues 9](#_Toc19295478)

[Review of current stage of project work 11](#_Toc19295479)

[Review of project management 12](#_Toc19295480)

[Review of personal development 17](#_Toc19295481)

[References 18](#_Toc19295482)

[Appendices 20](#_Toc19295483)

[Appendix A 20](#_Toc19295484)

[Appendix B 22](#_Toc19295485)

[Appendix C 22](#_Toc19295486)

[Appendix D 26](#_Toc19295487)

[Appendix E 28](#_Toc19295488)

# Problem Description

The problem that I am aiming to solve is that of food wastage in the average household. It is estimated that each year 1.3 billion tons of food produced worldwide is wasted, wasting money for the consumer, and increasing the worldwide levels of greenhouse gases (Kirk & Scott, 2018). My solution aims to reduce waste at the consumer level, by encouraging the cooking of ingredients nearing or in some cases just passed their expiration date. It will do this by providing facilities that will enable the user to search for recipes with the ingredients that they have in their fridge that were potentially going to be discarded. The user will also be able to manage an online store cupboard where they can enter items that they would like to be included in these recipes that they may already have stored at home. There is of course a potential risk with any food that is beyond its use-by date, and so the application will also provide some information on food safety to enable the user to make safe decisions about what to eat.

The goal of the project is to deliver a responsive web application that:

* Allows searching of online recipes by ingredients which are specified by the user
* Allows viewing of method, nutritional information and ingredients for selected recipe
* Provides information on food safety
* Allows maintenance and viewing of an online store cupboard

The solution that I am proposing is that of a web-based application, that can be used on desktop or mobile devices, due to its responsive nature. I aim to build the application using a JavaScript framework, that I will research and utilise during the project. The possible frameworks that I have explored are React and Vue. I have some previous experience using Angular, and so for this project I would like to gain some new skills and avoid the use of Angular if possible. I also aim to integrate with an 3rd party recipe API to provide the search feature for my application, as well as creating my own APIs to allow the maintenance of the store cupboard feature.

The scope of the project will also include the implementation of a database to record the store cupboard entries. This means that the solution will include sign up and login features, so entries can be stored and retrieved for a user, as well as user authentication and authorisation elements. Data access code will also need to be written to communicate with this database, as well as consideration made for the hosting of the application.

The project scope also involves the evaluation and selection of several tools that I will use throughout the project, including that of an integrated development environment (IDE) and database tools.

Should this solution be delivered, the created product should help reduce food wastage, therefore saving consumers money and cutting emissions of greenhouse gases.

# Account of related literature

When considering the validity of the problem I proposed and impact that my solution could have on solving it, I gained some insight from a study titled ‘Food Waste and Safety: Negotiating Conflicting Social Anxieties into the Practices of Domestic Provisioning’ (Watson and Meah, 2012). The study aims to provide conclusions about when and why food becomes waste within a household. The insights I gained from this study showed that there may be some people who could not be swayed to ignore use-by-dates on their food items, but for others they could be encouraged to do so with the use of my solution.

I have treated the findings of this study as somewhat reliable, because the content of The Sociological Review is peer-reviewed, meaning that it is regulated before it’s published. Saying this, the study does only consider 23 subjects, so I am not convinced of how safely the findings could be generalised to the population at large.

One of the main parts of the solution is the ‘Food Safety’ page, which includes information for the user on foods it is safe to consume when they are past their best, and foods it is best to discard. I made use of an online resource to gather the information that I have presented to the user on this page. The article titled ‘Is it safe to eat food past its sell-by date’ (Mosley, n.d) contained guidelines for eating certain foods based on advice from Dr. Patrick Hickey, who completed a PhD in cell biology, and appears to be a source of trustworthy information on the subject (Reed, 2011). I did read many other similar articles during my search for information, but most could not provide evidence that the guidelines they suggested were based on any scientific evidence, so I discounted them from use.

I have made use of ‘Jump Start Responsive Web Design’ (Ward, 2017) to research responsive design principles and techniques while carrying out research for my project. This book provides an in-depth view of many topics concerning responsive design, including what it is, ways in which it can be achieved and code examples of how techniques can be implemented. The book is very rich in information, and only some proved useful in the project work, but I was able to use this resource to determine how I would implement the responsive styles for the product. The book provided good information about alternative approaches with no bias towards any one, and this enabled me to settle on what I felt was most suitable. I can find little information about the author of the book online, but other readers have reviewed the material highly, so I deem it to be a very useful resource. The only limitation of the book is that it does not go in to detail about the use of responsive CSS frameworks, which is an area I wanted to explore before choosing how best to implement the styling of the application.

Following on from researching responsive design, I settled on using Flexbox to implement the layout for the solution and needed to use further resources to improve my knowledge. I utilised ‘A Complete Guide to Flexbox’ (CSS-Tricks, 2019) to improve my skills. I deemed this resource suitable for the task due to the recentness with which the information within the article was updated, meaning it’s based on the latest Flexbox specification and I am not writing outdated code. The website also provides information about the authors of the content, so I was able to determine their proficiency in the area before using the source (CSS-Tricks, n.d).

A main source of information that I have utilised extensively during the project is the documentation available from the developers of the Vue.js source code (Vuejs.org, 2019). I can count this as a reliable source of information as it is produced by the developers who create the code for the framework, and it is in their best interest to keep it relevant and up to date so that developers such as me adopt the use of the framework. In conjunction with the Vue.js documentation I have used the Vue router guide (router.vuejs.org) to implement routing within my application. These documents provide further information than that supplied in the Vue.js reference documents but are linked to from within the Vue.js documentation and provided by the same community of developers, and so I have treated these resources as highly reliable.

While experiencing issues during my project set-up and various coding activities, I used online information source Stack Overflow (Stack Overflow), which is an online information sharing website, where the developer community can ask and answer questions on various development issues. I found this an extremely valuable source of information which I used throughout the project when experiencing issues in many areas including writing code, database implementation and general troubleshooting. I consider Stack Overflow a reliable source of information due to the point system that is used by the website to indicate the best quality answers, and the reputation points that are awarded to the contributors on the site. Using these indicators, I can ascertain which are the higher quality responses, and utilise the advice given within them, ensuring that I am using the highest quality answers to solve my issues.

Video content site Pluralsight has proven invaluable in my decision to use Vue.js in my project. By watching videos on both React and Vue.js I was able to determine which framework was better suited for my project. The authors of content on Pluralsight are carefully auditioned before they can post content on the website, and as such I consider this to mean content is reliable (Pluralsight.com, n.d)

During coding activities, I have used the W3C website (w3cschools, n.d) for reference to find the appropriate JavaScript methods to use for my needs, specifically when trying to encode a URI to send as an API parameter. Due to the naming of the website I assumed it had an affiliation with the W3C Consortium, in which case I could class this resource as highly reliable due to their reputation, but on further investigation it seems that this website is a separate entity, and perhaps not a totally reliable source of information at times (Post, 2017). Following this discovery, I made sure to check the official JavaScript specification document to ensure that the method I have used has not been deprecated, which I was glad to find that it hadn’t (Mozilla, n.d).

I have also been utilising the documentation provided by the developers of the Edamam API during my integration of this with my project (Edamam, 2019). I have treated this as a reliable source of information since it is provided by the creators of the API and is likely to provide accurate instructions and information.

I have made use of ‘Getting MEAN with Mongo, Express, Angular and Node’ (Holmes and Harber, 2019) to aid me in the creation of the database and APIs for the project. It has offered valuable information along with code examples which I have been able to adapt to create the Mongo database and RESTful APIs for my solution. I have regarded this book as a valuable and reliable source of information due to five-star reviews that have been left for the book online, and the development experience that the authors have. Simon Holmes has decades of experience within web development and runs an online training company for developers (Holmes, 2019), while Clive Harber has similar experience and runs an agency providing web solutions (Harber, C).

# Account of project work and its outcome

**How does everything help me solve the problem?**

To begin to solve the problem firstly it was essential to finalise the requirements for the product that would provide the solution. I did so by interviewing a potential user of the application to gain insight into a 3rd party’s opinions on the potential solution for the problem. The user that I chose to interview was a colleague, who I had had no prior discussions with regarding the module or project that I am completing. This is to ensure that their responses to the questions I pose could not be biased in any way towards project ideas that I have expressed to friends or colleagues that I speak to more regularly, which would hopefully provide an accurate list of genuine requirements. Without an outside perspective it is very probable that my final solution would not solve the problem effectively, as it would have been based purely on my opinions and ideas. A transcript of the interview can be seen in Appendix A. Following the interview, I compiled a list or requirements that I deemed essential for the solution to fulfil in order to solve the specified problem, which are shown in Appendix B.

The next task was to create wireframes for the application based on the gathered requirements. These wireframes are low-fidelity, as I feel the requirements are quite simple and don’t require any complex prototyping to assess their validity in solving the problem. I produced wireframes for both a standard desktop view and mobile device, which can be seen in Appendix C. I considered tools called Balsamiq and Pencil for the creation of the wireframes as I have some experience of using both from previous modules, and they seemed like a good fit for this project due to the simple and easy to use drag and drop GUI. Ultimately, I chose Pencil as it is free to use unlike Balsamiq, and I felt the size of the project did not justify the need for financial outlay. I feel that the level of detail that the program offers in respect of wireframe fidelity is adequate for a project of this size.

Using these requirements and wireframes, I then wrote an acceptance test script, using the approach described by (Scott Ambler (n.d). I have chosen this approach as I have experience creating tests in this format from my previous OU study during TT284 and feel that the tests created following this template provide good detail about how to carry out and what to expect from the tests. This will be important if I utilise the help of an external party to carry out the tests to provide extra assurance that the requirements have been met. An example of the test cases that I have written is shown in Appendix D.

Following the analysis and design stages, I then began multiple research tasks. The first was to choose the JavaScript framework that I would use to create the components for the application. The frameworks that I had in mind to use were Vue.js and React, and after consideration I chose to use Vue.js. At this point in the project I wasted valuable time as I spent many hours deciding which framework to use and began programming along with long winded tutorials in both frameworks to help me decide. Ultimately, I abandoned this approach and used content on Pluralsight to help me settle on the use of Vue.js. I chose Vue.js due to its use of TypeScript, which felt familiar to me due to previous experience using object orientated languages during my studies, such as Java. It is also a more lightweight framework than React, which felt appropriate for the simple project that I was creating.

The second task was to research principles and solutions for responsive design, which I began to do by reading the beginning three chapters of ‘Jump Start Responsive Web Design’ (Ward, 2017). The book offered many alternative approaches to implementing responsive styles including creating my own grid layout using CSS to produce rows and columns in which I could place content, using the CSS Flexible Box (flexbox) Layout Module or using the CSS Grid Layout module. I decided to choose Flexbox for this project, as implementing my own grid system felt excessive give the simplicity of my user interface, and Flexbox has marginally better support among web browsers than the Grid specification (Caniuse.com, 2019). In retrospect this task also took a disproportionately long time to complete compared to the rewards it provided as the layout of the application proved to be very simple and did not need a great deal of responsive styles.

The third research task was to find an appropriate 3rd party API to integrate into my recipe search feature. To begin with I first needed to find a tool that I could use to easily test the API candidates without the need to write any code to consume them. The two options that I considered were Postman and Insomnia. While both products offer similar functionality, at least in terms of the functions that I require for this project, I found the Insomnia user interface easier to navigate and use to perform the tasks that I needed.  This resource is also freely available to download, and so I will be able to access whenever needed.

Once I had a means of testing the API’s I began a search engine investigation, testing the most viable options using Insomnia, and finally settling on the use of the API provided by Edamam (Edamam, 2019).  They provide a recipe search API, where I can search by ingredient, and am provided with a list of recipes and their associated data such as ingredients.  This resource is integral to the user being able to carry out the main tasks provided by the application I have built.  One requirement that I had wanted to include in the project but am unable to due to restrictions in the available API’s, is to filter by meal type, and so the scope of the project had to be slightly adjusted at the beginning to exclude that feature.  If this were an enterprise application, I would be able to use a paid-for service and access an API with this function, but for the sake of this project, I have selected the best freely available recipe API I could find.  Alternative API’s that I considered were Food2Fork (Food2Fork, 2019)and BigOven (Bigoven, 2019)but discounted these because Food2Fork only provided a link to recipes with no details of them available in the response, and BigOven needed a subscription, which was not obvious before sign-up.  All other options explored were paid-for, and so not suitable for use in the application. Using a 3rd party API does mean that the resource could potentially be unavailable when needed, and I must handle such situations gracefully in the application.  Appendix E shows the Insomnia workspace I set up for the Edamam API, which I used throughout development to test the results of the endpoint. The Edamam API returns the response in JavaScript Object Notation (JSON) format, which is a lightweight format for transmitting data objects, and is easily read using JavaScript.

Following these research tasks, and before I began code implementation, I selected some tools and resources that I would need to use to do so. Firstly, I needed to select an integrated development environment (IDE) in which I would write the code. I considered NetBeans for this task as I have used it in previous OU modules, but the support for Typescript in NetBeans must be added via a plugin, and as I will be using Typescript to write the Vue.js components, I felt that there would be a better alternative in this instance. The next option I considered was VSCode, which is the tool I decided on using for the project. I made this decision as it is lightweight, highly-regarded in the developer community and provides out-of-the box support and tools for writing TypeScript (Holland, 2018).  It is also the IDE suggested by the Vue.js development support team, and for this reason I felt that it was the best choice. It is freely available for download, and therefore should be available to me whenever required, even if the hardware I am using for development fails.

The Vue.js documentation also suggested the use of the Vetur VSCode extension (Marketplace.visualstudio.com, 2019) to aid my development, which is advice I followed as I am a novice and felt any tools that could make my learning and coding experience easier would be of great use. The extension supports error checking in the project code, as well as formatting and auto completion. As this is was suggested by the documentation, I took it as the best choice tool to use and did not explore alternatives in this instance.

At this point, I created a repository in GitHub to implement version control for the project. I also considered the use of SVN for version control, as I have previous experience using this in my OU studies, but I prefer the use of the local repositories and decentralised nature of Git over the use of a centralised server to store the source code that SVN utilises. After installing Git onto my computer, I was able to create a new repository in my GitHub account, and then use the Git Bash command line interface (CLI) tool to initialise a folder on my computer file system to be connected to the new repository (Appendix F). I deemed the implementation of version control an extremely important step in the project as it protects against code loss, as well as allowing me to clone the repository and work on the project from any computer. The git repository can be found at <https://github.com/sarahw607/tm470-app>.

I was then at the stage in my project where I could begin writing the client-side code that would create the UI for the users to interact with. After reading the Vue documentation (Vue.js, 2019), I decided to create my new Vue.js project using the Vue CLI.  An alternative approach was to reference the Vue.js code library from a content delivery network (CDN), but this would have introduced a potential point of failure as I would always be relying on the CDN being available. By using the CLI I have a base project created for me that I can adapt and build upon to create my application. Creating the project in this way also provides tooling and configuration files that will be needed for packaging and deploying the project, which will be useful when it comes to considering hosting.

Following this setup stage, I began to create my UI components, starting with the structural components that would form my basic layout, which included the navigation bar and footer. The navigation bar was the main area of the site that I knew would need different CSS styles for desktop and mobile, and so this proved to be a slightly longer task than anticipated, as the resources I had read regarding responsive design did not offer much insight into how I could achieve this specific task. With further reading on the capabilities of Flexbox (CSS-Tricks, 2019), and implementing media queries that I had learnt from ‘Jump Start Responsive Web Design’ (Ward, 2017), I was able to create a navigation bar that adapts depending on the device used. The two versions are shown in Appendix G and Appendix H**,** as well as the CSS code that I implemented to achieve this Appendix I**.** There are some improvements that could be made to this navigation, such as minimising it after navigation in the mobile view, but it is fit for purpose as I have implemented it, and I am proud that I managed to implement this component with the knowledge I gathered during my research. During the development of the navigation bar I also created placeholder pages for each one in the application, and connected the router links, with the aid of the Vue Router documentation (router.vuejs.org).

The next section that I began implementing was the Recipe Search page. I added HTML code for a simple form to accept ingredients that are added to a list of ingredients to use as parameters for the Edamam API. I created an IngredientList component which accepts this list of ingredients as a prop, and displays them**.** Following this I implemented the recipe search using the Edamam API. After consulting the Vue.js documentation I decided to adhere the method outlined there, which included using the axios (npm, 2019) library to make the HTTP requests to the endpoint. I included axios in my project using the Node Package Manager (NPM). NPM is an online software registry where open source developers share and can retrieve packages for use in development projects (Docs.npmjs.com., 2019). Appendix J shows the full Recipe Search component code, the Network tab of my browser showing successful communication with the API and the resulting user interface showing a list of returned recipes. At this point in development I realised that the response provided from the API does not include a method for the recipe, which means that part of the requirements of my solution were not met. At this stage in the project I would have put myself behind schedule had I spent any more time researching an alternative, so I made the decision to carry on using Edamam.

The next component I created was the RecipeDetails, which uses a recipe ID which is passed to the page as a URL parameter to look up the full recipe using the Edamam API. It then displays information from the returned JSON, including looping through the ingredients to show a list to the user. **Appendix x**

Following the creation of these components, I moved on to considering how I would implement the code for the API’s for saving store cupboard ingredients. From the outset I knew that I wanted to write these endpoints in a RESTful format, as opposed to using the Simple Object Access Protocol (SOAP), thanks to the experience I had gained in previous modules. Using SOAP would have added another level of complexity in that I would have had to parse the returned XML to be able to use it in my JavaScript code, whereas I can use the JSON returned from a REST API as is without any parsing to an object. Also, the Edamam API is a RESTful service, and I so I thought it best to be consistent with this in the code that I created. Security was also a factor in this decision asREST services are more secure as they can be monitored by firewalls, since they operate on HTTP, which is not the case for a SOAP service.

I had also made the decision early on that I wanted to use a document-based database approach in this project. In previous studies I have mainly focused on relational databases and wanted to use the project as an opportunity to expand my knowledge in other areas, so wanted to explore document-based approaches further. Also, the schema for the data I am saving is extremely simple, and in my opinion does not warrant the overhead of setting up a relational database. MongoDB is database I had been introduced to in previous study and I didn’t want to increase the scope of the project too much by spending more time researching alternatives, so that is what I decided to use for the project.

The next decision to make was which language I would use to write the API’s, and I took advice from the Vue developer community by choosing to use Node.js with Express to write the backend code in JavaScript (Gore, 2018). Due to time constraints I did not have time to investigate alternatives in much depth, but I feel confident that this was the correct decision due to the framework’s popularity among other developers, and the fact that I can use JavaScript which is the language I feel most confident using. Express (Expressjs.com, 2019) is a framework that is used within a Node.js environment and provides features to create web applications. It has more than 100 contributors from the open source community, and is actively developed and supported (Holmes, S. and Harber, C, 2018) Node.js is

Subsequent to making of these decisions it took me some time to find a suitable resource from which I could learn the required skills to complete this code, with many simply going through steps to code creation without any explanation as to why things were being done. I finally found ‘Getting MEAN with Mongo, Express, Angular and Node’ during a library search. A brief skim through the content showed that I would be able to exchange the Angular part for my own Vue.js code, and so I began to follow the tutorial set out in the book.

This involved firstly installing the express-generator (npm, 2019) package globally on my computer, using NPM. This tool is used to create Express applications via a CLI and is what I used to create my skeleton Express application. I also installed and used nodemon (npm, 2019), which is a tool that watches for changes in the node.js application and restarts the server automatically to save me having to do it manually after every change I make. Following this I had a working Express application, which I then hosted on Heroku (Heroku.com, 2019), as per the tutorial. This was very straight forward as it links with the Github repository, and so after a small amount of set-up my API was hosted and available at <https://fierce-caverns-66474.herokuapp.com/>

I had not considered Heroku as an option for hosting, supposing that I would use AWS once I got to this part of the project, as this is a provider that I have had experience with in my previous studies. Now that I have worked with Heroku I feel that it was a great choice for the project as it offers a free tier for small projects, and setup does not involve configuration that I would have needed to do with AWS. As I was getting behind in my schedule using this option seemed the most time-effective. Heroku deploys changes with each push to the Git repo, so I did not need to take any action to deploy separately, which was a great time-saver **Appendix x Heroku shot**

The next stage was to build the data model using MongoDB. I first had to install MongoDB on my computer, using an installer found on the MongoDB website (Docs.mongodb.com, n.d.). This installation also included the MongoDB compass tool that I used to explore the databases and their collections on my local machine. **Appendix.** As per the tutorial I also installed in my project the JavaScript library Mongoose using NPM. Mongoose is a library that allows interaction with a MongoDB database without the need to write lots of boilerplate code. I used it to maintain the connections to the database **(Appendix)** as well as to create the schema for the User model I have implemented for the application. **Appendix**. By implementing the model using Mongoose I can enforce a structure in the document as well as validating data values and defining the data types that are saved to the database. These were the only steps I needed to carry out to implement the database, as MongoDB creates the database itself when I first try to interact with it.

At this point in the tutorial I created a production database, using the suggested cloud provider MongoDB Atlas. As this service is provided by MongoDB themselves, I saw no need to search for an alternative, and followed the steps given in the tutorial to create a database to be used by my hosted API application. I used the Mongo Shell CLI tool to interact with the database to test everything was set up properly, and to push some test data **(Appendix X)**

The next step in forming the solution was to create the RESTful endpoints that would provide the UI with data, along with facilitating user sign up and login. I firstly had to define a number of routes within the Express application that would map an incoming URL request to a controller, which would then perform the necessary actions and return the requested data in its response (Appendix X). Once this code had been written I created an Insomnia workspace that I used to

Following the tutorial involved using a lot of tools I used without being able to consider alternatives, but given that this would have greatly increased the scope of the project work and taken valuable time, I could not spend the time doing this exploration. The writer of the book is very experienced in development, and therefore I trust that his recommendations are sound.

The final task in developing my solution was to test that it met the requirements that I gathered at the beginning of the project. I did so by asking a potential user to execute each of the tests from the acceptance test script that I produced in the planning stage, and to record the results as they did. Unfortunately, due to my failure to get the application hosted they had to do so running the code locally in my development environment, but the results were still indicative of how appropriate the solution was to solve my problem. A sample of the test results are shown in **Appendix X**

Testing – did all my tests pass? No, Edamam API needed better testing as didn’t include method. UX things like loading indicator

Food safety page smaller than I anticipated at beginning, ended up being a small information page rather than a full lookup by food

Some technologies haven’t been able to research, just used because they are used in the tutorial. Need to explore further to increase understanding and potentially use alternatives. There are just so many alternatives!!

## Legal, Social, Ethical and Professional Issues

During the completion of my project and creation of my solution, there were multiple legal, social, ethical and professional issues that I needed to take into consideration.

On such consideration is the storing of personal user data, for the purposes of allowing user login and user account retrieval. My application is storing a user’s email address and name along with a password, which falls into the category of personal data, and therefore I must adhere to regulations set out in the General Data Protection Regulation (GDPR). A main stipulation of the GDPR is that the user must consent to personal data being stored, and this consent must be explicitly given (Allen, 2018). For this reason, I will need the user to confirm their consent for me to save their data before I do so. I must also make clear to the user the purpose for which the information is being collected as well as ensuring that the data I am collecting is stored securely, and not open to data breaches (Martinez, 2018).

Related to the issue of personal data is also that of security within the application. If I am storing personal information, then I must ensure it can only be accessed in ways that I allow. One way I have protected this data is by adding authentication using JWT to ensure that my API’s will only return data to authorised clients. I have also ensured that I have not saved user’s passwords in plain text within the database but have used encryption methods to save encrypted passwords. This means that even if a 3rd party were able to access my user records they would not be able to decipher user passwords and would be unable to access user accounts.

I do not intend to track users of this product with the use of cookies or Google analytics tools, and so should not need to consider privacy issues with regards to these activities. I am however storing a JWT web token in the localStorage of the user’s browser for authentication purposes. This being the case I must consider the EU ePrivacy Directive, as this states that the user must give consent to the application to store and retrieve information on their computer (Law Stack Exchange, 2018). The application does not currently seek consent from the user or include any information on this storage within a Privacy Policy page, which would need to be done were it to be used in a real-world scenario.

A further issue I have had to consider during my project is that of the use of human subjects to gather requirements for the application. To ensure that I acted ethically with regard the subjects, I first gained informed consent from them, with the use of the form shown in Appendix D. This action meant that the user was informed of the data I would be collecting, how it would be stored and that they had a right to withdraw from the interview at any time. I also endeavoured to ensure that I did not lead the interview with my own opinions and allowed the interviewee to express their own views without direction.

During the project, I must consider whether it is ethical to encourage the consumption of out of date food products, and how this could potentially affect any users of the application. I would have a duty of care to my users to ensure that the application would not cause them harm in any way, and so the information that I provide on the consumption of foods past their sell-by date must be proven to be safe and correct, and endorsed by professionals. Aside from this issue, I can only foresee the impact of this product as positive, due to its main purpose of promoting eating of food that would potentially be wasted, with the aim of helping the environment. I cannot envisage a situation in which the product could be misused for nefarious purposes, or impact stakeholders negatively. I have included a disclaimer on the ‘Food Safety’ page so that users are aware that they are consuming any out of date foods at their own risk.

The product that I am producing could potentially be used by a variety of users, each of which could have a varying degree of ability. To this end, I need to ensure that the product I am developing is usable by as many different users as possible, and therefore that the code I am writing complies with accessibility requirements (GOV.UK) and is compliant with all the relevant coding standards.

Professional considerations that I need take in to account are that I am using only reputable 3rd party libraries and integrating with trustworthy 3rd party services. Vue.js is an open-source project that is maintained by several developers within the development community, and as such, is unlikely to be of low quality, and not follow best practises, due to the use of the pull request process when changes are made to the code (GitHub, 2019). The Vue website specifies a code of conduct that it’s contributors should adhere to, adding to the framework’s reputability (Vuejs, n.d.)

Another legal issue that I must consider is the agreement under which I am able to use the 3rd party recipe API that I will integrate into my application. To do so I must ensure that I am not using the service for any commercial or business purpose, which I am not (Developer.edamam.com., n.d.). I have also used images and icons that I sourced from the internet and have had to ensure that I am using them under the conditions that have been set out by the creator. This involved including at attribution to the creators in the footer section of my application.

By choosing cloud providers to deliver parts of my solution, I have had to make considerations about the legal implications of data being passed to a 3rd party. By deciding to host both the production database and the application code itself using service providers rather than on a local server, I have passed data to these providers. As this data includes personal data, I must make sure I am using providers with ample measures in place to protect this data.

# Review of current stage of project work

In my opinion, the project work is currently in a state that does solve the problem I have outlined but could use some additional work to make it the best possible solution. Following acceptance testing it has been demonstrated that the majority of the requirements that I set out at the beginning of the project have been met, which in my mind means that I have provided a viable solution to the problem.

Acceptance testing did uncover that two of the initial requirements have not been met by the solution, but I would deem these as non-essential to the effectiveness of the application and would be considered nice-to-haves. Fulfilling these requirements would make the application more of an efficient solution, but do not render the solution I have provided unsuccessful by their omission.

The first requirement that wasn’t met was to provide the ability to filter and sort the recipes that were returned from the recipe search API. Currently the API is only returning ten results for each search, and I was unable to find information in the documentation about how I might increase the number of results. For this reason, implementing filtering and searching feels unnecessary at this point. Going forwards the product could be improved by finding a way to return more results and allowing filtering by certain criteria such as dietary requirements, but this would potentially involve integration with a paid-for recipe API which would need to be considered for its benefits.

During development it became apparent that the Edamam API response does not return a full method for each recipe, and so this has meant that there is a shortcoming in the information displayed to the user for each recipe. There is a link to the full recipe that I have included, but the solution would be of better quality if the method could be displayed within the application. This is another area that would benefit from further investigation into a better API in the future.

The second requirement that was not met was providing search capabilities on the food safety page. At the beginning of the project I envisaged this area of the application to be an index of different foods offering advice for each on whether they were safe to consume when past their best or not. During the research for this section of the application I was unable to find sources of information with this granularity of data, and so instead had to offer more general food advice on this page. I do still think this part of the solution is sufficient as it is, as it still offers information that will encourage consumption of some foods that may otherwise have been discarded.

Beyond the requirements, there was a final project task that I was unable to complete, which would make my solution reachable to a greater audience, and that is hosting the application. As part of the coding for the API’s I managed to deploy the Node.js project to Heroku, meaning that the API endpoints are available over the Internet. However, I did not manage to get the Vue.js project hosted, meaning it can currently only be accessed by retrieving the code from source control and running it locally in a development environment. In this state the project is obviously not capable of making a difference in many households, and so would be greatly more effective by being made available over the Internet. This would also mean I would be able to carry out more extensive cross-device testing, as I would be able to access the application on multiple devices.

The testing phase of the project showed that the application was functional in terms of requirements but could use some improvements in terms of user experience. For example, it was noted that error handling from the API’s could be improved, as sometimes a message is not displayed to the user with details about why an action has failed. Also it was noted that it wasn’t clear to the user when data from the API’s was loading, and that a loading indicator would have been a good addition to provide feedback to the user and improve user experience. In addition to these issues, while reviewing my code I noticed that there are lines that I have repeated in numerous places that would benefit from being written in one place and reused where needed. Due to a lack of time I have implemented some code in the quickest means possible which perhaps has not adhered to best practises, such as code reuse. I feel that the project would benefit from some refactoring and further consideration to the user experience, although I do not believe this a necessity to improving the solution I have provided to the problem.

If the application were to be hosted and become available to the public then some additional work would need to be done to ensure that it adheres to the EU ePrivacy Directive and GDPR legislation. The application does not currently offer information about the use of the personal data that is collected or offer a way for the user to consent to this data being stored. To this end, the application should include a privacy policy providing the user with the necessary information regarding the storage of their data and provide a means by which the user can consent to this data being stored.

A plan for the full completion of the project would be as follows:

|  |  |
| --- | --- |
| **Task** | **Completion Date** |
| Vue.js API error handling review and loading indicator | 20/09/2019 |
| Refactoring to reuse common code | 24/09/2019 |
| Create Privacy page and Cookie consent popup | 27/09/2019 |
| Complete hosting of Vue.js site | 05/10/2019 |
| Testing and fixes | 13/10/2019 |

# Review of project management

My project management of this project began with defining the tasks that would need to be completed and identifying tools and resources to ensure that the solution I presented solved the given problem. I then created the following project plan, bases on the lifecycle model that I chose to adopt, which was an iterative waterfall approach.

| **Tasks** | **Sub-Tasks** | **Resources and tools required** | **Skills required** |
| --- | --- | --- | --- |
| Planning and Design | Produce and evaluate wireframes | Computer based wireframing tool. | Wireframing skills using chosen tool, which I have from previous module studies. |
| Write acceptance tests |  | Skills on how to write acceptance tests will need refreshing as have not done since level 2 modules. |
| Carry out responsive design research | Reliable learning sources in this area | Basic CSS and HTML understanding, which has been gained from previous modules. |
| Recipe Search | Write HTML, CSS and JavaScript for *Recipe Search* | IDE for development | HTML, CSS and JavaScript |
| *Recipe Search* testing | Potential users could be utilised to provide more thorough testing | Issue tracking system for bugs, need to research what would be suitable. |
| Recipe Search refinements | IDE for development | HTML, CSS and JavaScript |
| *Recipe Search* API integration | Web search engine, thorough and readable documentation provided by 3rd part API provider. API testing software. | Those gathered during TM352 module |
| Recipe Search Results | Write HTML, CSS and JavaScript for *Recipe Search Results* | As above | As above |
| *Recipe Search Results* testing |
| *Recipe Search Results* refinements |
| View Recipe | Write HTML, CSS and JavaScript for *View Recipe* |
| *View Recipe* testing |
| *View Recipe* refinements |
| Database | Research and implementation of database | Documentation for database technology chosen, potentially additional database management software | Either relational database or document-based approach will be adopted, experience of which I have from TM351, but that will need expanding upon |
| Write code for data access layer | IDE and documentation on desired implementation. | It should be possible to utilise existing coding skills to produce this code but may require some research regarding the desired architecture of software components. |
| Login | Implement user login feature | IDE, documentation on best practises, or potentially a 3rd party authentication provider | Existing coding skills can be utilised, but knowledge on the area and best practises will need improving. |
| Add Store Cupboard Item | Write HTML, CSS and JavaScript for *Add Store Cupboard Item* | As above | As above |
| *Add Store Cupboard Item* Testing |
| *Add Store Cupboard Item* Refinements |
| View Store Cupboard Items | Write HTML, CSS and JavaScript for *View Store Cupboard Items* |
| *View Store Cupboard Items* Testing |
| *View Store Cupboard Items* Refinements |
| Write HTML, CSS and JavaScript for *Food Safety Index* |
| Hosting | Website and database hosting | Hosting platform Heroku | I have gained knowledge about cloud hosting from previous OU modules, but this will need refreshing |
| Testing | Acceptance Testing | Access to potential user of the solution |  |
| Cross browser and mobile device testing | Tool that allows emulated testing across multiple browsers | Knowledge of how to operate various types of devices for testing purposes. |
| Module tasks | TMA01 |  |  |
| TMA02 |  |  |
| TMA03 |  |  |
| Final project report |  |  |

During planning the schedule for completion of these tasks was translated into a Gantt chart **(Appendix x),** which I used during the project to keep track of timescales. I also used Trello board to track issues and progress of the project **(Appendix x)**. I considered Jira as a tool I could use for project management, but Trello seemed the correct fit due to the simplicity of the project tasks, and the fact that I was working on the project in isolation. Jira offers a lot of features that I would not have needed to utilise and so I felt the use of it would be excessive in this case. Using Trello for issue tracking aided me in visualising the workload and kept me on track with approaching task deadlines. I also maintained a work diary **(Appendix x)** throughout the project which was invaluable in recording outcomes of tasks, specifically research and learning tasks, and greatly aided me in my reflection of this project.

For the duration of the project I had to constantly consider any risks to its completion and attempt to mitigate any issues that occurred that may hinder my progress. The main risks to project completion that I identified are as follows:

| **Risk No** | **Major Project Risks** | **Likelihood** | **Impact** | **Weighting** | **Mitigation** |
| --- | --- | --- | --- | --- | --- |
| R01 | Learning a new JavaScript framework will take longer than anticipated | M | H | 6 | Try to get going with coding as early in to learning as possible, and code along with the tutorial |
| R02 | Product requirements may change | L | M | 2 | Ensuring that I am regularly checking the list of defined requirements as I am carrying out development |
| R03 | ‘Feature creep’ may occur as I begin coding and the application will become more complex than I anticipated | M | H | 6 | Ensure that the wireframes and requirements are referred to regularly throughout the development process, so that I am focused on the solution that has been proposed, and nothing beyond this. |
| R04 | Hardware failure | L | L | 1 | Source code will be committed to source control, and other project files are saved to the Dropbox. Recovery should be swift if hardware fails. |
| R05 | Struggles to set up development environment | L | M | 2 | Ensuring that I have credible and relevant information sources available to help with any issues, such as the Vue.js documentation and Stack Overflow. |
| R06 | Falling behind in project schedule due to concurrent studying of another OU module | H | H | 9 | Ensure that the schedule includes time needed for commitments for my other module. |
| R07 | Estimations and scheduling activities carried out for project are not accurate or achievable | M | H | 6 | Make sure estimates include a buffer so that any extra time that is needed for certain tasks can be made up during time set aside for easier tasks. |
| R08 | Inaccessibility of chosen recipe API or breaking changes implemented by its creators | L | H | 3 | As this is a 3rd party provided service there is not much I can do to prevent any issues caused here. It might be helpful to create a file of dummy data, so that I can work ‘offline’ with this JSON if needed. |
| R09 | Project scope becomes unmanageable | H | H | 9 | Do not spend time on any tasks that have not been specified as a requirement for the solution or have not been included in the project schedule. Always keep the task list in mind to ensure only meaningful tasks are completed. |
| R10 | Illness | L | L | 1 | There should be enough slack in the schedule to allow for days missed due to illness, there is not much I can do to mitigate this risk. |
| R11 | Developing tunnel vision during the project completion, and losing sight of the desired end product | M | H | 6 | Ensure that users are involved as much as possible throughout the process. Keep referring to requirements to ensure they are being met by the tasks being carried out. |

Of the risks identified, R09 and R06 were identified as having the greatest potential to affect the completion of the project, and now that the project is completed, I would say this has been proven true. **RO1 spent lots of time reading things that weren’t relevant, good for learning, pointless for project** R07 also had quite an impact, as I did not include enough time in later tasks for researching technologies to use, which impacted on task completion dates. Even though this has been the case, I have adjusted timescales accordingly throughout the project, and have still produced a solution that I feel solves the stated problem.

The scope of the project was adjusted within its duration, but I endeavoured to only do so to include work that I felt was integral to the completion of the project. For example, following initial planning user log in functionality was not included as a requirement, but it soon became clear that the solution would not be fit for purpose without it, and so it had to be included. This change in scope did influence some of the later tasks in the plan, and I wasn’t able to complete the ‘Hosting’ task due to this. To my mind this shift in priorities was worth it for an improved end product.

I created an updated plan some months into the project, when it became clear that I had fallen behind in the tasks due to commitments to my other OU module. This new plan was mostly achieved, as I had overestimated the time needed for some tasks, although a lot of this slack was absorbed in other areas as I had greatly underestimated the time needed. For example, the latter Vue components I created did not take very long at all, as my proficiency in coding them was increasing with each one, but the time it took to implement the APIs and authentication was much greater than I anticipated, due to me poorly considering how I would implement these features and the steep learning curve it may entail. The adjusted project plan did not leave me much slack, so there was quite a lot of pressure to keep on track to ensure completion of the work. These time concerns balanced out in the end, and I was able to complete all but one of the project tasks in time.

I feel I could have improved my project planning and management with creation of more granular tasks that referenced more specific activities during my planning activities. For example, there was lots of work involved in writing the RESTful API’s for the store cupboard and log in, including researching which technology stack to use, creating the database models, writing the code including error handling and testing, but I had one task in my plan to cover all aspects. Breaking these tasks down further would have made them feel more manageable, would have enabled me to plan my time more efficiently and would have enabled me to feel like I was achieving results more quickly.

The lifecycle model that I initially decided to adopt for this project was an iterative waterfall approach. I planned to predominately undertake each stage of the project in sequence, rather than iterating multiple times, but planned to iterate within the implementation stage by completing some testing in this stage. In practise I have not completed this testing after the implementation of each feature, and instead completed all acceptance testing once the implementation had been completed. Therefore, in practise, I have used a more standard waterfall project lifecycle. This was not a conscious change in methodology, but the momentum I gathered from the enjoyment of learning Vue.js meant that I was eager to carry on coding linked application components rather than stopping to test a feature before I moved onto the next, and I didn’t want to slow down the progress I was making with learning Vue.js. I think this change of approach is quite specific to this project as I have been learning a new framework, and if I were more familiar with Vue.js I feel I would take the time to test each feature before moving on to the next.

I feel that I have not rigidly stuck to the waterfall lifecycle due to the poor project management on my part, namely because I fell behind with some tasks early in the process. Due to not adhering to the project plan completely in the early stages I had to revisit the design stage once I had already moved on to the implementation stage. This was because I did not complete my responsive framework research in time and needed to move on to coding while this was still taking place. As such I needed to revisit the wireframes once they should have already been completed. This did not cause too much of an issue in the grand scheme of the project but was disappointing to me as it went against the lifecycle model that I had chosen.

I do feel that the waterfall model has been suitable for me in this project, as I have needed to spend a lot of time researching and obtaining new skills and knowledge. The waterfall approach leaves me plenty of time to do this as I am not implementing before I have designed or testing before I have finished implementation. If I had adapted a more iterative approach, I would have needed to learn even more upfront, to allow me to complete small chunks of the application in each iteration, meaning it would have been even longer before I was able to jump in and start implementing code. The waterfall approach has enabled me to learn and code in unison. Saying this, I do feel as if adopting an iterative lifecycle would have enabled me to feel like I have achieved more once I started coding, as although I feel like I have made a good amount of progress in the project, an iterative lifecycle would have enabled me to finish complete sections of the project early on, rather than completing everything in one stage.

In future projects I envisage that I will use a more agile method to ensure that users are involved with testing parts of the project as soon as possible. Using the waterfall methodology has meant that I only met with users right at the beginning for requirements gathering, and then again right at the end for acceptance testing. In my opinion much more user input is required throughout the entire process to ensure a product is produced that meets requirements and is fit for purpose.

I did find working in the waterfall style a bit chaotic at times, it felt that there was a lot of scope in the implementation stage to flit between numerous different tasks, and I was guilty at times of picking up new tasks when I had become stuck and needed a break from other tasks. I think working in the agile method in the future would deter me from this as I would have very specific tasks in each iteration. Despite this, I do think that the waterfall approach was suitable for my project due to its size and the fact that the requirements were fixed early in the lifecycle without much need to change. I can see how this approach would not be ideal for larger projects and those with requirements that are liable to expand.

Overall, I feel that my project management definitely could have been improved, particularly in regard to keeping on schedule, but as the project progressed, I did become better at managing my time and prioritising tasks, and have learnt some valuable lessons from planning and managing this project.

Spent way too long making decisions and reading/practising, when I should have got on with practical stuff.

Number of research tasks took way too long and weren’t worth it for the payoff, barely used much of the information gathered in responsive research task, although it was good to learn for my own development.

Feel the scope may have been too large to begin with as there are a lot of new technologies, concepts to learn for the project, nowhere near enough time to implement the backend technologies, didn’t help I didn’t have a clear idea of how I would implement at the beginning. Would have benefited from a research task

# Review of personal development

Reflecting now that the project is complete, I can say I am pleased with the personal development I’ve made during the project and particularly with the additional development skills that I have attained so far. I have gained several skills that will prove extremely valuable to me in my future career in web development, including programming skills, and understanding of project management concerns and wider appreciation of legal and professional issues that go alongside these types of projects. All these skills will be extremely useful in my chosen career and will hopefully increase my employability in the developer market. Saying this, I have found the completion of this project to be rewarding and challenging in equal measure. After studying in a very structured way with previous modules it’s been a test to plan and manage the workload of this module, and I have struggled with this aspect at times throughout the project. Despite this, I am very pleased with what I have achieved, and have gained many valuable skills from the work I have undertaken.

I have realised that I learn much more effectively from practically using something, rather than purely reading about it, and will ensure in the future that I do not get lost in documentation when exploring new technologies or concepts. The best way for me to learn about these new things is to use them, and following this project I will be much more aware of the length of time I spend researching topics that lead to no tangible

I have struggled with my time management skills, but this project has aided me in giving me good practical experience of issues that occur due to poor time management and has provided good lessons for me to learn from in future endeavours. I do however believe that during the project my shortcomings in this area have been reduced, in part due to the rigorous planning that I have had to do as part of the project. These practises have given me good tools to carry forward into future projects, although there is still room for improvement going forwards in this area. With more practise I feel that I will be able to reduce my degree of procrastination and increase my focus on tasks.

The project has also improved my problem-solving skills, as I have been able to effectively solve issues I have encountered during the project. I have been able to quickly identify useful sources of information and obtain the pertinent information from these sources to enable me to solve issues quickly. I have also been able to effectively utilise the module forums and social media groups when I needed clarification on issues that I have encountered throughout the project. I have also gained an increased level of confidence in my own development skills, and feel that I would not be capable of becoming more involved in the developer community, perhaps by contributing on Stack Overflow or becoming involved in an open source project. These are definitely avenues I will explore as my development skills continue to develop.

One area that I still feel I could improve in is my communication skills. I have conducted the project mainly in isolation, except for discussions with potential users during the requirements gathering and testing stages. I also mainly used online sources and OU forums to aid me I needed clarification on issues during the project, but I have felt satisfied that I was able to find the information that I required by these means. I have conducted my study of all other modules in this manner, but perhaps could have benefited from greater discussion with my tutor, and as such this is a consideration I will take forward into future endeavours.

# References

Kirk, A & Scott, P (2018) *The Guardian* [Online]. Available at <https://www.telegraph.co.uk/news/2018/01/02/no-time-leftovers-astonishing-scale-food-waste-uk-around-world/>(Accessed 28 February 2019)

Mosley, M (n.d) *Is it safe to eat food past its sell-by date?* [Online] BBC. Available at <https://www.bbc.co.uk/programmes/articles/2V5HNS9srK9bXpLyhxLg8Lx/is-it-safe-to-eat-food-past-its-sell-by-date>/(Accessed 28 August 2019)

Reed, S. (2011). *A Mycologist Reaps the (Glowing) Fruit of His Labor*. [online] Science | AAAS. Available at: https://www.sciencemag.org/careers/2011/11/mycologist-reaps-glowing-fruit-his-labor [Accessed 28 Aug. 2019].

Ward, C. (2017) *Jump Start Responsive Web Design*, Victoria, SitePoint

Vuejs.org. (2019). *Vue.js*. [online] Available at: https://vuejs.org/ [Accessed 6 May 2019].

cli.router.org. (n.d.). *Named Routes*. [online] Available at: <https://router.vuejs.org/guide/essentials/named-routes.html> [Accessed 15 Jun. 2019].

Holmes, S. and Harber, C. (2019). *Getting MEAN with Mongo, Express, Angular, and Node, Second Edition*. [S.l.]: Manning Publications.

Holmes, S. (2019). *Simon Holmes - web developer, engineer, architect, author*. [online] Simonholmes.com. Available at: http://www.simonholmes.com/ [Accessed 9 Jul. 2019].

Ambler, S.W. (n.d.), ‘Customer Tests as Requirements Artifacts’; available at <http://www.agilemodeling.com/artifacts/acceptanceTests.htm> (accessed 06/06/2019).

Harber, C. (2019). *DistortedThinking.Agency is a web agency specialising in creating rich internet applications that are engaging and easy to use*. [online] DistortedThinking.agency. Available at: http://www.distortedthinking.agency/ [Accessed 9 Jul. 2019].

CSS-Tricks. (2019). *A Complete Guide to Flexbox | CSS-Tricks*. [online] Available at: https://css-tricks.com/snippets/css/a-guide-to-flexbox/ [Accessed 5 Jun. 2019].

CSS-Tricks. (n.d). *About | CSS-Tricks*. [online] Available at: https://css-tricks.com/about/ [Accessed 9 Sep. 2019].

Pluralsight.com. (n.d.). *Teach | Pluralsight*. [online] Available at: https://www.pluralsight.com/teach [Accessed 16 Apr. 2019].

w3cschools (n.d) *JavaScript encodeURIComponent() Function* [online] Available at <https://www.w3schools.com/jsref/jsref_encodeuricomponent.asp> [Accessed 1 Jul 2019]

Post, T (2017) *Stack Overflow* [online] Available at <https://meta.stackoverflow.com/questions/280478/why-not-w3schools-com> [Accessed 1 Jul 2019]

Mozilla (n.d) *Mozilla eb docs* [online] Available at <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/encodeURIComponent> [Accessed 1 Jul 2019]

Edamam. (2019) *Recipe Search API Documentation*. [online] Available at: https://developer.edamam.com/edamam-docs-recipe-api [Accessed 11 Apr. 2019].

Law Stack Exchange. (2018). *Do the GDPR and Cookie-Law regulations apply to localStorage?*. [online] Available at: https://law.stackexchange.com/questions/30739/do-the-gdpr-and-cookie-law-regulations-apply-to-localstorage [Accessed 9 Sep. 2019].

Watson, M. and Meah, A. (2012). Food, Waste and Safety: Negotiating Conflicting Social Anxieties into the Practices of Domestic Provisioning. *The Sociological Review*, 60, pp.102-120.

Caniuse.com. (2019). *Can I use... Support tables for HTML5, CSS3, etc*. [online] Available at: https://caniuse.com/#search=grid [Accessed 6 Jun. 2019].

Caniuse.com. (2019). *Can I use... Support tables for HTML5, CSS3, etc*. [online] Available at: https://caniuse.com/#search=flexbox [Accessed 6 Jun. 2019].

Food2Fork. (2019). *Recipe API | Food2Fork*. [online] Available at: https://www.food2fork.com/about/api [Accessed 11 Apr. 2019].

Bigoven. (2019). *Documentation | BigOven API*. [online] Available at: http://api2.bigoven.com/web/documentation [Accessed 11 Apr. 2019].

Holland, B. (2018). *Visual Studio Code Can Do That? — Smashing Magazine*. [online] Smashing Magazine. Available at: https://www.smashingmagazine.com/2018/01/visual-studio-code/ [Accessed 16 Apr. 2019].

Vuejs.org. (2019). *Installation — Vue.js*. [online] Available at: https://vuejs.org/v2/guide/installation.html [Accessed 5 May 2019].

Docs.npmjs.com. (2019). *About npm | npm Documentation*. [online] Available at: https://docs.npmjs.com/about-npm/ [Accessed 25 May 2019].

Marketplace.visualstudio.com. (2019). *Vetur - Visual Studio Marketplace*. [online] Available at: https://marketplace.visualstudio.com/items?itemName=octref.vetur [Accessed 5 May 2019].

npm. (2019). *axios*. [online] Available at: https://www.npmjs.com/package/axios [Accessed 5 Jun. 2019].

Gore, A. (2018). *7 Vue.js Backends Compared*. [online] 7 Vue.js Backends Compared. Available at: https://vuejsdevelopers.com/2018/05/07/vue-js-backends-express-laravel-firebase-wordpress-django-rails/ [Accessed 12 Jul. 2019].

Expressjs.com. (2019). *Express - Node.js web application framework*. [online] Available at: https://expressjs.com [Accessed 11 Jul. 2019].

npm. (2019). *express-generator*. [online] Available at: https://www.npmjs.com/package/express-generator [Accessed 11 Sep. 2019].

Heroku.com. (2019). *About Heroku | Heroku*. [online] Available at: https://www.heroku.com/about [Accessed 6 Aug. 2019].

Docs.mongodb.com. (n.d.). *Install MongoDB Community Edition on Windows — MongoDB Manual*. [online] Available at: https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows/ [Accessed 30 Jul. 2019].

# Appendices

## Appendix A

Interviewer: Hi [name]!  Just to make you aware of the rights as a participant in my interview, you have the right to terminate at any time, your data will be used by me for my EMA and a transcript included in my TMA02 document, that will be viewed by my tutor.

Interviewee: Ok.

Interviewer: First of all, do you currently use any  recipe finding applications or websites?

Interviewee: I do use websites that I find via Google, but I never stick to one.

Interviewer: So you literally just search for recipes using the search engine rather than a particular site?

Interviewee: Yeah, just because there’s never all the things that I want really on one website.  If there was one that I would say I go to most it’s probably the BBC food website

Interviewer: So when you say there’s not one that has all the things you want in one place, what are all the things you want?

Interviewee: Recipes for different meals like dinner, breakfast etc, also baking … like a little bit of everything.  You find websites for baking, but then can’t use the same for other recipes.

Interviewer: So it’s not that the interface of the application that doesn’t offer you what you want, it’s just they don’t have the recipes you want?

Interviewee: Uh-huh

Interviewer: Ok … So you literally just link through to a recipe from Google, you don’t use a search function on the websites?

Interviewee: Uh yeah, I do occasionally, it depends.  If I found a website, and I liked it, then I would use the search function on that website again, but I don’t always find one that’s useful.

Interviewer: So what features do you find useful in a recipe search, if you were to use one?

Interviewee: There is one, I can’t remember which one it is, but when you use it on a mobile it has little tabs, so you have the ingredients you need on a tab, and then you just click on the next tab at the top, then it’s got the method bullet pointed on it.

Interviewer: So that’s better for mobile view?

Interviewee: Uh-huh

Interviewer: What about when you’re actually searching?  So by ingredient, or recipe name or …

Interviewee: Um, I think sometimes it hard to search by ingredient, so I usually just search by a name, but I think if you could search by ingredients that would be more useful.

Interviewer:  Do you mean it’s hard because the website or app doesn’t do it easily

Interviewee: It just doesn’t recognise what you want

Interviewer: OK.  So if you could make yourself a recipe finder that did everything that you want, what would be the features that you would include as a priority?

Interviewee: As we were just saying, search … so you could just put in … like if you opened your cupboard and you had chicken and noodles and a few random things then you could type in what you had and then it would give you a recipe, not necessarily with everything you had, but roughly based on then ingredients that you have.

Interviewer: Cool, and what about the way it would return you your results, would you like just a list, or pictures …

Interviewee:  Definitely pictures.  I wouldn’t ever make anything if there wasn’t a picture.

Interviewer:  OK,

Interviewee:  A list with a picture is fine.

Interviewer: And you said on mobile that the tabs on that particular site are good for mobile, how about on like a normal computer monitor, would you want a different layout, or tabs,

Interviewee:  Um, if it was a computer then I’d probably go for the layout like all on one page, so you can see it all at once.  But I just think on a phone that that would be too small and you’d constantly be zooming in.  But I think if you’ve got a bigger screen and you can see it all at once then it would be a lot easier.

Interviewer: What about any friends and family?  DO you know if they use anything particular [for recipe finding], anyone got a favourite that you know of?

Interviewee:  Yeah they do use them, but I think they’re just a bit the same as me, like there’s never one that they can always find stuff on, so they just go everywhere.

Interviewer:  Cool … so in your household, what happens with food that’s going out of date, or has maybe just gone passed the sell-by date?

Interviewee:  It all gets thrown away.  We throw away so much food

Interviewer: And can you think of anything that would encourage you to not throw it away?

Interviewee:  If I knew that it was gonna go out of date, then I would use it, but sometimes it just sits in the fridge doesn’t it, you buy something and you forget it’s in there, and then you end up not using it, and it goes in the bin.  So if you knew that that was coming up then I’d definitely use it up.

Interviewer:  Cool, that’s it then, thank you so much for your participation

## Appendix B

1. Allow viewing of a virtual store cupboard, based on the logged in user.
2. Allow the addition of an item to a virtual store cupboard, for a specific user.
3. Allow the deletion of an item from a virtual store cupboard, for a specific user.
4. User can add multiple ingredients to a list and search a recipe database based on this list
5. User can Add store cupboard ingredients to ingredients list when searching the recipe database.
6. User can view a list of available recipes based on selected ingredients
7. Allow user to sort and filter searched recipes
8. Show method and ingredients list of a selected recipe
9. Display information about food safety
10. Allow searching of information related to food safety
11. A new user to the site should be able to sign up for an account on the website
12. An existing user to the site should be able to log in

## Appendix C

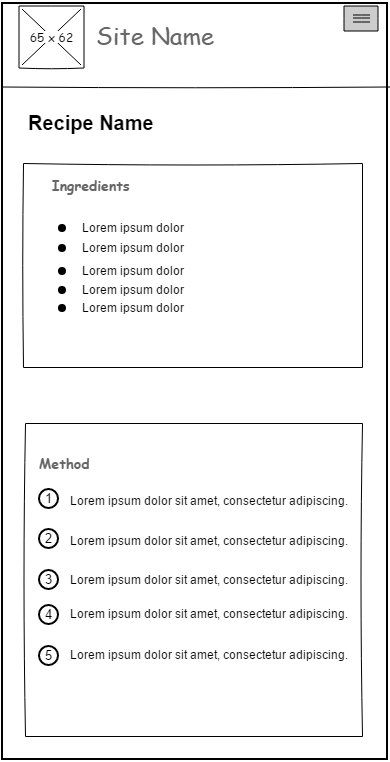


Figure 1. View Recipe mobile view

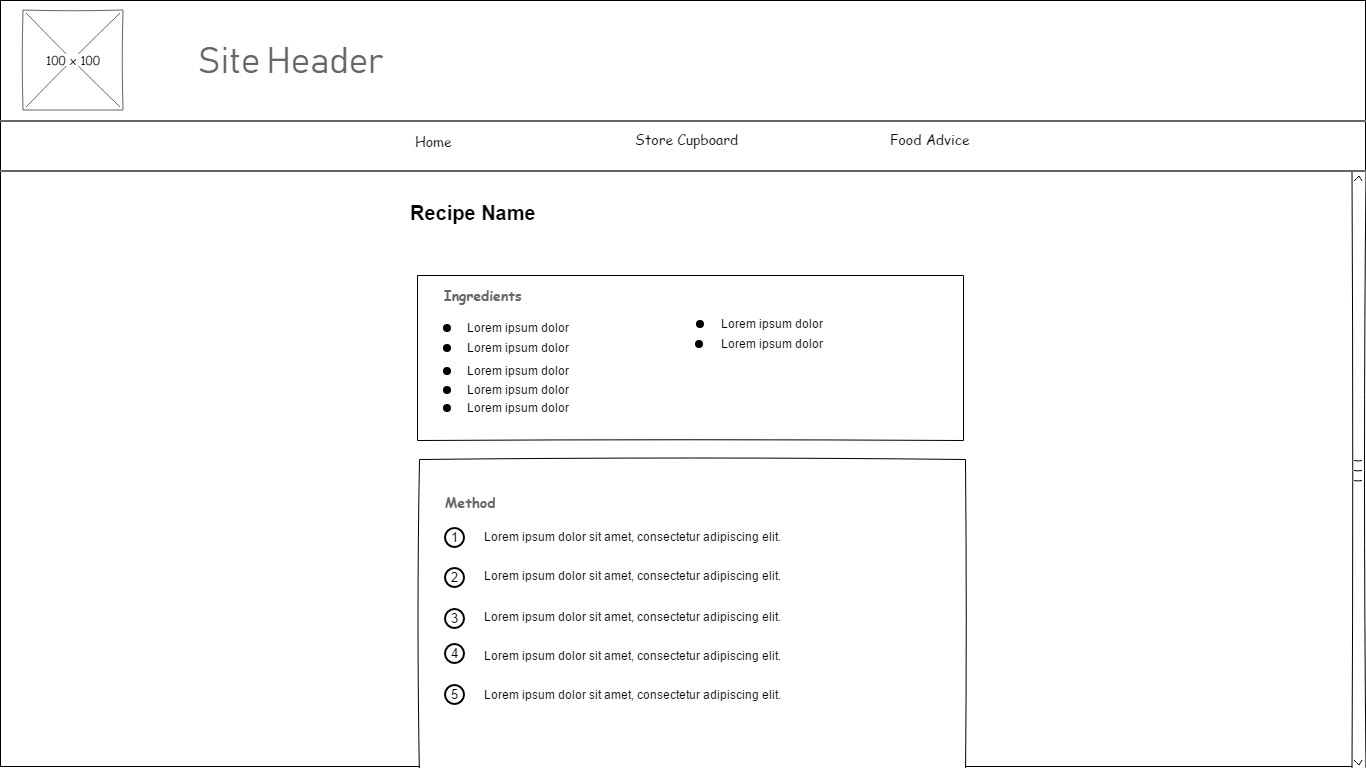


Figure 2. View Recipe desktop view

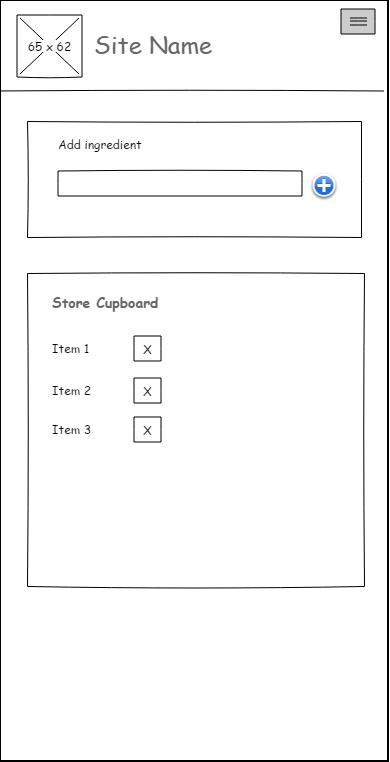


Figure 3. Store Cupboard mobile view

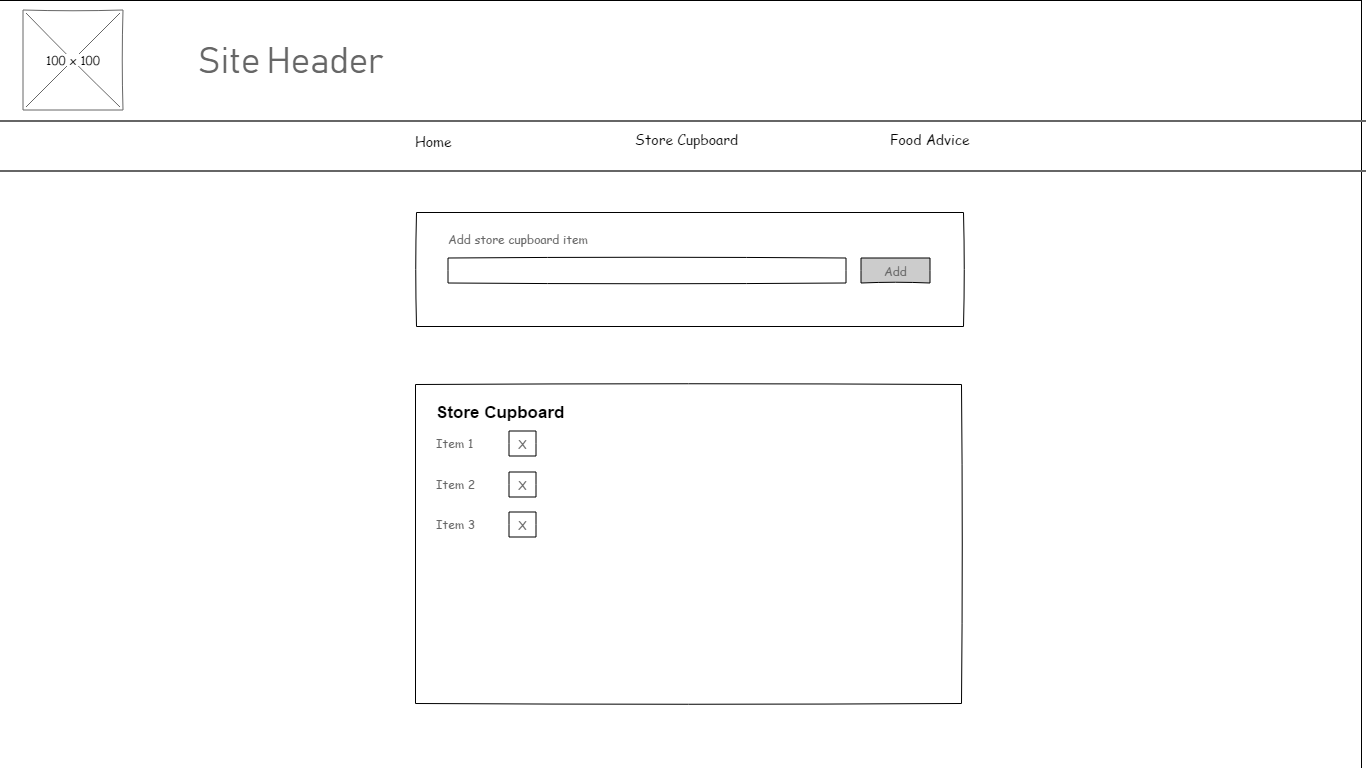


Figure 4. Store Cupboard desktop view

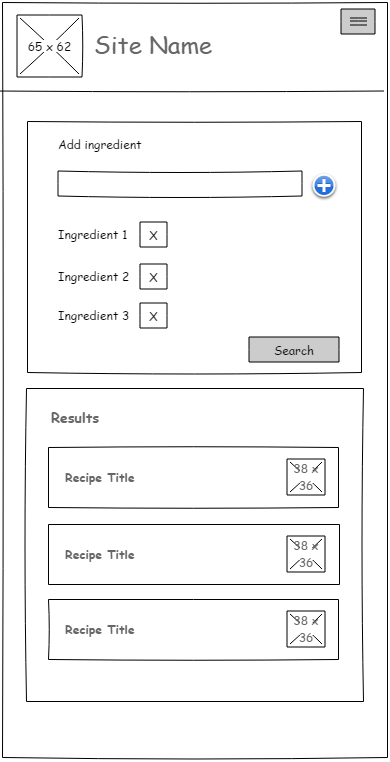


Figure 5. Recipe Search mobile view

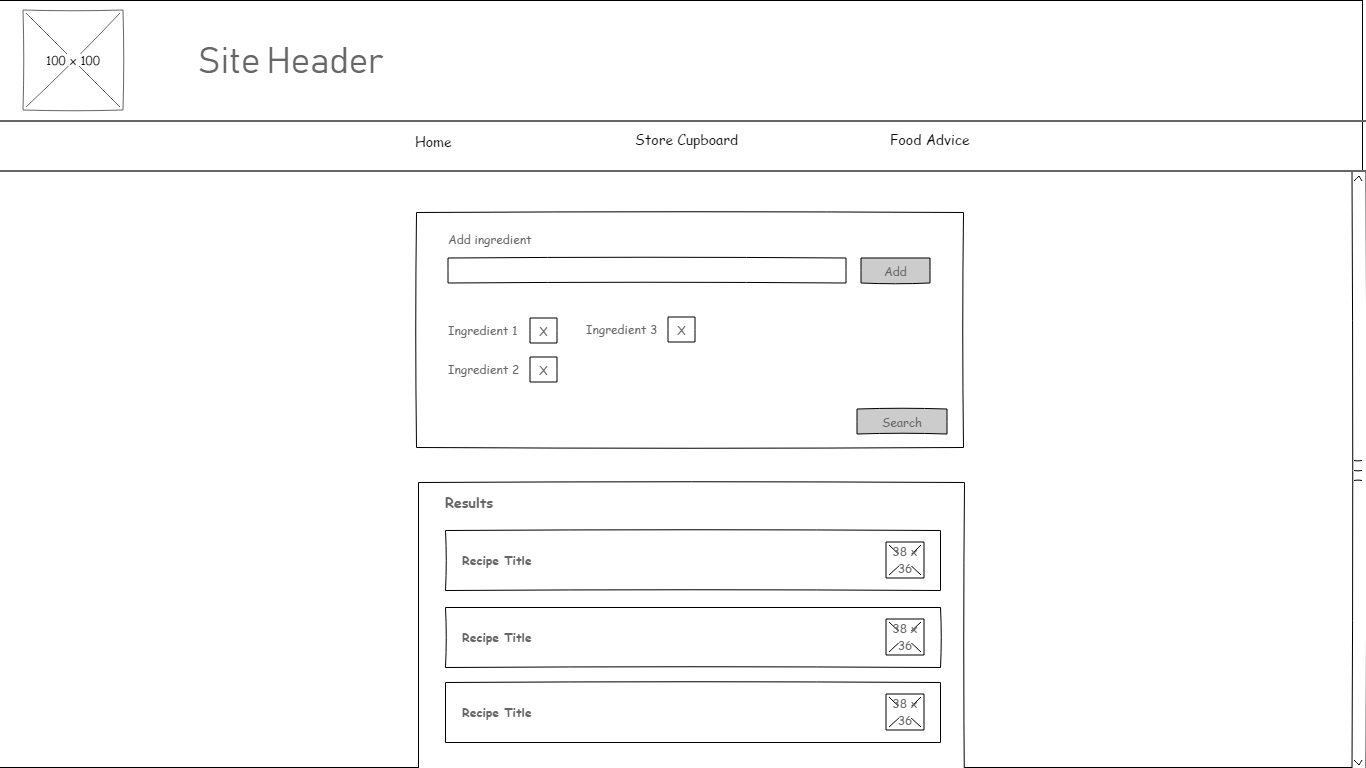


Figure 6. Recipe Search desktop view

## Appendix D

|  |  |
| --- | --- |
| Test ID | T001 |
| Description | Test user can create account |
| Setup |  |
| Instructions | Click on ‘Sign Up’ link in navigation bar. Enter required details and click ‘Sign Up’ button |
| Expected Results | A message should be displayed informing the user that an account has been created |

|  |  |
| --- | --- |
| Test ID | T002 |
| Description | Test user can login |
| Setup | Create account if not already done so |
| Instructions | Click on ‘Log in’ link in navigation bar. Enter required details and click ‘Log In’ button |
| Expected Results | A message should be displayed informing user that they have been logged in |

|  |  |
| --- | --- |
| Test ID | T003 |
| Description | Test user cannot login with incorrect information |
| Setup | Create account if not already done so |
| Instructions | Click on ‘Log in’ link in navigation bar. Enter required incorrect username or password |
| Expected Results | A message should be displayed informing user that they cannot be logged in |

|  |  |
| --- | --- |
| Test ID | T004 |
| Description | Test user can add Store Cupboard item to store cupboard |
| Setup | Login via the login page. Navigate to ‘Store Cupboard’ page |
| Instructions | Enter ‘garlic’ as a value in input field, then click Add button |
| Expected Results | A new entry for ‘garlic’ should appear in listed store cupboard ingredients. If ‘garlic’ already exists in the ingredients list then an error should be displayed |

|  |  |
| --- | --- |
| Test ID | T005 |
| Description | Test user can view Store Cupboard |
| Setup | Login via the login page. Add a new item to the store cupboard if it is currently empty |
| Instructions | From ‘Home’, click ‘Store Cupboard link. |
| Expected Results | If the user has items in the Store Cupboard, a list of these items should be displayed. If not, a message should be displayed indicating as such |

|  |  |
| --- | --- |
| Test ID | T006 |
| Description | Test user can delete an item from the Store Cupboard |
| Setup | Add a new item to the store cupboard if it is currently empty |
| Instructions | From ‘Home’, click ‘Store Cupboard link. Select an existing item in the Store Cupboard list and click the ‘Delete’ button, indicated with a ‘X’ |
| Expected Results | The deleted item should be removed from the Store Cupboard list displayed |

|  |  |
| --- | --- |
| Test ID | T007 |
| Description | Test user can add multiple ingredients to ingredients search list |
| Setup |  |
| Instructions | On ‘Home’ page, enter ingredient into text box, then click ‘Add Ingredient’ button. |
| Expected Results | The added ingredient should appear in Ingredients list on the page. |

|  |  |
| --- | --- |
| Test ID | T008 |
| Description | Test users can view a list of recipes based on entered ingredients |
| Setup |  |
| Instructions | On ‘Home’ page, enter ingredient into text box, then click ‘Add Ingredient’ button. Click ‘Search’ button |
| Expected Results | A list of recipes should be displayed, relevant to the ingredients entered in search |

## Appendix E

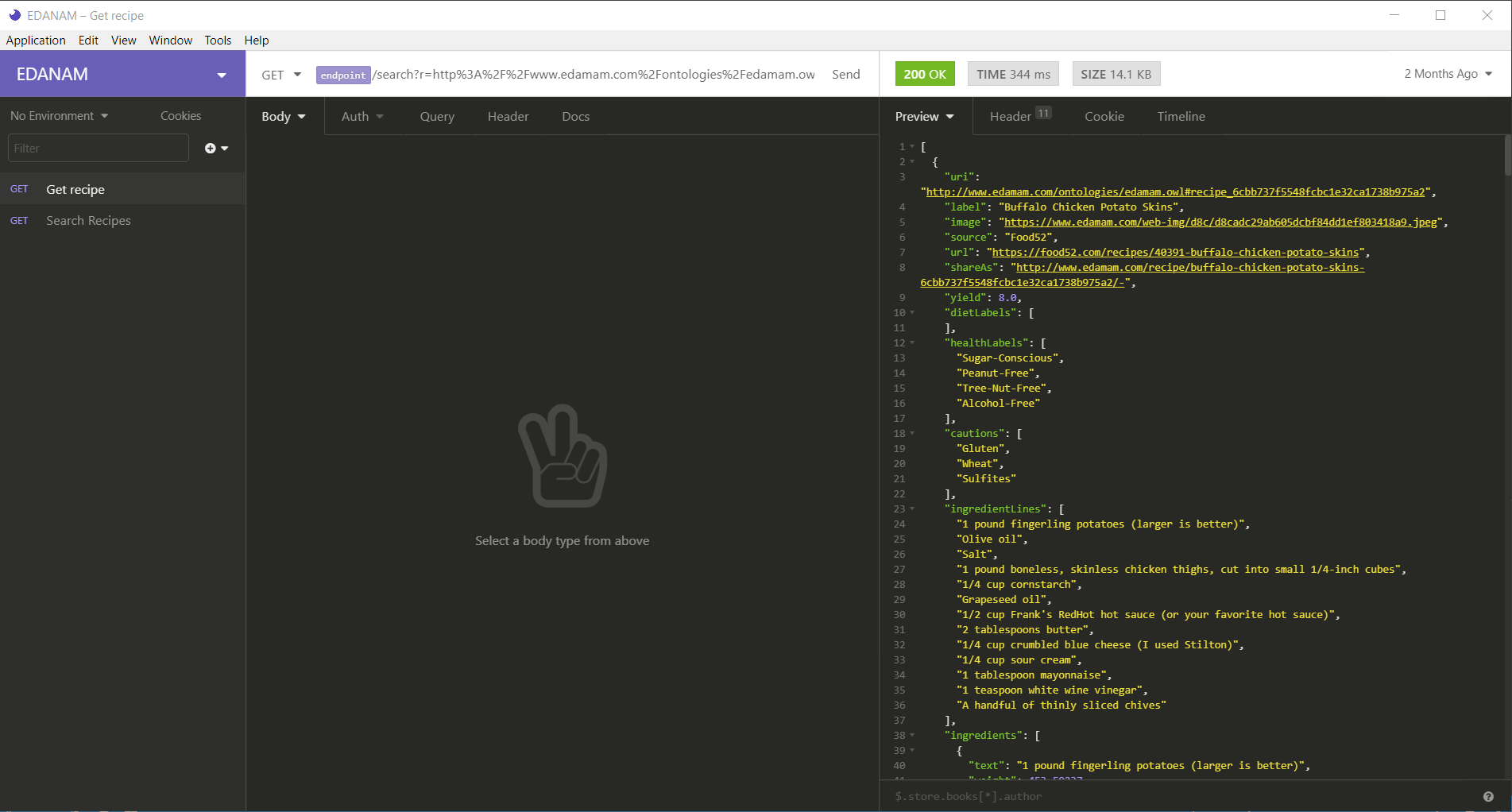


Figure 7. Insomnia workspace demonstrating Edamam API

## Appendix F

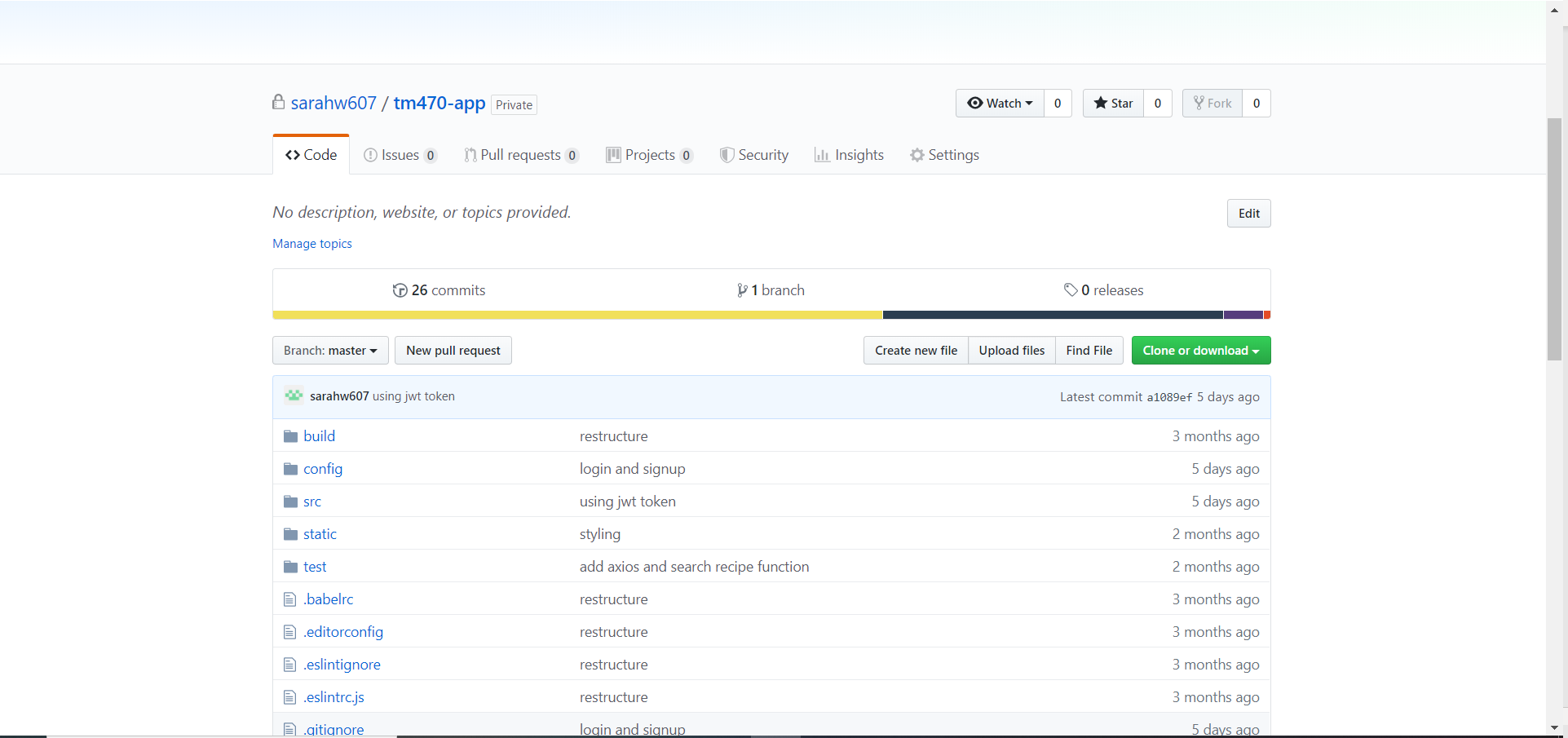


Figure 8. Git repository for Vue.js application

## Appendix G

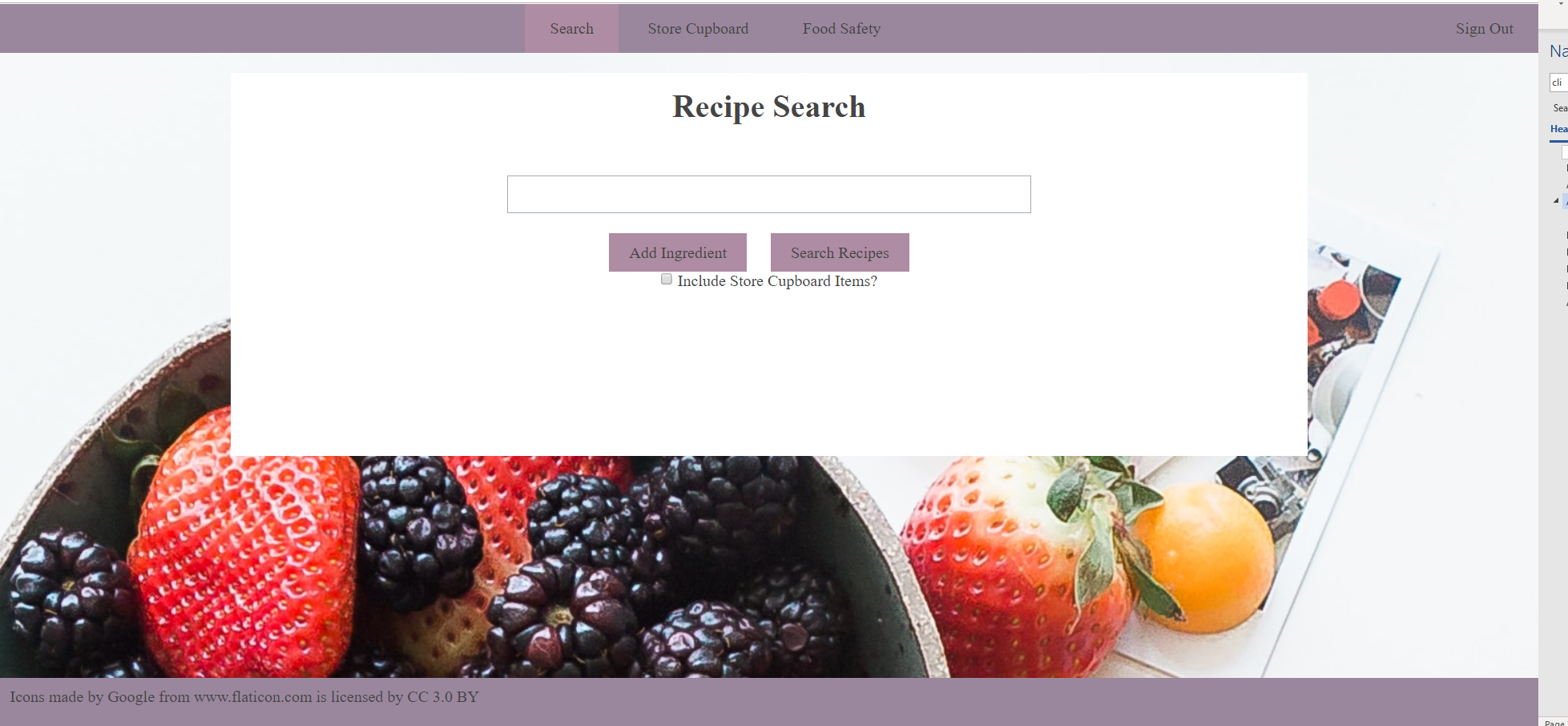


Figure 9. Application showing the navigation bar on a desktop computer

## Appendix H

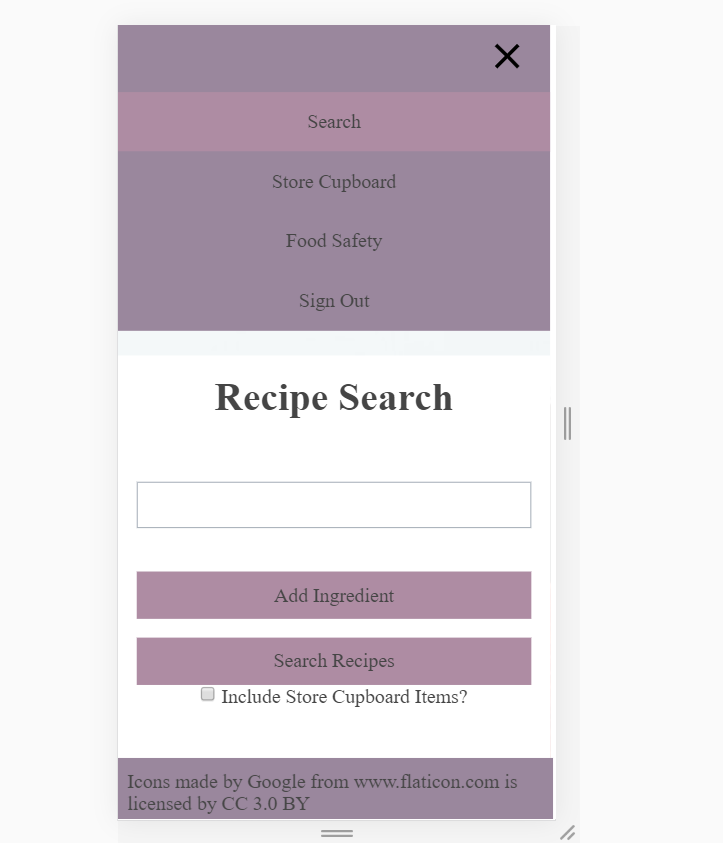


Figure 10. Application showing the navigation bar on a mobile device

## Appendix I

<template>

  <nav>

    <ul class="menu">

      <li class="toggle" v-on:click="toggleMenuIcon()">

        <a href="#">

          <img

            v-bind:class="{ hidden: !showMenu }"

            class="icon"

            src='@/assets/img/close-button.svg'

          />

          <img v-bind:class="{ hidden: showMenu }" class="icon" src="@/assets/img/menu-button.svg" />

        </a>

      </li>

      <div v-bind:class="{ hidden: !showMenu }" class="menu-content">

        <router-link to="/" tag="li" class="item">

          <a>Search</a>

        </router-link>

        <router-link to="/cupboard" tag="li" class="item" v-if="isLoggedIn()">

          <a>Store Cupboard</a>

        </router-link>

        <router-link to="/safety" tag="li" class="item">

          <a>Food Safety</a>

        </router-link>

      </div>

      <div class="last" v-if="!isLoggedIn()" v-bind:class="{ hidden: !showMenu }">

        <router-link to="/login" tag="li" class="item">

          <a>Log In</a>

        </router-link>

        <router-link to="/signup" tag="li" class="item">

          <a>Sign Up</a>

        </router-link>

      </div>

      <div>

        <li

          class="item"

          v-if="isLoggedIn()"

          v-on:click="logOut()"

          v-bind:class="{ hidden: !showMenu }"

        >

          <a v-on:click="logOut()">Sign Out</a>

        </li>

      </div>

    </ul>

  </nav>

</template>

<script>

export default {

  name: 'AppNavbar',

  data: function () {

    return {

      showMenu: false

    }

  },

  methods: {

    toggleMenuIcon: function () {

      this.showMenu = !this.showMenu

    },

    isLoggedIn: function () {

      return (

        localStorage.getItem('user') !== undefined &&

        localStorage.getItem('user') !== null

      )

    },

    logOut: function () {

      console.log('log out')

      localStorage.removeItem('user')

    }

  }

}

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<style scoped>

nav {

  background-color: #9a879d;

}

ul {

  list-style-type: none;

}

li {

  display: inline-block;

  height: 100%;

}

.menu {

  display: flex;

}

.menu-content {

  flex: 1;

  text-align: center;

}

.menu li {

  padding: 15px 25px;

  white-space: nowrap;

}

.active {

  background-color: #ae8ca3;

}

.icon {

  height: 20px;

  width: 20px;

}

.toggle {

  margin-left: auto;

  display: none;

}

.last {

  margin-left: auto;

}

@media only screen and (min-device-width: 320px) and (max-device-width: 480px) {

  .menu {

    flex-direction: column;

    align-items: center;

    flex-wrap: wrap;

    justify-content: space-between;

  }

  .menu-content {

    width: 100%;

  }

  .toggle {

    display: block;

  }

  .item {

    width: 100%;

  }

  .hidden {

    display: none;

  }

}

</style>

## Appendix J

<template>

  <div>

    <form>

      <div>

        <input v-model="newIngredient" type="text" />

      </div>

      <div>

        <button type="button" v-on:click="addIngredient()">Add Ingredient</button>

        <button type="button" v-on:click="searchRecipes()">Search Recipes</button>

      </div><div>

        <input type="checkbox" v-model="includeStore"/>

        <label>Include Store Cupboard Items?</label>

      </div>

    </form>

    <ingredient-list v-bind:ingredientList="ingredients"></ingredient-list>

    <div>{{error}}</div>

    <recipe-list v-bind:recipeList="recipes"></recipe-list>

  </div>

</template>

<script>

import IngredientList from '../shared/IngredientList.vue'

import RecipeList from './RecipeList.vue'

import axios from 'axios'

export default {

  name: 'RecipeSearch',

  data: function () {

    return {

      ingredients: [],

      newIngredient: '',

      error: '',

      recipes: [],

      includeStore: false

    }

  },

  methods: {

    addIngredient: function () {

      this.error = '';

      (this.ingredients.indexOf(this.newIngredient) === -1 && this.newIngredient.length > 1)

        ? this.ingredients.push(this.newIngredient)

        : this.error = 'You cannot add the same ingredient twice'

      this.newIngredient = ''

    },

    searchRecipes: function () {

      if (this.ingredients.length < 1) {

        this.error = 'Please add some ingredients to search'

        return

      }

      let apiPath = ''

      if (this.includeStore) {

        const ingredientApi = `${process.env.ROOT\_API}/users/ingredients`

        axios.get(ingredientApi, { method: 'GET', mode: 'CORS', headers: { 'Authorization': 'Bearer ' + JSON.parse(localStorage.getItem('user')) } })

          .then(response => {

            const ingredientsString = [...response.data.ingredients.map(ing => ing.name), ...this.ingredients]

            apiPath = `${process.env.RECIPE\_API}/search?q=${ingredientsString}&app\_id=77782426&app\_key=04992e180e5fa5497e347529b8570e88`

            axios.get(apiPath).then(results => {

              if (results.data.hits.length < 1) {

                this.error = 'No matching recipes were found'

              } else {

                this.recipes = results.data.hits

              }

            })

          }).catch(response => {

            this.message = response.message

          })

      } else {

        apiPath = `${process.env.RECIPE\_API}/search?q=${this.ingredients.join(',')}&app\_id=77782426&app\_key=04992e180e5fa5497e347529b8570e88`

        axios.get(apiPath).then(response => { this.recipes = response.data.hits })

      }

    }

  },

  components: { IngredientList, RecipeList }

}

</script>

## 

