# CHAPTER ONE INTRODUCTION

**1.1 Background of the Study**

In recent years, technological improvements have benefited the market of easily accessible websites for meeting their consumers' daily demands. Cooking is one such requirement. As a result, recipe websites are proliferating on the internet, overloading the space with extravagant amounts of data. As a result, users may become overwhelmed by the sheer volume and find themselves unable to find the information they were looking for in the first place. (Lemonaki & Beka, 2022).

Millions of cooking recipes are now available online through culinary-sharing sites such as AllRecipes, Cookpad, and Yummly, among others. A recipe is typically provided in a multimedia format, with verbal descriptions of cooking stages paired with culinary photographs to demonstrate the visual result of each step. (Liangming et al., 2020)

The concept of online food recipe websites has been around since the early days of the internet which allowed users to search for and share their own recipes with others. In the years since the number of online food recipe websites has grown exponentially. According to a 2018 report by Statista, there were over 8,000 food and recipe websites in the United States alone (Statista, 2018). These websites have become increasingly popular as they offer a convenient way for people to find and share recipes, as well as discover new dishes and cooking techniques.

In addition to traditional recipe websites, the rise of social media has also contributed to the growth of online food recipe sharing. Platforms such as Pinterest and Instagram have become popular places for users to share their own recipes, as well as discover new ones through hashtags and food-related accounts.

Cooking is a process that often becomes a great pastime for many people. The realities of today are such that many cooking schools, chefs, and even amateurs share and offer their services through the web so they need a nice-looking website to deliver their awesome content. Because with content this much attractive by itself, using beautiful cooking website templates will allow you to achieve a real synergy and really rock on the web (Kosych, 2021).

Cooking websites are the perfect way of connecting cooking aficionados and sharing a new experience. it provides plenty of recipes with a user-friendly UI/UX and an editor that anyone can use to customize their culinary palate according to personal needs (Kosych, 2021).

A cooking website is a web resource that is promoted according to information requests on culinary and food topics. Besides presenting your services and sharing helpful information on cooking, you can even monetize your site and get additional money (by contextual advertising, advertising various culinary services, affiliate programs, and other retail marketing strategies) (Kosych, 2021).

**1.2 Statement of the Problem**

Developing the knowledge and motivation for healthy eating early in life remains a challenge. While eating patterns tend to form before and during young adulthood, individuals may have little opportunity to develop their food literacy until they begin to live independently (Colatruglio et al., 2016). This has resulted in the need for fast food by most young adults causing unhealthy eating or eating disorder.

**1.3 Aim and Objectives of the Study**

This project aims to develop a recipe website that provides its users with different types of recipes based on their taste and even in case of need for certain dietary needs in the case of diabetes.

**The objectives are:**

1. An engaging and easy-to-use UI with good UX using HTML.
2. The logic will be handled using JavaScript and the data will be stored using MySQL.
3. Evaluating and verifying the app.

**1.4 Scope of the Project**

The area covered in this research work focuses on recipes not limited to a particular region in the world but based on recipes for certain dietary needs like for diabetes patients.

**1.5 Limitations of the Study**

The platform is limited to recipe guide provision in the format of text without a video tutorial. The study’s scope has also been constrained by time, and the researcher's busy academic pursuits severely limited the time allotted for research for this study.

**1.6 Significance of the Study**

The website will offer a convenient and easily accessible way for people to find and share recipes, and it can serve as a valuable resource for individuals and families looking to plan their meals and try new dishes, as well as providing users with certain dietary needs with the options of seeing the food available to them.

**1.7 Project Organization**

The project is divided into three chapters. The outlines are presented below:

**Chapter One: Introduction**

Chapter one summarizes the introductory study on the project work, the background of the study, the statement of the problem, the aim and objectives, the scope of the study, the limitations of the study, the significance of the study, the project organization, and the definition of terms.

**Chapter Two: Literature review**

This chapter focuses on the literature review, and the contributions of other scholars on the subject matter being discussed.

**Chapter Three: Methodology and Design**

This chapter is concerned with the presentation of the results of system analysis and design. It presents the research methodology used in the development of the system to facilitate an understanding and effective future implementation of the system.

**Chapter Four: System Implementation Evaluation**

This chapter describes the system implementation and documentation, analysis of modules, and system requirements for implementation.

**Chapter Five: Summary, Conclusion, and Recommendation**

The chapter provides a summary of major findings, conclusions, and recommendations based on the study conducted

**1.8 Definition of Terms**

1. **Recipe:** A set of instructions for preparing a specific dish, including a list of ingredients and their quantities, and the steps involved in cooking or preparing the dish.
2. **Dietary restriction:** A constraint on an individual's diet due to health, cultural, or personal reasons. This can include allergies, intolerances, or preferences such as vegetarian or vegan diets.
3. **Cuisine:** A style or type of cooking characterized by a specific set of ingredients, techniques, and cultural influences.
4. **User interface (UI):** The part of a software application that the user interacts with, including the layout, buttons, and other elements**.**
5. **User experience (UX):** The overall experience of a user interacting with a product or service, including their emotions, perceptions, and behaviours**.**
6. **Search tool:** A feature on an online food recipe website that allows users to locate specific recipes by entering keywords or using filters such as ingredient, cuisine, or dietary restriction.
7. **Rating and review:** A feature on an online food recipe website that allows users to rate a recipe and leave a written review, providing feedback on the recipe and sharing their experiences with it.

**CHAPTER TWO**

**LITERATURE REVIEW**

**2.1 Introduction**

This chapter seeks to show how the topic under investigation links to earlier research, current practice, or other fields of knowledge by citing relevant publications by other researchers that have addressed a related issue. Furthermore, this chapter will give a synthesis of current research on the topic, noting areas of agreement, disagreement, and gaps in the literature, to demonstrate the project topic's importance in the field and to identify prospects for further research.

**2.2 Literature Review**

Ratatouille: A tool for Novel Recipe Generation. (Goel et al,. 2022). The challenge of automatic recipe production is both interesting and demanding because recipes must create ingredients and cooking instructions depending on the user's desired list of components. To generate such recipes, computational linguistics and machine learning are used. The evaluation of such recipes is currently difficult, and available metrics only offer a limited view of the quality of the created recipes.

Moreso, the web application's architecture is based on the Python framework. ReactJS, a simple and lightweight library, is utilized for the front end, and Flask is used for the back end. The front end is detached from the backend utilizing microservices architecture to manage more user requests and prevent application disruption. The front end and back end are dockerized and hosted independently on the webserver. If the load increases in the future, the developer merely needs to repeat the docker.

In conclusion, this paper, we employed a neural network-based LSTM model and a transformer-based GPT2 model to generate innovative recipes from a list of ingredients. We recorded the number of ingredients in each dish, which was not done in previous experiments. With a BLEU score of 0.806, the transformer-based GPT2 model surpasses the neural network-based LSTM model. The most significant problem we encountered was analyzing the created recipes. In the future, we plan to adopt GPT-Neo, which is based on the same architecture as GPT-3.

The biggest problem encountered throughout this research was a lack of funding and hardware. On Google Colab, we have restricted GPU, RAM, and disk space hours, which causes sessions to fail every 5 to 7 epochs.

The Chef’s Choice: System for Allergen and Style Classification in Recipes. (Andreas et al,. 2022). Individuals who have allergic sensitivities to specific food allergies must exercise caution while browsing for recipes online, since it may be a matter of life and death for some. With the growth in the frequency of food allergies, it is becoming increasingly vital to take further steps to educate people with food allergies. A system that assists in choosing recipes might save time and minimize the frustration of affected persons. Instead of creating a specialized app into which the text must be transferred, integration with current technologies such as a web browser would save time and aid in scanning.

The following are the prerequisites for our suggested system:

1. Recipe style categorization
2. Allergy classification with ingredients.
3. Warnings for custom ingredients
4. Platform-independent architecture
5. Browser extension for common internet browsers.

Moreso, using the elements in a recipe, the suggested system should be able to identify style and allergies. A browser plugin should provide the findings together with the confidence for the multi-class classification (recipe style) and a list of allergy predictions with confidence for the multi-label classification (allergens in a recipe). Furthermore, the user should be able to define a custom list of distinct elements that should be taken into account while analyzing. Custom ingredients should provide a distinct warning than allergy detection.

In conclusion, this article sought to evaluate what an allergy and style detection system in recipes may look like, as well as how well a trained classifier works versus people. Given the abundance of allergies and components, a system that identifies allergens may surely assist the user in identifying whether or not internet recipes contain allergens. Furthermore, a system that is readily connected with existing technologies such as internet browsers might enhance system utilization and raise overall allergy awareness.

A Recipe for Food Literacy: Designing and Evaluating Technologies for Informed Food Choices. (Marcela, 2022). This thesis combines Human-Computer Interaction (HCI) with Nutrition Science. It studies how technologies, by researching and integrating technological features and nutrition content in their designs, might encourage food literacy and educated eating choices.

Furthermore, sessions were held online and directly recorded using Microsoft TeamsTM. To acquire demographic data, Qualtrics XM was utilized. The evaluation and interview took an average of 60 minutes for each session. Interviews were conducted until data saturation was attained; saturation was defined as the absence of fresh material received by questioning further individuals that would contribute to the formulation of new themes.

In summary, our first study was to develop and verify a set of food literacy criteria for technology design. We were able to establish heuristics through an iterative design approach that can effectively and efficiently detect a variety of food literacy concerns that lie under the umbrella of knowledge, awareness, and skills. Furthermore, we demonstrated that the same heuristics might be useful as formative design tools for designers. They can assist designers in identifying food literacy difficulties within various technologies and apps, considering how such technologies may affect others' planning and buying decisions, and self-reflecting on their own challenges.

Exploring the Effects of Natural Language Justifications in Food Recommender Systems. (Cataldo et al., 2021). The concept of using individualized suggestions to help individuals eat healthier has gained popularity in recent years. This concept is being examined by a research line focusing on health-aware food recommender systems, which take into account user information such as dietary preferences and limitations (e.g., allergies) to build an appropriate meal plan. The fundamental issue is that most popular internet-sourced recipes utilized in recommendation systems are unhealthy, and as a result, people prefer them.

Moreso, dataset. Recipes were drawn from a database of 4,671 recipes available online. Recipes were gathered and translated into English from a popular food community platform. The recipes included information such as their name, category, preparation difficulty, ingredients, (macro-) nutrients, calories, rating count, and average website rating. Furthermore, they featured binary markers such as vegetarian, vegan, lactose-free, and low-nickel.

Finally, we want to underline that the study may be used as a model for future research on healthy diet recommendations. We demonstrated that our algorithm successfully provides healthy suggestions since users who choose them reported that they did so for health-related reasons. Furthermore, we demonstrated how such advice should be presented to assist healthy eating choices.

Challenges to acquiring and using food literacy: Perspectives of young Canadian adults. Obesity rates have risen in tandem with the growing consumption of processed and ultra-processed, low-nutrient, and energy-dense foods, such as sweetened drinks, which are often mass-produced, aggressively promoted, and widely available. This has also contributed to greater food consumption away from home and eating outside of typical meal arrangements (eating at the kitchen table in households or eating at regular times of day). (Sarah & Joyce, 2022)

Moreso, to acquire eligible individuals, purposeful and theoretical sampling strategies were utilized. Individual interviews were conducted in a separate room and took between 30 and 45 minutes to complete. A semi-structured interview guide with face validity testing was employed. Demographic data was also collected.

In conclusion, this study shed light on the obstacles and opportunities for obtaining and applying food literacy as seen through the eyes of Canadian young people who have just moved to independent living. Possible solutions to these difficulties are many and multidimensional and will require more study into personal, societal, and environmental aspects, as well as health-related consequences.

**2.3 Summary of Related Literature Reviews**

|  |  |  |
| --- | --- | --- |
| **Author & Year** | **Title & Description** | **Merit and Demerits** |
| Goel et al. (2022). | Ratatouille: A tool for Novel Recipe Generation.  Novel Recipe Generation is a problem in the field of Natural Language Processing in which the main interest is to generate realistic, novel cooking recipes. | the transformer-based GPT2 model surpasses the neural network-based LSTM model.  The system architecture is based on GPT-2. |
| Andreas et al. (2022) | The Chef’s Choice: System for Allergen and Style Classification in Recipes.  The goal is to develop a system that can be used for online recipe sites and adds an extra degree of security by detecting allergies and providing more information about discovered allergens and the recipe style. | The method assists individuals in selecting the proper recipe before the user inspects the recipe more closely to ensure there are no allergies present.  The machine learning process suffered from a lack of dataset, data quality, and class imbalance. |
| Marcela (2022). | A Recipe for Food Literacy: Designing and Evaluating Technologies for Informed Food Choices.  The study looks at how alternative technology designs combining food literacy ideas impact food choices in a grocery store using a contextual approach. | The results of the study suggest that the approach can be effective.  Despite the importance of cost, neither MFG nor PBGA directly addresses food insecurity. |
| Cataldo et al. (2021). | Exploring the Effects of Natural Language Justifications in Food Recommender Systems.  This study provides a way to produce and convey a natural language reasoning that emphasizes the nutritional content, or health risks and advantages of recommended meals, to encourage users to pick healthier suggestions by making more educated eating selections. | The system algorithm provides healthy suggestions to its users.  Delay in providing suggestions. |
| Sarah and Joyce. (2022). | Challenges to acquiring and using food literacy: Perspectives of young Canadian adults.  The purpose of the study was to investigate the idea of food literacy through the eyes of young Canadian people who had just moved to independent living. | It improved our understanding of participants' eating experiences, including obstacles to gaining food literacy.  The limited sample size of Canadian university students in Western Canada, which may not be indicative of the perspectives and experiences of young adults from various backgrounds and geographical places. |

**2.4 HTML**

HTML stands for HyperText Markup Language. It is a standard markup language for web page creation. It allows the creation and structure of sections, paragraphs, and links using HTML elements (the building blocks of a web page) such as tags and attributes (Astari, 2023).

HTML has a lot of use cases, namely:

1. **Web development**. Developers use HTML code to design how a browser displays web page elements, such as text, hyperlinks, and media files.
2. **Internet navigation**. Users can easily navigate and insert links between related pages and websites as HTML is heavily used to embed hyperlinks.
3. **Web documentation**. HTML makes it possible to organize and format documents, similarly to Microsoft Word.

It’s also worth noting that HTML is not considered a programming language as it can’t create dynamic functionality. It is now considered an official web standard. The World Wide Web Consortium (W3C) maintains and develops HTML specifications, along with providing regular updates (Astari, 2023).

**2.5 JavaScript**

JavaScript is a lightweight programming language that web developers commonly use to create more dynamic interactions when developing web pages, applications, servers, and or even games (Jordana, 2022).

Developers generally use JavaScript alongside [HTML](https://www.hostinger.com/tutorials/html-cheat-sheet) and [CSS](https://www.hostinger.com/tutorials/css-cheat-sheet) The scripting language works well with CSS in formatting HTML elements. However, JavaScript still maintains user interaction, something that CSS cannot do by itself (Jordana, 2022).

The initial versions of the scripting language were for internal use only. After Netscape submitted it to [ECMA International](https://www.ecma-international.org/) as a standard specification for web browsers, JavaScript pioneered the release of ECMAScript (Jordana, 2022).

It was a general-purpose scripting language to ensure web pages’ interoperability across different browsers and devices (Jordana, 2022).

JavaScript has continued to grow alongside new browsers like Mozilla Firefox and Google Chrome since then. The latter even started developing the first modern JavaScript engine, called V8, which compiles bytecode into native machine code (Jordana, 2022).

Today, JavaScript has plenty of frameworks and libraries to simplify complex projects, such as [AngularJS](https://www.hostinger.com/tutorials/what-is-angular), [jQuery](https://www.hostinger.com/tutorials/what-is-jquery/), and [ReactJS](https://www.hostinger.com/tutorials/what-is-react) (Jordana, 2022).

Originally run on the client-side, the JavaScript implementation has branched out to the server-side after the Node.js development ‒ a cross-platform server environment built on the Google Chrome JavaScript V8 engine (Jordana, 2022).

While it caters to web-based programs the most, JavaScript programming features have other implementations in different areas (Jordana, 2022).

**CHAPTER THREE**

**METHODOLOGY AND DESIGN**

**3.1 Introduction**

A methodology is an approach to rigorous study or investigation, particularly to uncover new facts or information; hence, research methodology should be good enough to make the attainment of the established objectives attainable with certain components, such as methods of data collecting and design. This chapter includes the input/output specifications and system requirements for the development of a food recipe system, as well as the system modeling (use case, activity, and class diagrams).

**3.2 Methods of Data Collection**

It is crucial to acquire data and facts about the current system before implementing any system since one has to understand what is happening. Three techniques were used to conduct this study.

1. Observation of the Work Environment
2. Documentation

**3.2.1 Observation of the Work Environment**

This approach was used to collect information/data for this study by examining how the manual system was carried out, the method provides varying degrees of control over the context in which they are used, and the careful inspection revealed the obvious flaws in the present system.

**3.2.2 Documentation**

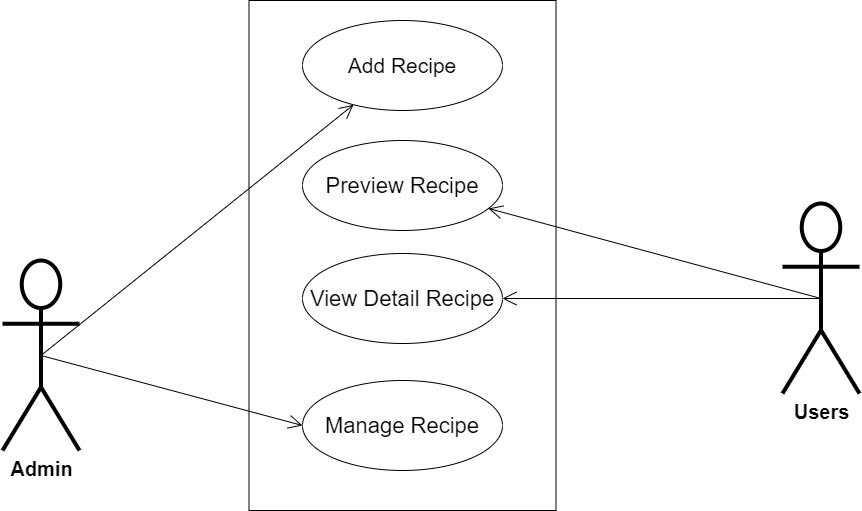
A secondary form of data acquisition is documentation. Journals, manuals, previous projects, publications, and other sources are used in this approach. This type of data collecting is employed because it provides a foundation for comparison with previous research. This includes the internet, a tool for gathering data. The internet was utilized to find information on topics that seemed challenging or unclear.

**3.3 System Modeling**

A system model is a conceptual model that describes and represents a system. Any interaction between a group of components that work together to accomplish a single goal is referred to as a system. Visual models of the object-oriented software-intensive systems can be made using a set of graphic notation techniques that are part of the Unified Modeling Language, which is employed in this modern system design. Use case diagrams, class diagrams, and activity diagrams are among the UML diagrams used in this new design.

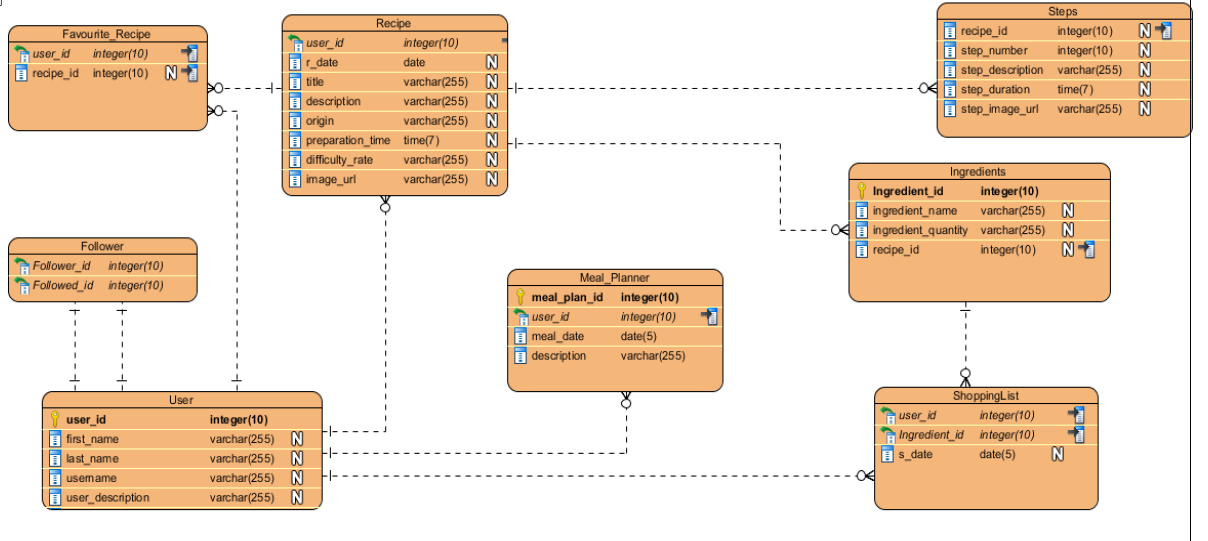
**3.3.1 Use Case Diagrams**

Use cases are collections of interactions between systems and users. Use case diagrams are used to visually summarize a system's functionality in terms of its actors, its goals (represented as use cases), and any dependencies between those use cases.



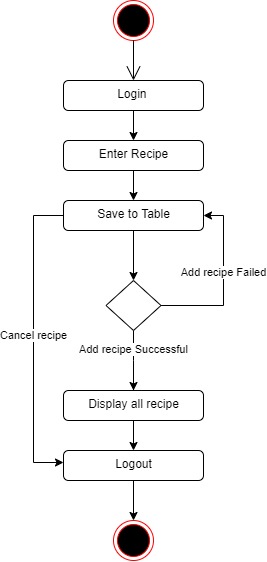
**Fig 3.1 System Use Case Diagram**

**3.3.2 Class Diagrams**

The Unified Modeling Language (UML) class diagram is an implementation of an independent view of how the system interface will be, with each class having its own properties and illustrating how they interact with one another. Class diagrams use the rules established by the Unified Modeling Language to visually depict the static structure and composition of a specific system (UML).

**Fig 3.2 System Class Diagram**

**3.3.3 Activity Diagrams**

Similar to a flowchart or a data flow diagram, an activity diagram visually depicts a sequence of events or the flow of control in a system, but it functions more like an advanced version of both.

**Fig 3.3 Activity Diagram**

**3.4 Database Design**

The logical explanation of how data is kept in the computer's memory is called input specification. The freedom experienced in using the system, as well as the convenience of retrieving and reading the data and assuring applicability across the internet, make SQL standards essential for ensuring that structured data is uniform and independent of applications. Some of the input specifications employed in this project work are presented below.

1. Users Table: contains basic information about all system users.
2. Recipe Table: contains every system recipe information.

**Table 3.1 Users Table input specification table**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| Email | Varchar | 150 | Email for login (case sensitive) |
| Password | String | 150 | Access Code (case sensitive) |
| Firstname | String | 150 | User first name |
| Lastname | String | 150 | User last name |
| Phone | String | 150 | User phone number |
| Picture | Image | 100 | User profile picture |
| acct\_id | String | 64 | A unique string for identifying users |

**Primary key:** acct\_id

**Table 3.2 Recipe Table input specification table**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| Recipe\_id | String | 150 | A unique string for identifying tricycles |
| Recipe\_title | String | 150 | Recipe title |
| Recipe\_desc | String | 500 | Detailed description of the recipe |
| Reg\_date | Date | 20 | Date when the recipe was registered |
| Preparation\_time | String | 150 | Duration of recipe |
| Difficult\_rate | String | 20 | Difficulty level of the recipe |
| Recipe\_image | Image | 100 | Image to identify the recipe |

**Primary key:** recipe\_id

**3.5 Output Design**

This declares and displays the outcome of the given input. The automated system's output is dependent on its input. The output specification is listed below.

**Table 3.3 Users Table Output Design**

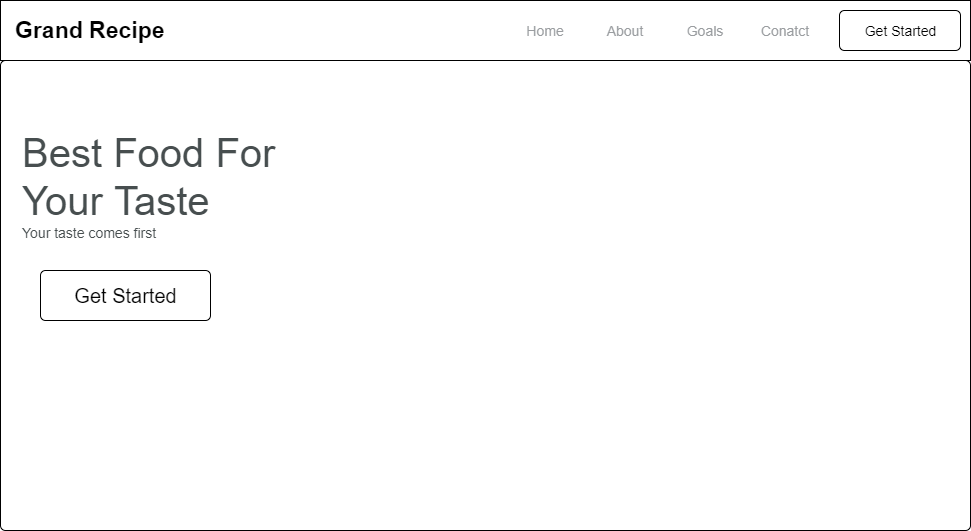
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Email** | **Password** | **Firstname** | **Lastname** | **Phone** | **Picture** | **Acct\_id** |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |

**Table 3.4 Recipe Table Output Design**

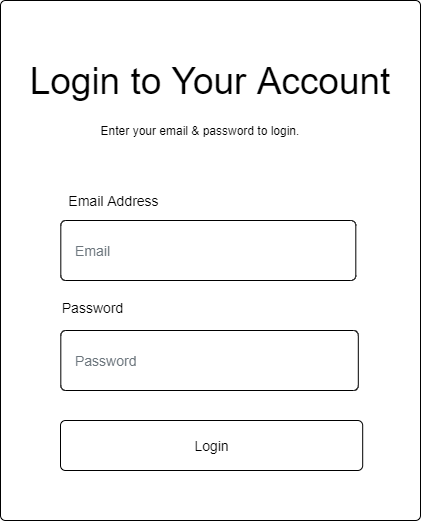
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Recipe\_id** | **Recipe\_title** | **Recipe\_desc** | **Reg\_date** | **Preparation\_time** | **Difficult\_rate** | **Recipe\_image** |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |

**3.6 Input & User Interface Design**

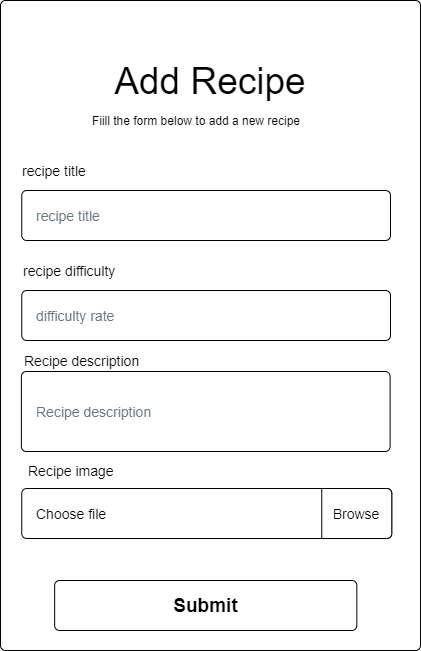
This displays the pictorial representation of the system interface, the interface is going to be designed in a way that it will be user friendly, responsive and attractive. It will also be well secured such that login will be required to access some level of contents. The designs are aided by a mid-fidelity wireframing tool called Draw.io



**Fig 3.4 Home Page Screen**



**Fig 3.5 Login Form**



**Fig 3.6 Add Recipe Form**

**3.7 System Requirement**

Every piece of software that is created has preset system requirements that it must meet in order to run at its best. However, the system requirements are the bare minimum hardware and software needed for the system's intended operation.

**3.7.1 Hardware Requirement**

System Hardware Requirement Include:

a. Minimum of 8 GB of RAM (Random Access Memory) installed.

b. Minimum of intel core i3 processor.

c. Minimum of 250GB HDD (Hard Disk Drive).

**3.7.1 Software Requirement**

The software requirements include:

a. At least windows 10 OS (Operating System).

b. Modern browsers such as Chrome, and Microsoft edge.

c. Vs. Code installation.

d. XAMMP installation.

**3.8 Choice of Programming Language**

The proposed design will be implemented using HTML, which is a markup language will be used for its user interface (frontend) while JavaScript will be used as the programming language for interacting with the database, MySQL will be used for its database due to its portability, the combination of the above modern technology forms the technology for this research work.

**CHAPTER FOUR**

**SYSTEM IMPLEMENTATION EVALUATION**

**4.1 Introduction**

This section provides a comprehensive explanation of the implementation process for the new system, highlighting its efficiency and effectiveness. It presents practical instances of the functional aspects of the system and outlines the steps involved in its implementation.

* 1. **System Testing and Evaluation**

Testing the developed system is crucial for several reasons. One key purpose is to uncover any potential flaws within the system and devise appropriate solutions. In this project, a combination of unit and integration testing was employed to verify the effectiveness and efficiency of the design, ensuring that the new system fulfills its functional requirements without any errors.

**Unit Testing**

This part examines specific units or single components of the system individually to confirm that specific phases function properly and without problems.

**Integration Testing**

Integration testing was performed on the software, wherein all components were brought together and operated as a unified system. The objective of this testing was to validate the connectivity and proper integration of the various parts, ensuring seamless collaboration among the units.

**4.3 System Installation**

In order to use the proposed application on any computer system, the following steps need to be taken:

1. Make sure, vscode, is installed on the system.
2. Copy your project folder to any location of your choice.
3. Open the project folder in Visual Studio Code
4. Run the application through the live server extension

**4.4 Security Measures**

The application has a public scope, allowing all users to access the available information. However, certain functionalities are restricted to the admin, the admin can add recipes, etc.

**4.6 Sample Outputs**

These describe and give the pictorial representation of the program or software; it shows and gives a clear understanding of the design, and displays all the interfaces.

**Homepage**

The image provided illustrates the homepage, which serves as the initial page and serves as a gateway to navigate and explore the various sections and functionalities of the website.

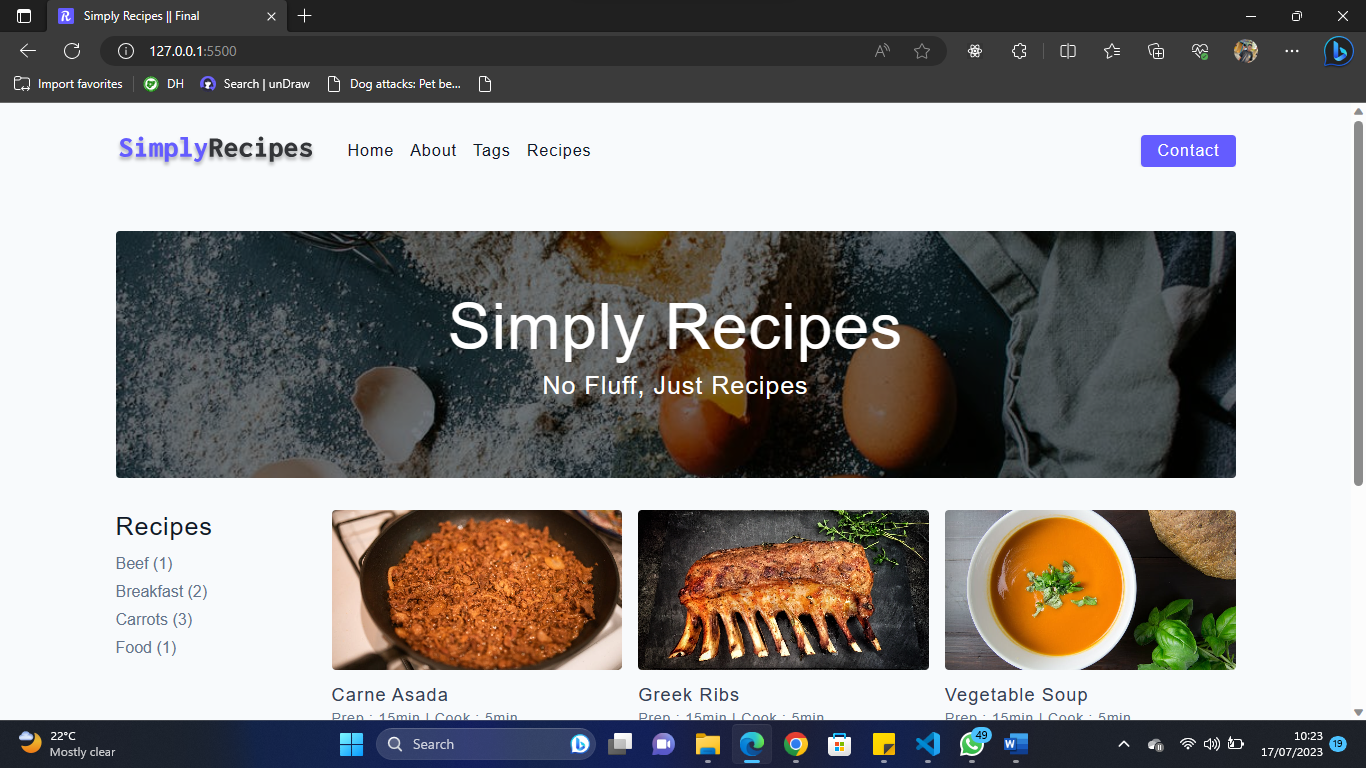


Fig 4.6.1 Homepage

**Tags**

The image provided illustrates all the tags of all recipes that are available on the system

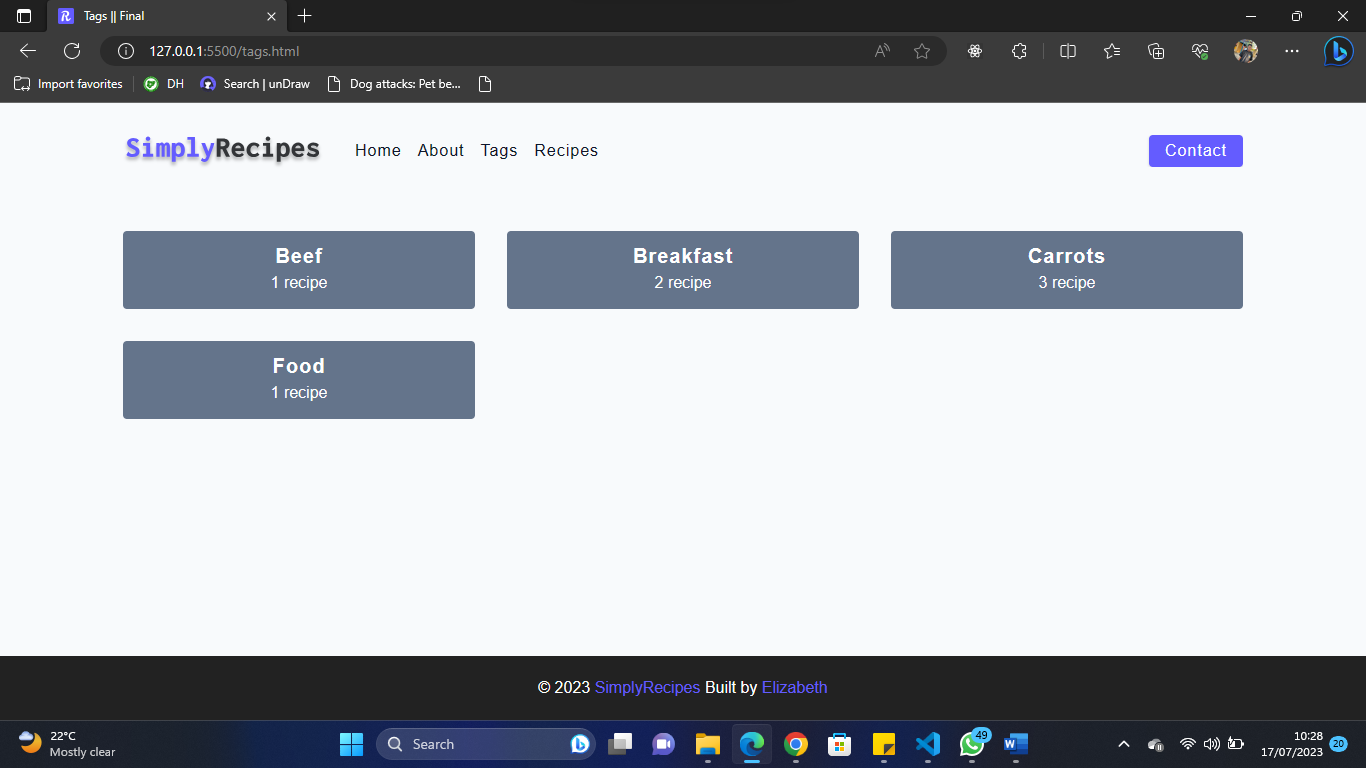


Fig 4.6.2 Tags

**Recipes**

The image provided illustrates all recipes and a preview of the duration of recipes

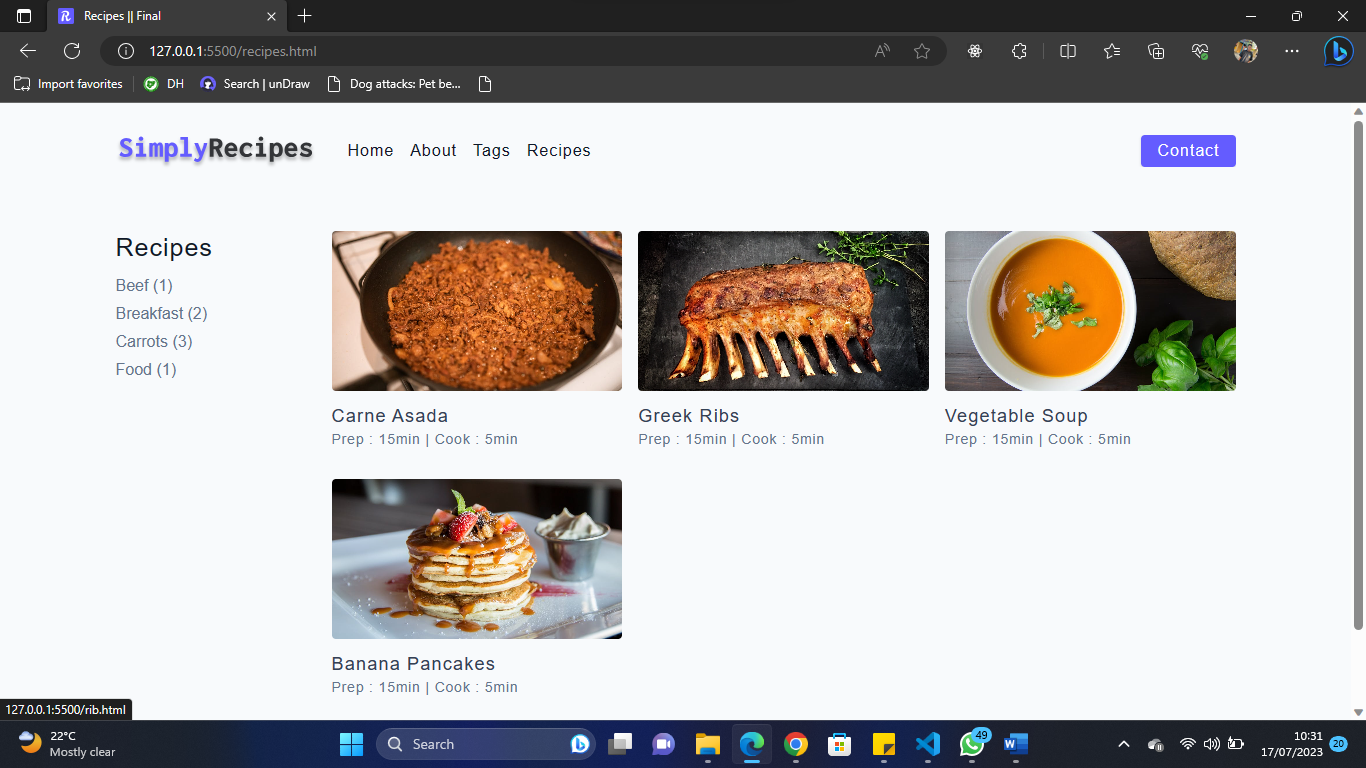


Fig 4.6.3 Recipes

**Recipes Details**

The image provided illustrates the detailed instructions of a recipe.

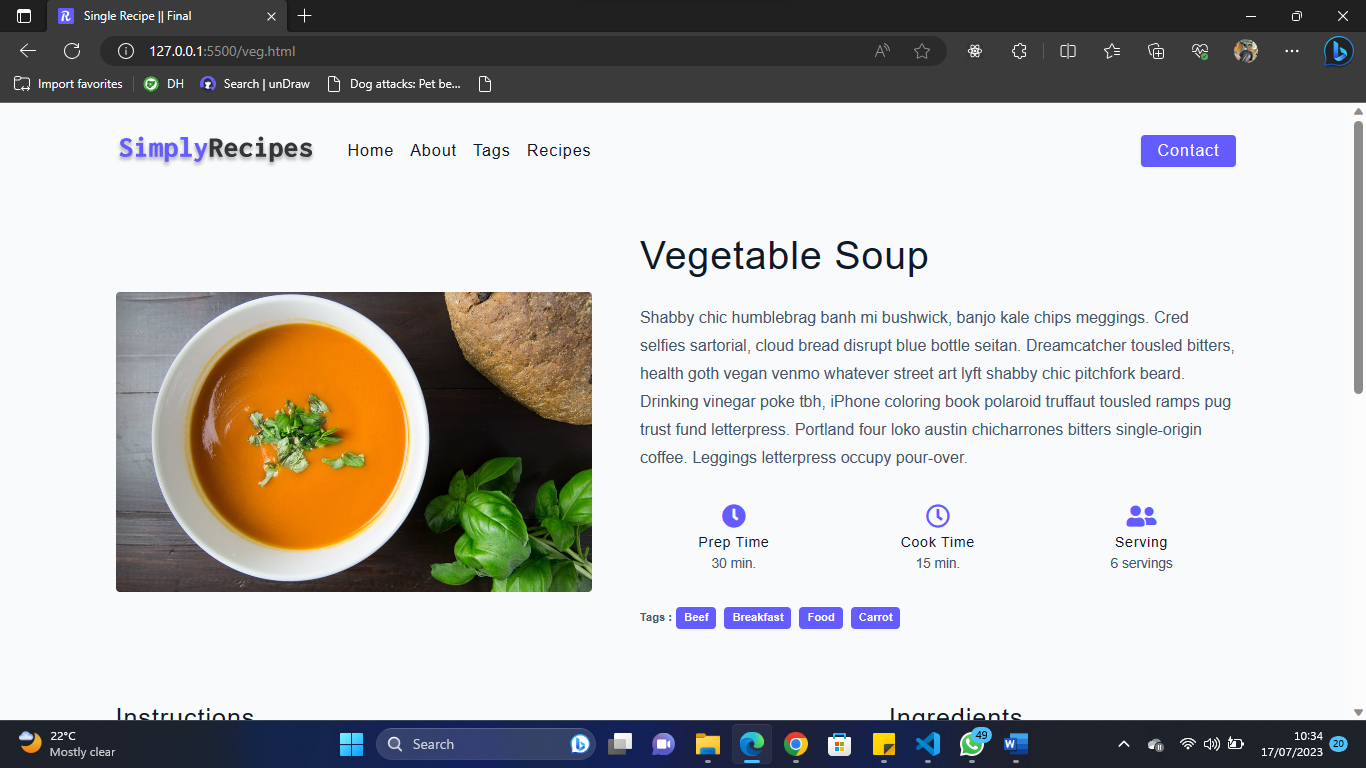


Fig 4.6.4 Recipes Details

**Breakfast**

This is the breakfast page where users can view the recipes of food.

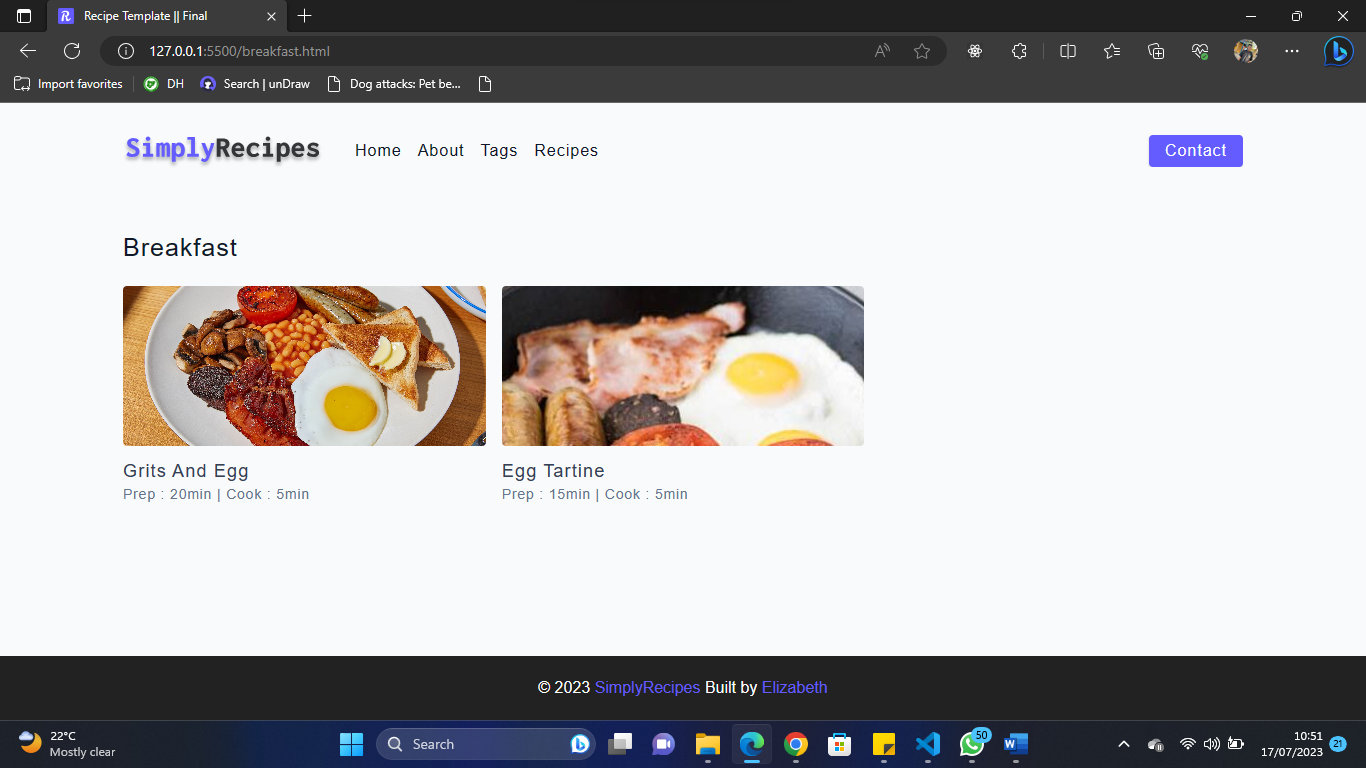


Fig 4.6.5 Breakfast

**About Us**

This is the about us page of the recipe

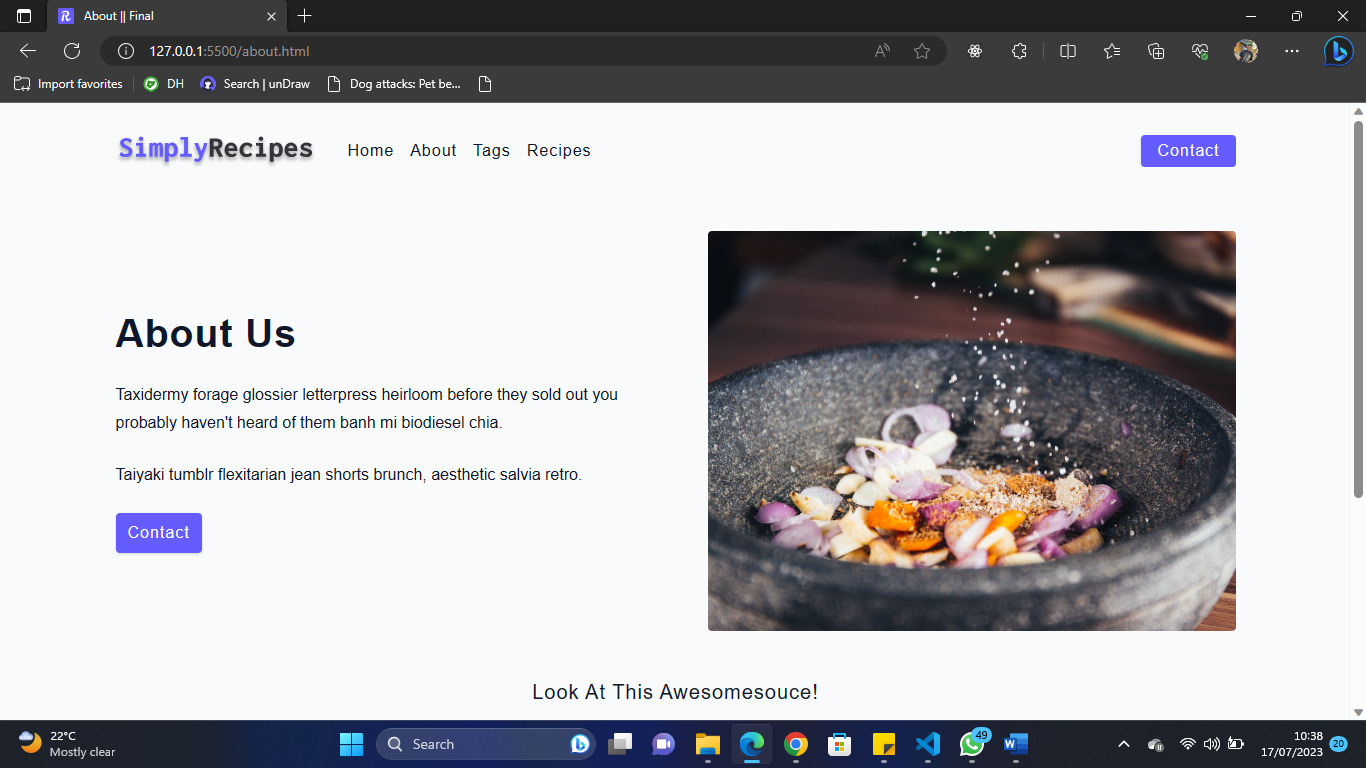


Fig 4.6.6 About Us

**Contact Us**

Users can use this page to get in touch with the admin

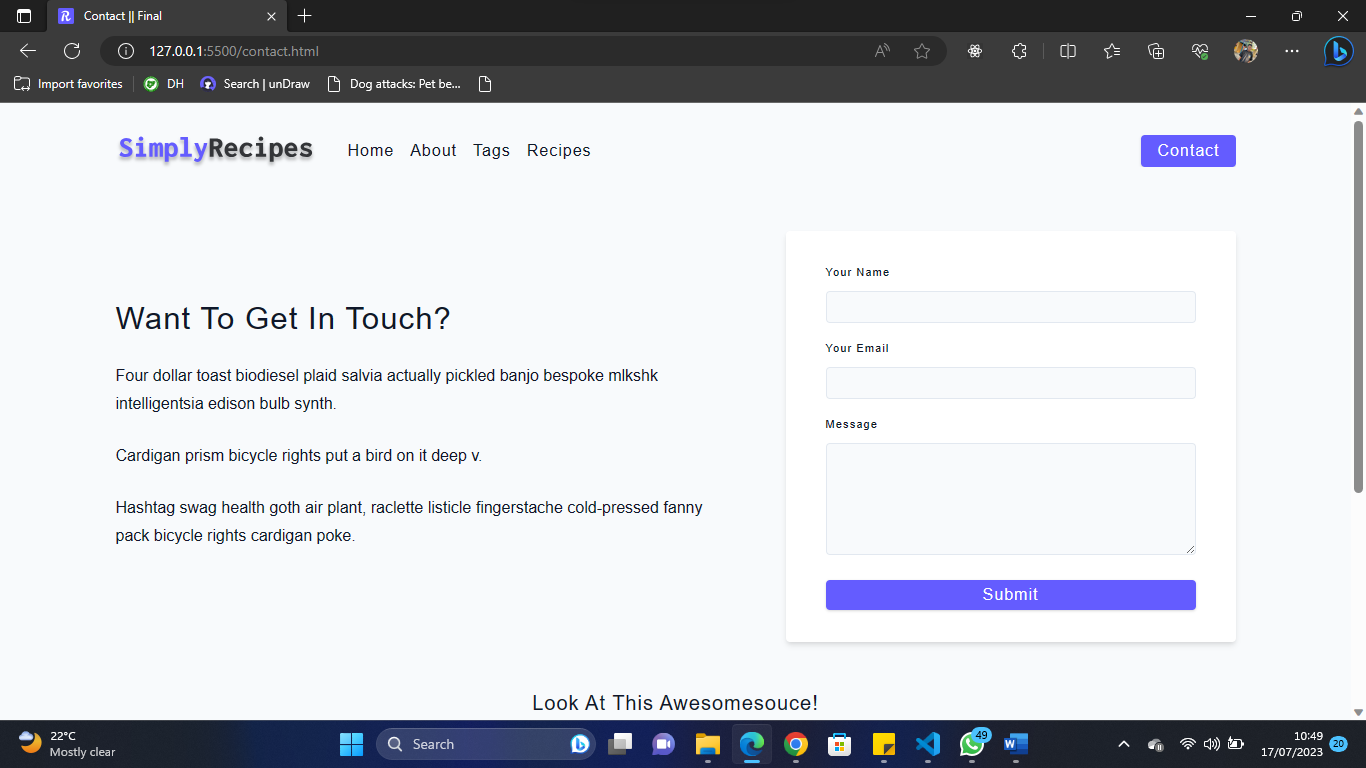


Fig 4.6.6 Contact Us

**CHAPTER FIVE**

**SUMMARY, CONCLUSION AND RECOMMENDATION**

**5.1 SUMMARY**

This project involves the development of a recipe website that caters to users' diverse culinary needs, including specific dietary requirements such as diabetes. The website aims to offer an engaging and user-friendly interface through HTML, ensuring a seamless user experience. JavaScript will be utilized for implementing the logical functionalities, while MySQL will be employed for data storage. The project objectives encompass designing an attractive user interface, incorporating necessary features, and conducting comprehensive evaluation and testing. The website will encompass a wide range of recipes from various regions, emphasizing options suitable for individuals with diabetes. However, it should be noted that the website will provide text-based recipes only, without accompanying video tutorials. Although the study is subject to limitations due to time constraints and the researcher's academic commitments, the website is expected to serve as a valuable resource for individuals and families seeking accessible recipes and meal planning guidance, particularly those with specific dietary needs.

**5.2 CONCLUSION**

In conclusion, the development of a recipe website that caters to diverse culinary needs, including specific dietary requirements like diabetes, is a valuable endeavor. The project focuses on creating an engaging and user-friendly interface using HTML, implementing logical functionalities through JavaScript, and utilizing MySQL for data storage. While the scope of the study covers a wide range of recipes from various regions, it is important to note that the website will provide text-based recipes without video tutorials. Despite the limitations imposed by time constraints and the researcher's academic commitments, the website is expected to serve as a convenient and accessible resource for individuals and families seeking recipe inspiration, meal planning assistance, and options tailored to their specific dietary needs. By offering a user-friendly experience and a comprehensive collection of recipes, the website aims to contribute to promoting healthy eating habits and empowering users to make informed food choices.

**5.3 RECOMMENDATION**

Based on the objectives and scope of the project, the following recommendations are suggested:

1. Incorporate Visual Elements: While the website focuses on text-based recipes, consider enhancing the user experience by including visual elements such as high-quality food images. Visual representations can enhance the appeal of recipes and help users better understand the final dish.
2. Expand Recipe Categories: To cater to a wider audience, consider expanding the range of recipe categories beyond specific dietary needs. Include options such as vegetarian, vegan, gluten-free, and lactose-free recipes to accommodate users with various dietary preferences.
3. Integrate Search and Filter Functionality: Implement a robust search and filter system that allows users to easily find recipes based on specific ingredients, cooking time, difficulty level, or dietary restrictions. This will improve user convenience and help them quickly discover recipes that suit their preferences.
4. Provide Nutritional Information: Considering the target audience, it is recommended to include nutritional information for each recipe, especially for those focused on dietary needs like diabetes. This information will assist users in making informed decisions about their meal planning and ensure adherence to their dietary requirements.
5. Regular Content Updates: To maintain user interest and engagement, ensure the website is regularly updated with new recipes, seasonal dishes, and cooking tips. This will encourage users to revisit the website frequently and explore fresh culinary ideas.

By implementing these recommendations, the recipe website can become a valuable resource for users seeking diverse recipes, tailored dietary options, and a user-friendly experience, ultimately promoting healthy eating habits and empowering individuals in their culinary journey.

**REFERENCES**

Astari, S. (2023, January 4). What is HTML? hypertext markup language basics explained.

Tutorials. Retrieved January 5, 2023, from <https://www.hostinger.com/tutorials/what->is-html

Bomfim, M. (2022). A Recipe for Food Literacy*: Designing and Evaluating Technologies for*

*Informed Food Choices (thesis). Association for Computing Machinery, New York*.

Colatruglio, Sarah and Joyce Slater (2016). Challenges to acquiring and utilizing food literacy:

Perceptions of young Canadian adults". *Canadian Food Studies, 3*(1), 96-118.

Cataldo, M., Alain, D., Christoph T., Amon, R.., Giovanni, S. (2021). Exploring the Effects of

Natural Language Justifications in Food Recommender Systems. *Association for Computing Machinery*. <https://doi.org/10.1145/3450613.3456827>

Goel, M., Chakraborty, P., Ponnaganti, V., Khan, M., Tatipamala, S., Saini, A., & Bagler, G.

(2022). Ratatouille: A tool for novel recipe generation. 2022 IEEE 38th International Conference on Data Engineering Workshops (ICDEW). https://doi.org/10.1109/icdew55742.2022.00022

Jordana, A. (2022, December 21). What is JavaScript? A basic introduction to JS for Beginners.

Hostinger Tutorials. Retrieved January 5, 2023, from https://www.hostinger.com/tutorials/what-is-javascript

Kosych, D. (2021, June 24). Cooking website templates - how to make Recipe & Cooking

website. MotoCMS Blog. Retrieved January 4, 2023, from https://www.motocms.com/blog/en/cooking-website-templates/

Pan, L. M., Chen, J., Wu, J., Liu, S., Ngo, C. W., Kan, M. Y., … Chua, T. S. (2020). Multi-

modal Cooking Workflow Construction for Food Recipes. In MM 2020 - *Proceedings of the 28th ACM International Conference on Multimedia* (pp. 1132–1141). Association for Computing Machinery, Inc. https://doi.org/10.1145/3394171.3413765

Roither, A., Kurz, M., & Sonnleitner, E. (2022). The Chef’s Choice: System for allergen and

style classification in recipes. Applied Sciences, 12(5), 25–90. https://doi.org/10.3390/app12052590

Yanai, Keiji & Maruyama, Takuma & Kawano, Yoshiyuki. (2014). A Cooking Recipe

Recommendation System with Visual Recognition of Food Ingredients. International Journal of Interactive Mobile Technologies (iJIM). 8. 28. 10.3991/ijim.v8i2.3623.

**APPENDIX**

**Homepage**

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

    <meta http-equiv="X-UA-Compatible" content="IE=edge" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <title>Simply Recipes || Final</title>

    <!-- favicon -->

    <link rel="shortcut icon" href="./assets/favicon.ico" type="image/x-icon" />

    <!-- normalize -->

    <link rel="stylesheet" href="./css/normalize.css" />

    <!-- font-awesome -->

    <link

      rel="stylesheet"

      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.14.0/css/all.min.css"

    />

    <!-- main css -->

    <link rel="stylesheet" href="./css/main.css" />

  </head>

  <body>

    <!-- nav  -->

    <nav class="navbar">

      <div class="nav-center">

        <div class="nav-header">

          <a href="index.html" class="nav-logo">

            <img src="./assets/logo.svg" alt="simply recipes" />

          </a>

          <button class="nav-btn btn">

            <i class="fas fa-align-justify"></i>

          </button>

        </div>

        <div class="nav-links">

          <a href="index.html" class="nav-link"> home </a>

          <a href="about.html" class="nav-link"> about </a>

          <a href="tags.html" class="nav-link"> tags </a>

          <a href="recipes.html" class="nav-link"> recipes </a>

          <div class="nav-link contact-link">

            <a href="contact.html" class="btn"> contact </a>

          </div>

        </div>

      </div>

    </nav>

    <!-- end of nav -->

    <!-- main -->

    <main class="page">

      <!-- header -->

      <header class="hero">

        <div class="hero-container">

          <div class="hero-text">

            <h1>simply recipes</h1>

            <h4>no fluff, just recipes</h4>

          </div>

        </div>

      </header>

      <!-- end of header -->

      <section class="recipes-container">

        <!-- tag container -->

        <div class="tags-container">

          <h4>recipes</h4>

          <div class="tags-list">

            <a href="beef.html">Beef (1)</a>

            <a href="breakfast.html">Breakfast (2)</a>

            <a href="carrot.html">Carrots (3)</a>

            <a href="food.html">Food (1)</a>

          </div>

        </div>

        <!-- end of tag container -->

        <!-- recipes list -->

        <div class="recipes-list">

          <!-- single recipe -->

          <a href="asada.html" class="recipe">

            <img

              src="./assets/recipes/recipe-1.jpeg"

              class="img recipe-img"

              alt=""

            />

            <h5>Carne Asada</h5>

            <p>Prep : 15min | Cook : 5min</p>

          </a>

          <!-- end of single recipe -->

          <!-- single recipe -->

          <a href="rib.html" class="recipe">

            <img

              src="./assets/recipes/recipe-2.jpeg"

              class="img recipe-img"

              alt=""

            />

            <h5>Greek Ribs</h5>

            <p>Prep : 15min | Cook : 5min</p>

          </a>

          <!-- end of single recipe -->

          <!-- single recipe -->

          <a href="veg.html" class="recipe">

            <img

              src="./assets/recipes/recipe-3.jpeg"

              class="img recipe-img"

              alt=""

            />

            <h5>Vegetable Soup</h5>

            <p>Prep : 15min | Cook : 5min</p>

          </a>

          <!-- end of single recipe -->

          <!-- single recipe -->

          <a href="banana.html" class="recipe">

            <img

              src="./assets/recipes/recipe-4.jpeg"

              class="img recipe-img"

              alt=""

            />

            <h5>Banana Pancakes</h5>

            <p>Prep : 15min | Cook : 5min</p>

          </a>

          <!-- end of single recipe -->

        </div>

        <!-- end of recipes list -->

      </section>

    </main>

    <!-- end of main -->

    <!-- footer -->

    <footer class="page-footer">

      <p>

        &copy; <span id="date"></span>

        <span class="footer-logo">SimplyRecipes</span> Built by

        <a href="https://www.student.com/">XXX</a>

      </p>

    </footer>

    <script src="./js/app.js"></script>

  </body>

</html>

**CSS**

\*,

::after,

::before {

  box-sizing: border-box;

}

/\* fonts \*/

@import url('https://fonts.googleapis.com/css2?family=Roboto:wght@400;500;600&family=Montserrat&display=swap');

html {

  font-size: 100%;

} /\*16px\*/

  /\* fonts  \*/

  --headingFont: 'Roboto', sans-serif;

  --bodyFont: 'Nunito', sans-serif;

  --smallText: 0.7em;

  /\* rest of the vars \*/

  --backgroundColor: var(--grey-50);

  --textColor: var(--grey-900);

  --borderRadius: 0.25rem;

  --letterSpacing: 1px;

  --transition: 0.3s ease-in-out all;

  --max-width: 1120px;

  --fixed-width: 600px;

  /\* box shadow\*/

  --shadow-1: 0 1px 3px 0 rgba(0, 0, 0, 0.1), 0 1px 2px 0 rgba(0, 0, 0, 0.06);

  --shadow-2: 0 4px 6px -1px rgba(0, 0, 0, 0.1),

    0 2px 4px -1px rgba(0, 0, 0, 0.06);

  --shadow-3: 0 10px 15px -3px rgba(0, 0, 0, 0.1),

    0 4px 6px -2px rgba(0, 0, 0, 0.05);

  --shadow-4: 0 20px 25px -5px rgba(0, 0, 0, 0.1),

    0 10px 10px -5px rgba(0, 0, 0, 0.04);

}

body {

  background: var(--backgroundColor);

  font-family: var(--bodyFont);

  font-weight: 400;

  line-height: 1.75;

  color: var(--textColor);

}

p {

  margin-top: 0;

  margin-bottom: 1.5rem;

  max-width: 40em;

}

h1,

h2,

h3,

h4,

h5 {

  margin: 0;

  margin-bottom: 1.38rem;

  font-family: var(--headingFont);

  font-weight: 400;

  line-height: 1.3;

  text-transform: capitalize;

  letter-spacing: var(--letterSpacing);

}

h1 {

  margin-top: 0;

  font-size: 3.052rem;

}

h2 {

  font-size: 2.441rem;

}

h3 {

  font-size: 1.953rem;

}

h4 {

  font-size: 1.563rem;

}

h5 {

  font-size: 1.25rem;

}

small,

.text-small {

  font-size: var(--smallText);

}

a {

  text-decoration: none;

}

ul {

  list-style-type: none;

  padding: 0;

}

.img {

  width: 100%;

  display: block;

  object-fit: cover;

}

/\* alerts \*/

.alert {

  padding: 0.375rem 0.75rem;

  margin-bottom: 1rem;

  border-color: transparent;

  border-radius: var(--borderRadius);

}

.alert-danger {

  color: var(--red-dark);

  background: var(--red-light);

}

.alert-success {

  color: var(--green-dark);

  background: var(--green-light);

}

/\* alert \*/

@keyframes spinner {

  to {

    transform: rotate(360deg);

  }

}

.loading {

  width: 6rem;

  height: 6rem;

  border: 5px solid var(--grey-400);

  border-radius: 50%;

  border-top-color: var(--primary-500);

  animation: spinner 0.6s linear infinite;

}

.loading {

  margin: 0 auto;

}

/\* title \*/

.title {

  text-align: center;

}

.title-underline {

  background: var(--primary-500);

  width: 7rem;

  height: 0.25rem;

  margin: 0 auto;

  margin-top: -1rem;

}

/\*

===============

Navbar

===============

\*/

.navbar {

  display: flex;

  align-items: center;

  justify-content: center;

}

.nav-center {

  width: 90vw;

  max-width: var(--max-width);

}

.nav-header {

  height: 6rem;

  display: flex;

  justify-content: space-between;

  align-items: center;

}

.nav-header img {

  width: 200px;

}

.nav-logo {

  display: flex;

  align-items: flex-end;

}

.nav-btn {

  padding: 0.15rem 0.75rem;

}

.nav-btn i {

  font-size: 1.25rem;

}

.nav-links {

  height: 0;

  overflow: hidden;

  display: flex;

  flex-direction: column;

  transition: var(--transition);

}

.show-links {

  height: 23.9375rem;

}

.nav-link {

  display: block;

  text-align: center;

  font-size: 1.25rem;

  text-transform: capitalize;

  color: var(--grey-900);

  letter-spacing: var(--letterSpacing);

  padding: 1rem 0;

  border-top: 1px solid var(--grey-500);

  transition: var(--transition);

}

.nav-link:hover {

  color: var(--primary-500);

}

.contact-link a {

  padding: 0.15rem 1rem;

}

@media screen and (min-width: 992px) {

  .navbar {

    height: 6rem;

  }

  .nav-center {

    display: flex;

    align-items: center;

  }

  .nav-header {

    padding: 0;

    margin-right: 2rem;

    height: auto;

  }

  .nav-btn {

    display: none;

  }

  .nav-links {

    height: auto !important;

    flex-direction: row;

    align-items: center;

    width: 100%;

  }

  .nav-link {

    padding: 0;

    border-top: none;

    margin-right: 1rem;

    font-size: 1rem;

  }

  .contact-link {

    margin-right: 0;

    margin-left: auto;

  }

}

/\*

===============

Page

===============

\*/

.page {

  width: 90vw;

  max-width: var(--max-width);

  margin: 0 auto;

}

.page {

  padding-top: 2rem;

  min-height: calc(100vh - (6rem + 4rem));

}

/\*

===============

Footer

===============

\*/

.page-footer {

  text-align: center;

  height: 4rem;

  display: flex;

  align-items: center;

  justify-content: center;

  background: var(--black);

  color: var(--white);

}

.page-footer h5 {

  margin-top: 0;

  margin-bottom: 0;

}

.page-footer p {

  margin-bottom: 0;

}

.page-footer .footer-logo,

.page-footer a {

  color: var(--primary-500);

}

/\*

/\*

===============

Recipes

===============

\*/

.recipes-container {

  display: grid;

  gap: 2rem 1rem;

}

.recipes-list {

  display: grid;

  gap: 2rem 1rem;

  padding-bottom: 3rem;

}

.tags-container {

  order: 1;

  display: flex;

  flex-direction: column;

  padding-bottom: 3rem;

  /\* background: blue; \*/

}

.tags-container h4 {

  margin-bottom: 0.5rem;

  font-weight: 500;

}

.tags-list {

  display: grid;

  grid-template-columns: 1fr 1fr 1fr;

}

.tags-list a {

  text-transform: capitalize;

  display: block;

  color: var(--grey-500);

  transition: var(--transition);

}