

<p><i>Assignment Cover Sheet</i></p> <p><i>School of Computing and Information Engineering</i></p> <p><i>Faculty of Computing and Engineering</i></p>	<p>Student Name: Steven Craig</p> <p>Registration No.: B00551021</p>
<p>Module Title: Project</p>	<p>Module Code: COM814</p>
<p>Supervisor: Dr. Adrian Moore</p>	<p>Year: 2015-2016</p>
<p>Hand-in date: 1st September 2016</p>	
<p>Plagiarism Statement:</p> <p><i>Plagiarism is the act of taking or copying someone else's work, including another student's, and presenting it as if it were one's own. Plagiarism is said to occur when ideas, texts, theories, data, created artistic artefacts or other material are presented without acknowledgement so that the person considering this work is given the impression that what they have before them is the student's own original work when it is not. Plagiarism also occurs where a student's own work is re-presented without being properly referenced. Plagiarism is a form of cheating and is a disciplinary offence.</i></p> <p>In submitting my work to Blackboard:</p> <p><i>I declare that this is my own work and that any material I have referred to has been accurately and consistently referenced. I have read the University's policy on plagiarism and understand the definition of plagiarism as given in the [course/subject] handbook. If it is shown that material has been plagiarised, or I have otherwise attempted to obtain an unfair advantage for myself or others, I understand that I may face sanctions in accordance with the policies and procedures of the University. A mark of zero may be awarded and the reason for that mark will be recorded on my file.</i></p>	
<p>I confirm that the program(s) and/or documents I have submitted for this assignment are my own work.</p> <p>Signature: S. Craig Date: 20/08/2016</p>	

TABLE OF CONTENTS

Title page and plagiarism statement	1
Table of Contents	2 - 4
Acknowledgements	5
Abstract	6
Introduction	7 - 10
1.1. Background	7
1.2. Problem statement	8
1.3. Aim	8
1.4. The Chosen Platform	8 - 9
1.5. The Development Environment	10
1.6. The Proposed Solution	10 - 11
Analysis	12 - 20
2.1. Market Research	12 - 14
2.2. Research Summary	14 - 15
2.3. Project Management	15 - 16
2.4. Architecture	16 - 17
2.5. Consideration of likely achievement	17
2.6. Professional Issues	17 - 18
2.7. Project Risks	19 - 20
2.8. Ethical Considerations	20
Requirements Analysis	21 - 23
3.1. System Requirements	21
3.2. Functional Requirements	21-22
3.3. Non-Functional Requirements	22 - 23

Design	24 - 47
4.1. User Interface	24 - 25
4.2. General Layout, Styles and Formatting	25 - 28
4.3. Architectural Design	28 - 29
4.4. Database Design	29 - 30
4.5. Website Structure and Content	30 - 47
Implementation	48 - 64
5.1. Development tools and training undertaken	48
5.2. Design prototype and objectives for implementation phase	48
5.3. Further development and iteration	49 - 64
Testing and Evaluation	65 - 70
6.1. Testing introduction	65
6.2. Test Cases	65 - 69
6.3. Evaluation	69 - 70
Conclusions and recommendations	71 - 75
7.1. Concluding thoughts	71 - 72
7.2. Recommendations for further development	73
7.3. Evaluation of development process	73 - 74
7.4. Reflections and lessons learned	74 - 75
References	76 - 81

Appendices	82 - 100
<u>Appendix 1 – Problem Rich Picture</u>	82
<u>Appendix 2 – Solution Rich Picture</u>	83 - 84
<u>Appendix 3 – Agile Sprints</u>	85
<u>Appendix 4 – Development Schedule</u>	86
<u>Appendix 5 – Storyboards</u>	87 - 92
<u>Appendix 6 – Architecture Diagram</u>	93
<u>Appendix 7 – Site Map</u>	94
<u>Appendix 8 – Database Design</u>	95 - 97
<u>Appendix 9 – Forms and Error Handling</u>	98 - 100

Acknowledgements

Thanks to Dr Moore, Dr He, and Mrs Allison for their very helpful feedback and support throughout the project. Thanks also to those family members who were considerate and understanding during some of the more stressful periods of development.

Abstract

The internet has enabled communities to form and connections to be made with others regardless of borders such as distance, background or time zone. As well as the large social networks like Facebook and Twitter which have a broad user base, there are also smaller networks which cater to more specific niches. From film fans to knitting hobbyists, the internet has allowed people to easily connect with those who share similar interests and passions. It has also made it easier for anyone to share their creativity, talents and ideas with a large audience. The popularity of services such as Netflix and Amazon which offer personalised and specific recommendations via algorithms has led to user's relationships to online content changing. It is now an expectation that websites and applications are able to adapt and respond to the needs of the user.

Chef's Table is a platform for those who want to share their love of food and cooking with others. As well as acting as a repository for user created recipes, the application caters to users that are looking for something specific, or are open to suggestions which take into account their preferences and previous interactions with the site. It also provides several features for engaging further with the content, such as recipe reviews, comments and public profiles.

By examining similar applications and carefully considering best practices in modern web development, a comprehensive solution has been developed which makes use of several development tools and languages to deliver a user-centric and responsive web application.

Keywords: *Web Application, User Content, Social, HTML, MySQL, PHP, Bootstrap*

1. Introduction

1.1. Background

Food and cooking as both a hobby and a creative activity has enjoyed a remarkable surge in popularity in recent years. Cookbooks have consistently proven to be 'one of the most reliable sectors of the book market' (Orr, 2012), with celebrity endorsed cookbooks particularly profitable.¹ TV shows such as MasterChef continue to draw 'more than 4 million viewers' (Plunkett, 2013) at their peak, despite a decline in overall television ratings (Zara, 2015). Online, there are a wide variety of food blogs, such as "Maison Cupcake" and "Deliciously Ella", as well as more traditional offerings from sources like the BBC and Food Network, which allow amateurs or experts to share their tips, tricks and food related experiences.

In addition to hobbyists and gourmands there are those who have an interest in cooking using quality ingredients in order to maintain a healthy varied diet, but perhaps face certain lifestyle obstacles to doing so. The hard-working professional in full time employment may not have an abundance of energy and time when they come home in the evening, and find that they often resort to unhealthy ready meals or expensive fast food. A parent who must plan out meals in advance for the whole family may struggle with balancing variety with the tastes and preferences of the other family members. A first year University student, living away from home for the first time might be living on a tight budget and feel intimidated by the prospect of preparing and cooking meals for themselves. Such individuals might feel intimidated by typical recipe websites which are often written by professional chefs and that sometimes assume a certain level of knowledge, proficiency, and convenient access to specialised and expensive ingredients.

The means by which people interact with web applications has moved away from the traditional desktop towards smartphones and tablets. At least '10 billion mobile internet devices' are expected to be in circulation by the end of 2016 (University of Alabama Birmingham, 2014). A study by Ofcom in 2015 found that smartphones were the 'preferred device' over a traditional desktop for interacting with social media platforms (Ofcom, 2015). It is important that modern websites are capable of responding to the device on which it is accessed, and can scale and streamline their content to enable a good user experience on smaller screens with touch-based interfaces.

¹ For example, Jamie Oliver's "30-Minute meals" became the 'fastest selling non-fiction title of all time' (BBC News, 2010)

1.2. The Problem

To provide an online recipe resource which caters to the requirements of a variety of users and can be accessed via desktops or mobile devices. The user should be able to find and discover recipes relevant to their tastes and preferences, publish their own creations, review those submitted by other users, store favourites and connect socially with those who share a similar love of food and cooking.

1.3. Aim

The aim of 'Chef's table' is to provide a web-based repository of user-created recipes which take account of a user's tastes and preferences and which can be accessed on a desktop, tablet or smartphone. The site will provide randomised and personalised recipe suggestions, as well as allowing a user to search for something specific via keywords, ingredients, upload date etc.

Chef's Table will also encourage a sense of community by allowing users to publish, share and review recipes and interact with fellow food lovers and cooks. Each user will also have the equivalent of a social media profile which stores their details and information and allows them to interact with other users and content.

The website should maintain a consistent user interface and adapt appropriately to different screen sizes, as well as adhering to best practices in terms of UI design, usability and accessibility.

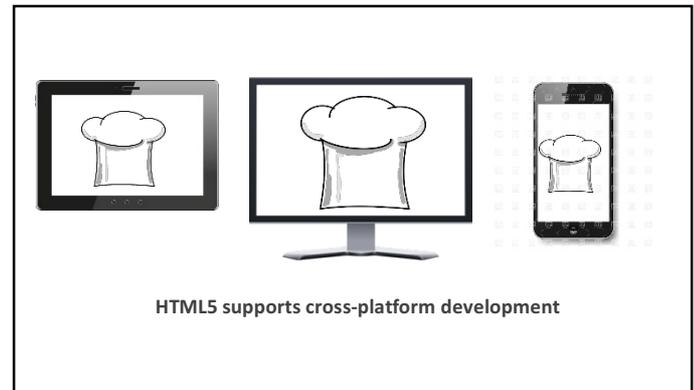
A rich picture diagram illustrating the problem can be found in [Appendix 1](#).

1.4. The Chosen Platform

Chef's table will be developed using a variety of commonly used web development languages and tools. HTML5 will be used to structure, format and display each page's content. CSS will be used to apply additional formatting and stylistic touches to content. Any interactive content will be implemented using JavaScript and JQuery. A MySQL database will be used to store all of the necessary user and recipe information. PHP will be used to access and manipulate this data, and will also be used to process any forms which capture input from a user. Finally, the Bootstrap mobile framework will be used to ensure portability and responsiveness on a range of devices.

When deciding on a development platform for Chef's Table, the following advantages of HTML were influential:

- Cross Platform Support – Chef's Table should be accessible on a variety of platforms, from Windows PCs to Android tablets and Apple iPhones. Unlike natively developed applications, HTML can work cross-platform without extensive reworking of code for each operating system.
- Multimedia support – HTML5 'provides native support for audio and video elements' (DPCI, n.d.) and eliminates the need for users to download and install 3rd party plugins such as Adobe Flash Player. This will also improve accessibility on smartphone and tablets which may not support such plugins.
- SQL integration and data storage – Chef's table will be storing a potentially large number of user accounts, recipes and user content. When used in combination with PHP, the HTML based website will be able to create and query records in a web-based SQL database. HTML5 also features an 'offline application cache' which allows users to 'reload a previously visited webpage fully even if they are temporarily offline', which would be useful should the user be accessing the site via a public WI-FI network (Tee, 2012).
- Interfaces easily with other programming languages – PHP scripts and JavaScripts have inbuilt support in HTML and can be easily implemented or imported where necessary.
- Easily readable code – HTML webpages follow a standardised template with the various elements clearly delineated by tags such as headers, paragraphs, footers etc. HTML5 also introduced new semantic layout elements for defining sections, articles and navigation bars. These tags will be useful for quickly prototyping page layouts and also for debugging and error checking purposes.
- Growing popularity of HTML5 – With 41% of app developers preferring pure HTML development to native or hybrid apps (Koetsier, 2013) and '30.6% of all Fortune 500 companies' (INCORE, 2013) using it to serve their business needs, HTML5 appears to be an attractive option for many sectors of the software industry. This popularity has also resulted in a variety of high quality tutorials and resources being made available online for support and teaching.



1.5. The Development Environment

There is a range of HTML editors and compilers available online which cater to different requirements. The code for Chef's Table will be written and edited using "Notepad++", a 'versatile and powerful source code editor' with many benefits, including its open-source nature, uncluttered layout and 'Syntax Highlighting', which clearly distinguishes blocks of code and potential errors (Mah, 2010).

During development, Chef's Table will be hosted locally using "Xampp", which is a free and open source Apache distribution which includes a local web server and MySQL database component. Developing on a local web server provides several benefits including low latency and quick response times (which aids testing and rapid iteration) and portability.

1.6. The Proposed Solution

Chef's Table will be a web-based application which will cater to users with a broad range of requirements and expectations. It will act as an online repository for user-submitted recipes which can be searched for using an array of tags such as ingredients, user rating, difficulty and preparation time. The site will allow users to interact with the site's content and others by creating, sharing and reviewing recipes and using other social features such as posting to their profiles or leaving comments.

The overall aim of the site is to promote the discovery of new and varied recipes, as well as providing a platform for people to share their love of good food. The user experience will be a consistent priority during design and implementation. In order to understand the expectations and desired features of each user group targeted by the site, similar websites will be researched and evaluated in terms of design, functionality and innovation.

Testing will be an important element at each stage of the development cycle. In order to achieve a high level of quality, the site will be tested extensively for bugs and device compatibility issues throughout development. When a functioning prototype has been developed, feedback will be carefully considered and analysed to identify any overlooked functional/design flaws.

The goal of the project is to deliver a fully functional application which is visually attractive to the user and encourages prolonged engagement and interaction. It should clearly evidence a familiarity and understanding of HTML, PHP, CSS, JavaScript, SQL and Bootstrap. Development will be thoroughly documented by way of various reports and log entries.

A rich picture diagram which illustrates the proposed solution to the problem can be found at [Appendix 2](#).

2. Analysis

2.1. Market Research

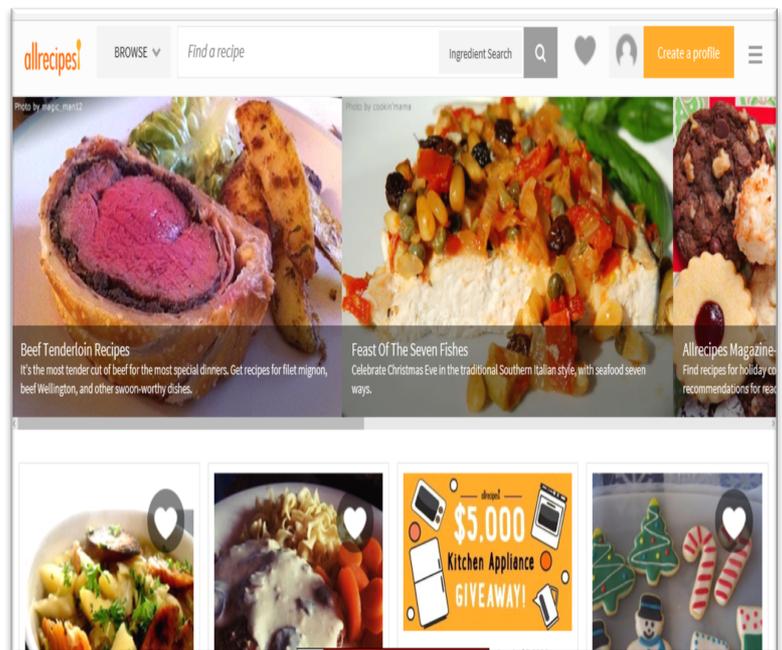
To gain an understanding of the minimum requirements for Chef's Table, best practices, and identify potential areas for innovation, similar web sites and applications will be critically analysed and a summary of findings will be presented. According to user statistics gathered by the web analytics firm 'Alexa' the following sites are amongst the most popular recipe and cooking destinations online:

1. *Allrecipes.com*
2. *Cooks.com*
3. *BBC Food*

Allrecipes.com

Strengths:

- + Sorting of recipes according to popularity and recently added etc. Similar sorting methods are planned for Chef's Table.
- + A shop feature which allows users to purchase various kitchen appliances directly. This is a good example of expanding beyond the usual expectations of a recipe site, however may be beyond the scope and resources of this project.
- + Support for user submitted recipes and ratings. 'My Recipes Box' feature allows users to save favourite recipes for convenience. A means of saving favourite recipes to a user's account for ease of access will also be a feature of Chef's Table.
- + Attractive layout and user interface.
- + Rich media features such as photo galleries and videos.



Screenshot from Allrecipes.com

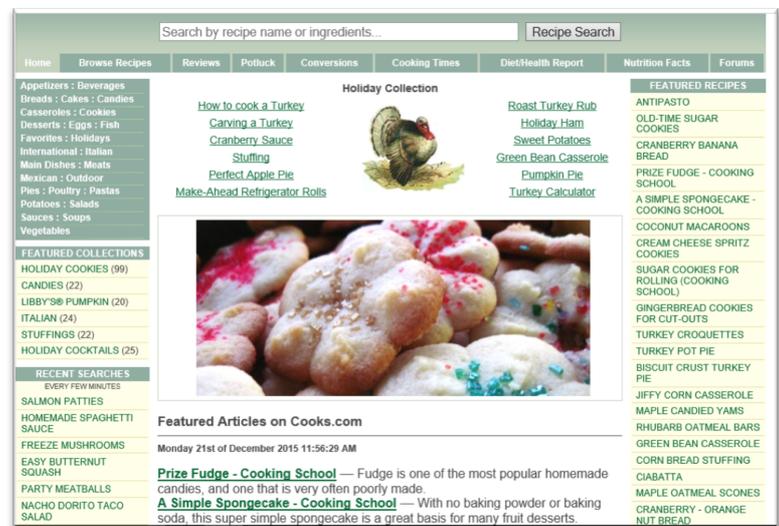
Weaknesses:

- Certain pages such as the home page and recipes browsing section contain a large number of images which are loaded at once on opening the page. This may cause sluggish performance on slower machines or internet connections.
- Some features require a subscription fee such as searching by nutritional content and a timer for use while cooking. This has the effect of potentially segmenting the user base.

Cooks.com

Strengths:

- + Can search by both recipe name and ingredients. An ability for users to search for recipes based on the ingredients which they have to hand will be a search feature when using Chef's Table.
- + Features articles written by professional contributors on how to prepare certain dishes. An alternative version of this could be implemented on Chef's Table by allowing users to post their own guides to certain kitchen tasks or preparing particular recipes/ingredients.
- + Has a forum feature for users to post questions and discuss tips and recipes. Elements of this can be implemented on Chef's Table via user reviews and user comments.
- + A nutrition section which includes information about the calorie and nutrient content of commonly used ingredients. As Chef's Table is focused on user created recipes, such a feature would add an unnecessary burden to users when they are publishing their recipe to the site.



Screenshot from Cooks.com

Weaknesses:

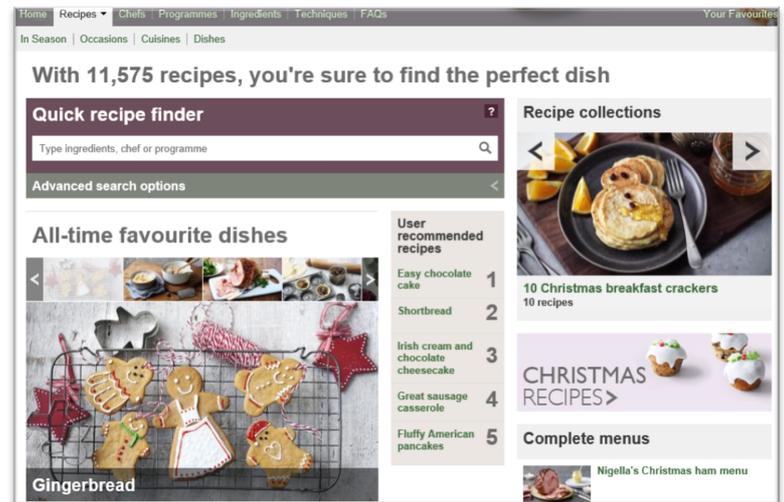
- Main page has quite a cluttered layout and new users who are navigating the site for the first time may feel disorientated. Care will be taken not to overwhelm users of Chef's Table by ensuring that content is clearly presented and divided evenly between pages.

- Certain recipes can be lacking in important detail such as skill required or preparation time. Chef's Table will allow users to describe the estimated difficulty and time to prepare for their recipes.

BBC Food

Strengths:

- + A quick recipe search according to ingredients, chef and the BBC programme on which it appeared, as well as an advanced search option.
- + Can browse recipes according to dietary preferences and allergies such as dairy-free and pregnancy friendly. Categories related to specialist diets should be included when a user is searching for or creating a recipe.
- + Can store recipes in a "favourites" collection.
- + Large collection of around 11,575 recipes in total. The database for Chef's Table should be capable of storing a large volume of data without suffering from inefficiencies in searching and sorting tables.



Screenshot from BBC food

Weaknesses:

- The site is closely associated with the BBC Food television programming and as such most recipes are created by professional chefs as opposed to non-professionals. This may lead to a slightly higher learning curve for following certain recipes.
- Limited community features.

2.2. Research Summary

This evaluation has helped to identify a number of essential criteria and lessons in quality web design. The website should contain a variety of categories which can be searched using

tailored search terms. The layout of each page should be uncluttered and easily navigated. It would be useful to have a navigation bar along the top of the page with a consistent layout on each webpage in order to lend structure to the site and provide quick access to the main sections. There are a number of very high quality recipe websites available at the moment, with large quantities of recipes and registered users. The emphasis of Chef's Table will be on creating a community of amateur and semi-professional chefs and to provide personalised recipes suggestions by tracking user data and trends.

2.3. Project Management

In order to ensure the solution can be delivered within the allotted time frame, it is important to have an understanding of how to successfully manage a long-term, work-intensive project. The two most popular methods have been considered and will be summarised.

Considered: Waterfall Method

The waterfall method involves working through each phase of the project in sequence. So, for example, the developer would do 'all the requirements gathering and design work' (Bowes, 2014) before code is actually written, then write the code, then perform testing etc. The benefit of the waterfall method is its focus on extensive planning at the outset of the project, which can lead to improved quality due to the clearly defined vision for the finished product and by dealing with any potential issues at the design phase. The rigid structure of the waterfall method can also be a disadvantage however, as it means that '[a]ltering the project design at any stage' can cause serious issues and feedback and testing does not occur until the late stages of the process which can be 'very difficult to respond to' (Mikoluk, 2013).

Chosen: Agile method

Agile development is an increasingly popular project management tool which focuses on 'incremental, iterative work cadences, known as sprints' (Agile, 2008). It differs from traditional sequential development in that features and components are added in increments and rapidly iterated on until they represent 'a potentially shippable product increment' (Agile, 2008). The flexibility of Agile is its major advantage, as it enables frequent self-reflection and reassessment of each step of development, from analysis to testing. The focus on shorter iterations can improve on quality as it encourages the developer to 'only build what is necessary to satisfy the requirements for that iteration' and the component will be the sole focus of attention for that period of development (Haunts, 2014). Quality is also improved by

the continuous testing throughout the project, rather than testing being relegated to the final stages. Disadvantages include the potential for “scope creep”, which is when the goals of a project ‘expands while it’s in progress’ and could lead to longer development times or a drastic divergence from the solution originally envisioned (Techopedia, n.d). Agile development can also be mentally tiring for the developer due to the ‘relentlessness of iterations’, and so maintaining a ‘sustainable pace’ is essential (Haunts, 2014).

An outline of the sprints which will be undertaken can be found in [appendix 3](#).

2.4. Architecture

Chef’s table will be built using a combination of the following programming languages:

- HTML will be used to enable web browsers to read and display the site, structure content on the pages, create elements such as containers, forms and tables as well as interface with other programming languages.
- CSS will be used to style the individual HTML components and improve the visual appeal of the site. An external stylesheet containing details of common styling across all webpages (e.g. for images, headers, footers etc.) will help to provide a consistent style and avoid the need for updating each page individually. Each page’s styling will also be stored in a separate style sheet to improve readability of the code.
- JavaScript and JQuery will be used for the interactive and dynamic elements of the site. Examples include hiding and displaying elements according to the needs of the user, providing previews of images and adding functionality to buttons.
- SQL will be used to communicate with the database for the website. This database will store user accounts, recipes, shopping lists and other required information. SQL is a simple and efficient means of executing queries against this database. Examples include adding a new user or recipe and responding accurately to a search query submitted by a user.
- PHP scripts are executed on the server side and will provide extra functionality when accessing the database. PHP will also be used to validate the various forms on the site and store session/cookie data.

- The Bootstrap mobile framework will be used to ensure that the application is responsive to different kinds of devices and screen sizes, as well as additional styling and visual effects.

During development and prototyping the online functionality of the site will be simulated on an open source Apache web server. In order for Chef's Table to be available online, a web host and domain provider will also be required. After researching several recommended providers, the service provided by "GoDaddy" was selected as the most suitable option. The reasons for choosing this provider include the following:

- Each plan includes a MySQL database with sufficient capacity for the storage requirements of Chef's Table. It also provides a user-friendly graphical user interface for working with the database via the "phpMyAdmin" interface.
- A free domain name included with each hosting plan. The domain name for the site will be "chefs-table-uuproject.net".
- A guarantee of 99.9% uptime for all sites.
- The cost of the most suitable plan is reasonable and falls within the financial budget for the project.

2.5. Consideration of likely achievement

The project time frame for the project is approximately 9 months, from December 2015 – September 2016. A proposed time table for development can be found in [Appendix 4](#). By clearly identifying essential and desired features, it is expected that a fully functional prototype will be in place by June, and the full system delivered after a period of further refinement and rigorous testing in August/September.

2.6. Professional Issues

Business Case

The purpose of a business case is to consider 'all feasible approaches to a given problem', in order to identify the most suitable solution (Wigmore, 2014). Chef's table is intended to be a resource for anyone who enjoys food and cooking, and wants to engage with others who share this interest. Current trends suggest that social media will continue to be a prominent feature in many people's lives, and integrating social functionality will be a valuable means of

promoting a sense of community amongst the sites user base and maintaining prolonged interest and engagement. Smartphones and tablets, as multi-function general purpose devices, are a constant presence for many individuals, including in the kitchen or when grocery shopping. As such, Chef's Table should be easily portable across devices.

Technical Benefits

- The Website will be built using HTML, a development environment which promotes thorough prototyping, experimentation and debugging. The cross-platform nature of HTML5 in conjunction with Bootstrap will reduce the time taken to customise the site for multiple operating systems and screen sizes.
- Cost effectiveness. The application will be built mainly using free open source software such as Notepad++. Online testing will be performed on an open-source Apache webserver.
- Development can be easily broken down into manageable units of work, with functionality added gradually, which lends itself ideally to the AGILE design method. Features can be divided between essential and desirable and then prioritised accordingly.
- Potential for further expansion and development provided the project is delivered successfully within the timeframe.
- As the target user base is quite wide, it should be relatively straightforward to identify the needs and expectations of typical users.

Costs

- The costs for this project will be mainly in time and effort. The number of hours required to complete the project is estimated at 600 hours, and must be balanced with various other academic and personal commitments. As such effective project management is a necessity in order to ensure that the final product is delivered within the time frame and to a satisfactory level of quality.
- Financial costs will consist of subscription costs for the chosen domain service provider. This will be discussed in more detail in the architecture section of the report.
- As the website will be accessed using a web browser, the costs associated with the approval process for making an application available on a mobile platform such as the Apple store or Android marketplace can be avoided.

2.7. Project Risks

At the outset of a software project of any scale, it is important that an effective risk management strategy is identified and implemented. The purpose of risk management is to 'identify, address, and eliminate software risk items' before they become problematic enough to affect the quality of the product or will significantly increase the time taken to deliver the finished solution (Boehm, 1989). The key element of a successful risk management plan is to take a proactive approach by clearly identifying the potential risks which could arise, the probable impact on development, and the strategies for minimising the potential damage should problems occur. The risks that might be encountered during development of Chef's Table can be categorised under three headings:

1. Technical risks refer to 'project-level concerns' (Software Engineering Institute Carnegie Mellon, 2012), such as the prior knowledge and familiarity with the tools used and having a firm grasp of the overall direction of the project.
2. Physical risks include tangible damage to hardware, software or storage methods used throughout the project.
3. Project management risks refer to aspects such as maintaining schedule commitments and realistic planning and prioritisation during the project.

The following table contains a summary of relevant risks considered at the outset of the project. Each risk is detailed and assigned a probability and impact value, with one being the least likely/least damaging and five being the most. The risk is also categorised according to whether it is physical, technical or a project management consideration. Finally, a suitable probability and impact reduction strategy is identified on the basis of the likelihood of the event occurring and its potential impact on the success of the project.

Risk	Probability	Impact	Type	Probability Reduction	Impact Reduction
Hardware failure resulting in loss of data. An example would be if a hard drive or USB pen became corrupted and several days of work was lost.	3	2-5	Physical	Ensuring data and builds are constantly saved and backed up on different media.	Maintain different versions of project on multiple storage devices which can be easily accessed in case of hardware failure.
Server failure resulting in loss of database or associated data	1	5	Physical	Selecting a reputable and high quality service provider to host both the	Keep regular backups of essential data.

				website and the SQL database	
Updates to development platforms.	2	3	Technical	Keep abreast of updates to relevant development tools. Browse forums and release notes in order to identify particular issues and solutions.	Perform thorough testing throughout development and in particular after software updates.
Loss of development time due to illness or other personal reasons.	2	3	Project Management	Ensure that project receives a consistent level of attention and effort for the entirety of the development time frame.	Should time be lost to illness or other reasons, ensure that a project plan is in place with key milestones highlighted, and that focus is placed on meeting the essential requirements of these milestones.
Unfamiliarity with required programming techniques or tools.	5	2	Technical	Identify at the outset all of the tools which will be used during development and recognise which of these will require familiarisation and learning. Follow relevant tutorials closely, paying particular attention to features that can be implemented into project.	Ensure that time is reserved within the project plan for learning and improving upon current programming skills and knowledge.
Overly ambitious project scale or feature set.	2	2	Project Management	Identify a realistic sense of the essential and desired features of the finished project through careful analysis and user feedback.	Should development take longer than anticipated, focus development on the essential features
Feedback from testers on prototype identifies serious flaws or oversights	3	3	Project Management	Test website extensively throughout development of prototype and the finished product.	Carefully consider user feedback and allocate appropriate time and attention to any bug fixing required.

2.8. Ethical Considerations

The project is not invasive towards users and any data held about test users in the database will adhere to data protection principles. Any testing will be conducted using adults and will not involve gathering personal information on minors under 18 years of age.

3. Requirements Analysis

3.1. System Requirements

The system requirements for Chef's Table will be divided into two categories – functional and non-functional. Functional requirements can be defined as 'what the system should do or provide for users' (SQA, 2007). These include specific functions such as the ability to create and upload a recipe or search according to a particular search term. The functional requirements are outlined in the table below. Non-functional requirements consist of attributes external to the system, and include considerations such as 'service availability, responsiveness, throughput and reliability' (SQA, 2007). Overall site stability and support for different devices and screen sizes could fall into these categories. The non-functional requirements are also detailed below.

Each requirement will feature user stories. User stories consist of 'a few sentences in simple language that outline the desired outcome' which are then used to define the steps required to implement the proposed system (Radigan, n.d.). They are also important as they allow the developer to visualise how a user will interact with the system in a realistic setting rather than in an abstract sense.

3.2. Functional Requirements

User Story	Requirement	Description
I often have left over ingredients, which end up being wasted. I would like to find recipes which contain specific ingredients	The user should be able to search for recipes which contain a specific ingredient.	On the search page, the user can specify an ingredient. Any recipes that contain that ingredient will then be presented to the user.
I want to find recipes which contain this keyword / uploaded yesterday / have a minimum 4-star rating.	The user should be able to add further search criteria to customise their search	On the search page the user can search using a variety of filters.
As an inexperienced cook, I find many recipes to be quite intimidating. I would like recipes to be simple, clear and easy to follow.	Submitted recipes should not be more complex than necessary and indicate clearly whether they are intended for beginners or more experienced cooks.	User-submitted recipes should follow a consistent template and the form for adding a recipe should closely mirror the layout of the recipe viewer.
I would like to share my best recipes with other users and develop new ones which can be revised according to feedback.	User must be able to create and upload recipes. Other users should be able to leave reviews and comments after trying a recipe.	The user should be able to easily add their recipes to the site using a clearly laid out form which is not difficult to complete. The site should encourage users to engage with the wider community

		through user-friendly create, share and review options.
As a parent who has to make weekly meal plans, I often resort to repetitive, reliable recipes. I would like to have a variety of suitable recipes available in order to break out of this cycle.	User should be able to browse according to specific categories. The site should offer personalised and randomised recommendations.	All recipes should contain tags which allow them to be sorted according to categories such as child-friendly, gluten free, type of cuisine etc. The site should offer suggestions which cater both to those looking for something different and looking for something specific.
Cooking is a hobby of mine, but one thing I dislike is the preparation required before I can even get started.	The site should provide tools and resources for making preparation more convenient.	A shopping list feature which allows users to easily add one or many ingredients from a recipe to a list would be useful.
Food and cooking is a passion which none of my family or friends share. I would like to meet others with who I can converse and share ideas.	The site should promote a sense of community.	Each user should have an equivalent of a social media profile, which they can use for expressing their personality, culinary adventures and favourite recipes. Users should be able to communicate with other users, and add users they like to their followed list. A personalised feed will then list any updates from those the user is following.

3.3. Non-Functional Requirements

Requirement	Description
Accessibility	The site should be accessible to hearing-impaired and colour blind-users and be screen-reader friendly.
Documentation	Development should be sufficiently documented throughout. The site should have instructions where necessary and a user manual will be created which will explain how to use the site.
Stability	The site should be accessible on most devices and run smoothly with a minimum of delay or errors.
Quality	The site should be visually appealing and free of major bugs and programming bugs/inefficiencies.
Robustness	The site should be easily modifiable at all stages of development and new features capable of being added without a significant detriment to existing ones.

Privacy	Any information stored for the purpose of maintaining user accounts or obtained via feedback should be held securely and in accordance with relevant data protection Principles.
Testing	Site should be consistently and thoroughly tested throughout development and in light of any potential software updates. Major bugs should be identified and resolved promptly.
Compatibility	A consistent user interface should be maintained regardless of the device which the site is accessed on.

4. Design

4.1. User Interface

HCI (or Human Computer Interaction) refers to the way in which a person interacts with a computer or piece of software. It is particularly important when the aim is to design ‘clear intuitive systems’ which can be used by ‘people with a varied range of abilities and expertise’ (Fitzpatrick, n.d.).

When researching HCI in the context of web design, two key principles for successful design have been identified:

1. **Accessibility and Inclusion** – This means that people with a disability or other impairment should not have a lesser experience when attempting to use the site. People with disabilities should be able to ‘perceive, understand, navigate and interact’ with the content, and to ‘contribute equally without barriers’ (W3C, 2010). Somewhat similar to accessibility, inclusion refers to issues such as the equipment of the user, internet access, ‘economic situation; education; geographic location; and language’ (W3C, 2010).

To promote accessibility, Chef’s Table will be designed to be screen-reader friendly i.e. accessible for sight-impaired users. This involves using clear and descriptive names for links, avoiding ‘[c]rowded, overly busy pages’, structuring code using headings, paragraphs and other tags, as well as including ‘alt text for all graphics’ (Amherst College, n.d.). To ensure colour-blind users are comfortable using the site, colours will not be used as the primary ‘method of conveying important information’, as informative text or other prompts will also be present (WebAIM, 2013).

```

</div>
<div>
<div class="row">
<div class="col sm 6">
<div class="view_category php/types/breakfast_breads">
<div class="img">
<img alt="Images/Breakfast and Breads.jpg" alt="Breakfast Pic" class="img-responsive img-rounded lazy"/>
<div class="caption"><h2>Breakfast and Breads</h2></div>
</div>
</div>
<div class="col sm 6">
<div class="view_category php/types/meal">
<div class="img">
<img alt="Images/Smooth and Salads.jpg" alt="Smooth Pic" class="img-responsive img-rounded lazy"/>
<div class="caption"><h2>Smooth and Salads</h2></div>
</div>
</div>
<div class="col sm 6">
<div class="view_category php/types/meal/meal_main">
<div class="img">
<img alt="Images/Appetizers and Snacks.jpg" alt="Appetizer Pic" class="img-responsive img-rounded lazy"/>
<div class="caption"><h2>Appetizers and Snacks</h2></div>
</div>
</div>
</div>
</div>

```

Code will be clearly and neatly structured

```

i_star.png' class='img-responsive img-thumbnail starImgs' alt='Gold Star'/>"

```

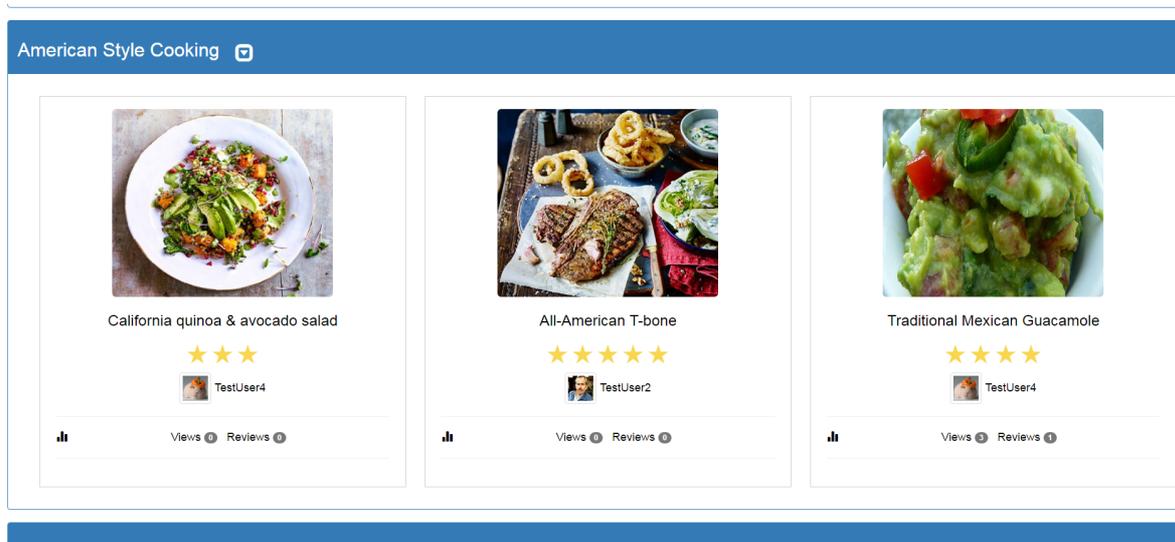
All graphics will include alt text for screen readers

Are you sure you want to delete this recipe?

Where colour coding is used, clear text prompts must also be included for colour blind users

2. Usability – It should be easy for a user to ‘achieve specified goals effectively, efficiently and with satisfaction’ (W3C, 2010). Methods for achieving good usability include presenting information ‘in a clear and concise way’, minimising ‘ambiguity regarding the consequences of an action’ and ensuring that important content is prominently situated on the page (European Commission, 2016).

Steps which will be taken to promote usability include simplifying navigation by organising ‘information into categories’, using appropriate spacing and avoiding ‘unnecessary information, complicated vocabulary, and long sentences’ (Barker, 2015).



Information should be logically organised and appropriately spaced.

As Chef's Table will be accessible on smart phones and tablets, additional challenges specific to mobile HCI design must be considered also. These include the touch based interface, vastly smaller screen size and less powerful processors.

4.2. General Layout, Styles and Formatting

Each page will follow a relatively consistent layout and will be structured using the Bootstrap responsive grid system. This involves wrapping content in rows and columns. 12 columns can be used across the horizontal space of a page, and columns can re-arrange based on the screen size. Column space can be divided equally or unequally between content elements, but the column span must always be 12.

span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1
span 4				span 4				span 4			
span 4				span 8							
span 6						span 6					
span 12											

Illustration of Bootstrap Grid System (W3Schools, 2016)

```

<!--START CONTENT-->
<div id="container" class="container-fluid">
  <div class="row">
    <div class="col-sm-12">
      <form action="search_results.php" method="POST">
        <div id="searchBox">
          <p>
            <input type="submit" id="searchBtn" name="searchBtn" class="btn btn-sm btn-default tiny" value="Go"></input>
            <input type="text" class="form-control" id="gSearchBox" name="gSearchBox" placeholder="Quick Recipe Search"></input>
          </p>
        </div>
      </form>
    </div>
  </div>
</div>

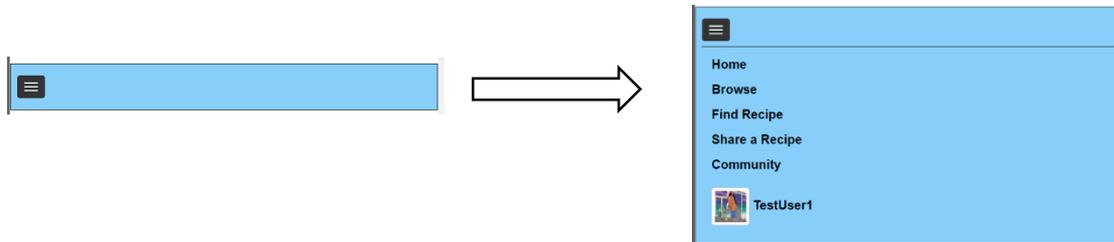
```

An example of implementing the grid system by wrapping code in rows and columns. In this example the column is described as “col-sm-12”. This means that content in this column will cover the full width of its parent row. The “sm” tag means that the page will use a small device such as a phone as the starting point and then scale up accordingly.

Each page will feature a header, navigation bar and footer in order to frame the content and provide a consistent means for users to navigate the site. The header will simply be the logo and name of the site, which will return the user to the home page if clicked on. The navigation bar will contain links to all of the main pages on the left side, and on the right will be a link to the login page. When a user is browsing the site as a guest, this will be presented as a small icon with a message asking if they want to register or sign in. There will be other prompts and redirects to the login page on other pages where the user must be signed in to complete an action (e.g. creating a recipe). Once a user is signed in, the navigation bar will change to reflect their new status and will instead display the user’s name and avatar. Clicking on either of these will then allow the user to access their account summary and take several actions related to their profile and published/saved content. As there are several items on the navigation bar, this can cause issues on smaller screens which do not have enough space to display them correctly. To remedy this, the navigation bar must be capable of being minimised to accommodate such devices. This can be achieved by using the Bootstrap “navbar-collapse” class which will hide menu options inside a toggle box on smaller screens.



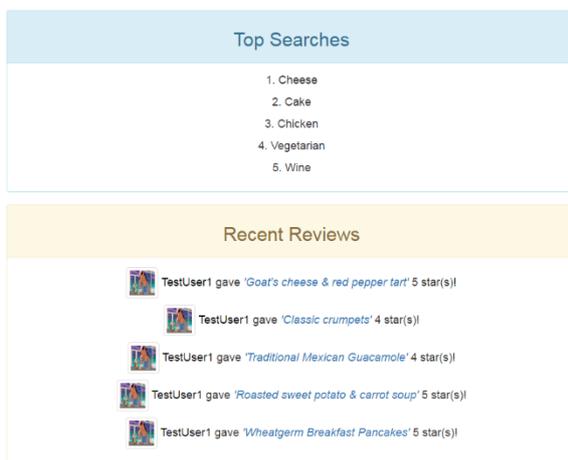
The navigation bar will reflect whether the user is signed in.



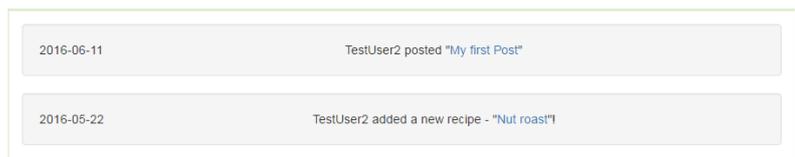
The Navigation bar should be minimised if the user is browsing on a mobile device.

The colour scheme will be neutral and minimalist, using black text on white backgrounds and pale colour shades to highlight distinct sections on each page. This will both improve the accessibility for those with vision impairments, and serve to better highlight the actual content.

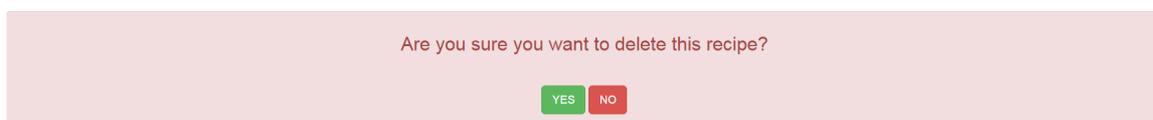
Several Bootstrap classes will be used to organise content on pages. “Alerts” are used to provide warnings and feedback to a user before or after they take an action. “Panels” can frame content in padded boxes with clear borders. “Wells” can also be used to distinguish items.



Using the “Panels” class

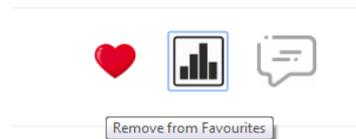


Using the “Wells” class



Using the “Alerts” class

When designing with usability in mind, the user should be able to intuitively understand the effects of their interaction with the site. This includes distinguishing inputs, links and buttons from non-interactive content and providing feedback on user's actions. This can be implemented in Chef's Table by including hover and click effects on buttons and links, and immediately notifying the user if an action they have taken was unsuccessful.



Hover effects can also be used to provide feedback on clicking a button.

Links should respond when the user hovers over them.

In Agile development (discussed in the analysis section), “Storyboarding” is a useful tool for identifying user requirements and expectations when interacting with the end application. Such storyboards should be ‘easy to understand’, capable of being iterated upon, and focus the developer on ‘common goals and expectations’ (Sherman, n.d.). The storyboards for Chef's Table can be found in [appendix 5](#).

4.3. Architectural Design

When a user first accesses the site on a desktop or mobile device, they will be presented with the home page, which offers a series of randomised recipe suggestions in a variety of categories. An example would be the ‘Top Rated’ category, which will select four of the most recent recipes which have an average rating of 4 or more stars. The randomised recipes will be provided by the ‘Build Recommendations’ PHP script, which will perform a series of SQL queries, store results in an array, and then select a random set to display.

The user can opt to either use the site as a guest or sign in/register in order to use the advanced features. Guest users will be able to access all the recipes hosted by the site, and view user's profiles and interactions. If the user is signed in, the home page will also provide more personalised suggestions (in addition to the randomised ones), according to factors such as other recipes the user has viewed, search history, and saved preferences. This feature is implemented by logging a user's activities in separate database tables, and then attempting to match this information using MySQL's ‘Natural Language Search’ tool. For example, if a user has previously searched for recipes containing ‘cheese’, the PHP script

for the recommendations page will query the database for any recipes which contain 'cheese' as an ingredient or in the title/description. The queries will also ensure that recipes created by the user will be omitted from the results, to avoid redundant suggestions.

The search feature of the website uses a similar technique to the recipe recommendations. A user can search according using general keywords, specific ingredients, a specific user, recently added recipes or those with a minimum required rating. General searches will try to match any columns in the recipes database table that contain the keywords specified. Ingredient search will limit its scope to the recipe ingredients table. User search will only return user profiles, rather than recipes.

[Appendix 6](#) contains an illustration of the architecture design for the site. A site map is also provided in [appendix 7](#).

4.4. Database Design

'Chef's Table' makes extensive use of a MySQL database for storage and data manipulation. After researching best practices, the following principles will inform the process of building and maintaining a relational database system for the application:

1. Data redundancy refers to the condition where 'the same piece of data is held in two separate places' (Techopedia, 2016). This is undesirable as it can 'waste storage space' and may 'lead to inconsistencies' (Hock-Chuan, 2010). A developer can reduce data redundancy by ensuring that each relationship between tables has been fully "normalised". "Normalisation" involves isolating data 'so that additions, deletions, and modifications of a field can be made in just one table', and these changes will then cascade through each linked table 'via the defined relationships' (Beal, 2016). There are three stages to normalisation and these will be described in the context of the system's data storage requirements.

The goal of "First Normal Form" (1NF) is to remove any 'repeating groups of data' and ensuring that each table has 'a primary key' (TutorialsPoint, 2016). For example, if the full details of the user were included each time a new recipe record was created, this would result in a very large table which would be cumbersome to search through. Repeating groups can be eliminated by splitting users and recipes into two separate tables, and then linking them via a shared key field (e.g. the UserID). Primary keys are unique columns which allows for the querying of 'each table row individually' and modifying 'each row without altering other rows in the same table' (Poollet, 1999). A composite primary key 'consists of more than one attribute to

uniquely identify an entity occurrence' (SQA, 2007), and is used in certain tables such as recipe ingredients (recipe ID to identify the recipe it belongs to, ingredient ID to identify its order in context of the other ingredients).

"Second Normal Form" (2NF) involves moving 'any non-key attributes that only depend on part of the table key' to a new table (SQA, 2008). For example, the table for storing details of a user's shopping lists and the items contained on them has a composite primary key made up of the list ID and username fields. Each item on the list is dependent only on the list ID, and thus can be moved to a separate table (shopping list items).

The final stage of normalisation, "Third Normal Form" (3NF) requires that the requirements of 1NF and 2NF have already been completed and that any columns 'that are not fully dependent upon the primary key' have been removed (Chapple, 2007). For example, each recipe will have up to ten preparation instructions. The actual text of the description is determined in instruction's ID (i.e. its order in the list relative to other instructions) and does not rely on the recipe ID. The instructions should therefore be moved to a separate table.

4.5. Website Structure and Content

Home page:

The main page for the site will provide a series of recipe recommendations, pulled from the recipe tables in the database. There will be two versions of the home page, based on whether the user is signed in or not. If they are a guest user, recipes suggestions will be provided under randomised headings (such as recently added, breakfast recipes or American style cooking). If the user is signed in, the suggestions will be more personalised i.e. they will take into account the user's saved preferences and view/search history.

The queries to the database which will build the recipe suggestions will be contained in an external PHP script, which executes when the user opens the page. A large number of suggestions will be prepared, and the order of the suggestions and the results pulled from the database will be randomised each time the user loads the page to provide variety to the user.

The category headings and results from the many SQL queries will be stored in two parallel arrays, and random results can be obtained by using PHP's "array_rand()" function to generate keys to the array. Once these elements from the array have been displayed, they are removed using PHP's "array_splice()" function to avoid duplicate suggestions.


```

//LOAD APPETISER SUGGESTIONS
$sql = "SELECT recipes.username, recipe_title, image, recipe_link, overall_rating, recipe_views, num_reviews, upload_date,
        profile_link, avatar_reference,
FROM recipes
INNER JOIN users
ON recipes.username=users.username
WHERE c_keywords LIKE '%Appetiser%'
ORDER BY RAND()
LIMIT 3";
$appetiser = mysqli_query($con, $sql);

if($appetiser AND mysqli_num_rows($appetiser) > 2){
    array_push($headerArray, "Appetisers");
    array_push($queryArray, $appetiser);
}

```

All relevant details of the recipe are selected, as well as some details of the author.

Results are filtered accordingly using the "Like" clause

The order of results is randomised for variety, and limited to 3 records.

If there are a sufficient number of results for the panel, the heading and results are added to their relevant arrays.

Users typically expect a search feature when they first access a website. Although there will also be a more extensive search feature on the dedicated "Find a Recipe" page, a quick search option will be provided in the top corner of the home page.



Recommended for you

Quick search feature on home page

Login/Register/Welcome Pages

To login, the user will be prompted to enter the username and password selected during registration. If these cannot be matched to a record in the user table, the user will be informed with an error message and will not be able to proceed. If the login is successful, the username and avatar will be saved to the PHP session variable which allows the site to remember a user for a short period of time, and the user will be redirected to the previous page they visited. The navigation bar will also be updated to reflect the user's status, and they will now be able to access every feature such as creating recipes and accessing the account section.

```

//IF NO SIGNIN ERRORS, LOAD USER TO SESSION
if(empty($signinError)){

    $sql = "SELECT avatar_reference
            FROM users
            WHERE username='$signinname'";
    $result = mysqli_query($con, $sql);
    $getAvatar = mysqli_fetch_row($result);

    //GET USERNAME AND AVATAR AND ASSIGN TO SESSION DATA
    $signinavatar = $getAvatar[0];
    $_SESSION["username"] = $signinname;
    $_SESSION["avatar"] = $signinavatar;

    //CLOSE DB CONNECTION
    mysqli_close($con);

    //REDIRECT TO INDEX PAGE
    if(empty($_SESSION["redir"]) OR !isset($_SESSION["redir"])){

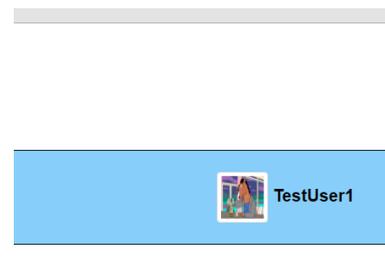
        header( "Location: index.php" );

    }//if
    else {

        header( "Location: ".$_SESSION["redir"] );

    }//else
} //if

```



The User is now signed in.

If the login attempt is successful, the user's details will be saved to the session variable.

If a user has not registered yet, they can complete the new user form. Here they will be asked to provide details such as a username, password, email address and an avatar for their profile. If each required field is completed and no errors have been found, a new user record will be created and the details will be saved to the session variable. The user will then be redirected to the welcome page which will confirm that the new account has been successfully created and allow the user to set a series of preferences which will then be used to tailor recipe suggestions to their needs. These preferences include specifying what type of food they like, and any dietary requirements they might have. These will then be stored in a preferences record for that user.

Create An Account

Username	<input type="text" value="NewUser"/>
Password	<input type="password" value="....."/>
Confirm Password	<input type="password" value="....."/>
Email Address	<input type="text" value="NewUserEmail@email.com"/>
First Name	<input type="text" value="New"/>
Surname	<input type="text" value="User"/>
Avatar (Optional)	
	
<input type="button" value="Choose File"/> Users-User-Male-Icon.png <input type="button" value="Clear"/>	
<input type="button" value="Create Account"/>	

Creating a new account.

Chef's Table can offer personalised recipe suggestions based on what you like and any dietary requirements that you might have. If you would like to use this feature, please complete the following section.

I like...

Please separate likes with commas

I have the following dietary preferences and/or allergies...

<input type="checkbox"/> Vegetarian	<input type="checkbox"/> Vegan	<input type="checkbox"/> Pescatarian	<input type="checkbox"/> Kosher	<input type="checkbox"/> Halal	<input type="checkbox"/> Dairy allergy
<input type="checkbox"/> Diabetic	<input type="checkbox"/> Gluten Free/Celiac	<input type="checkbox"/> Peanut allergy	<input type="checkbox"/> Egg Allergy	<input type="checkbox"/> Shellfish Allergy	<input type="checkbox"/> Soy Allergy

New users will be given the option to set a number of preferences after creating an account. These can then be edited from their account page.

These forms (and all other forms on the site) will be processed using PHP to record data and either add or compare to other values in the relevant database tables. Each form will have appropriate error handling and user feedback to detect and prevent erroneous or problematic values being stored. Full details of forms and error handling can be found in [appendix 9](#).

Creating and viewing recipes

A signed in user can publish a recipe to Chef's Table using the form for creating a recipe, which gathers all required information about the recipe, ensures each field is valid, and then adds a new record to the recipes, ingredients and instructions table.

The first two fields will ask the user to provide a title and short description to their recipe. They will then be given the option to upload an image for the recipe. A dynamic preview of the uploaded image will be provided so that the user can quickly preview how the image will appear. Recipe images (and any other images that a user can upload to Chef's Table) must meet several criteria such as a maximum file size and file type. If the image is acceptable, the file name will be updated and stored in the corresponding image directory on the site. A reference to the file in the directory will then be included in the record for the recipe in the database.

Would you like to upload a picture of your finished dish?



Clear

Choose File rice_and_curry.jpg

Keywords and Tags

Users can preview their uploaded image before publishing

```

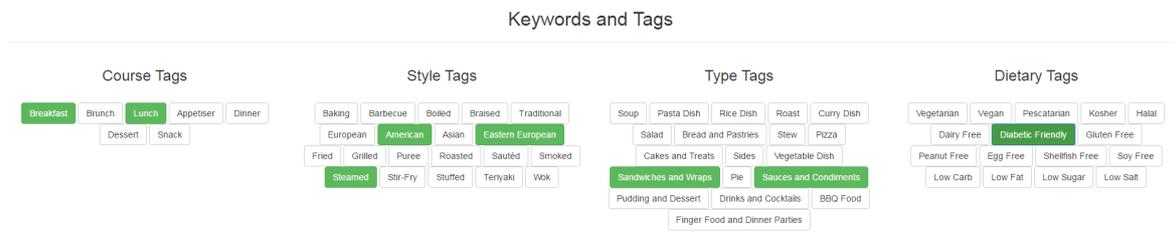
//CHECK THAT A FILE HAS BEEN UPLOADED
if ($_FILES['uploadImage']['error'] != UPLOAD_ERR_NO_FILE) {
    //CREATE A NEW FILE NAME
    $rand1 = rand(1, 99); $rand2 = rand(1, 99); $rand3 = rand(1,99);
    $file_name = substr($title, 0, 3).substr($username, 0, 3).$rand1.$rand2.$rand3;
    //SET A MAX FILE SIZE
    $max_file_size = 1097152;
    //SET A TEMP FILE FOR TESTING BEFORE SAVING IMAGE TO DIRECTORY
    $file_tmp = $_FILES['uploadImage']['tmp_name'];
    $file_type = $_FILES['uploadImage']['type'];
    //GET THE FILE TYPE FOR VALIDATION PURPOSES
    $fname = $_FILES["uploadImage"]["name"];
    $efName = explode(".", $fname);
    $efName = end($efName);
    $file_ext = strtolower($efName);
    //STORE AN ARRAY OF ACCEPTABLE FILE EXTENSIONS
    $extensions = array("jpeg", "jpg", "png");
    //CHECK THAT FILETYPE IS ACCEPTABLE
    if(in_array($file_ext, $extensions) === false) {
        if(isset($_FILES['uploadImage'])) {
            $imageError = "Sorry, only JPEG & PNG files permitted.";
        }
    }
    //CHECK FILE SIZE ACCEPTABLE
    if($_FILES['uploadImage']['size'] >= $max_file_size) {
        $imageError = "File size must not exceed 1MB";
    }
    //IF NO ERRORS, UPLOAD IMAGE TO DIRECTORY
    if(empty($imageError) == true) {
        move_uploaded_file($file_tmp, "recipe_images/".$file_name.".".$file_ext);
        $imageReference = "recipe_images/".$file_name.".".$file_ext;
    }
}
//IF NO IMAGE UPLOADED, USE DEFAULT INSTEAD
else {
    $imageReference = "images/no_recipe_image.png";
}
}

```

Processing images using PHP form handling

The next section involves the user selecting keywords to describe the user. These will be provided under four headings – Course, Style, Type and Dietary info. These tags improve the chances for a recipe to be discovered by other users via the recommendations, search and category pages. Each keyword will be presented as a button, and when a user clicks on it the button will change to show it has been selected. This will be achieved by a simple

JavaScript function which changes the class of the button when the user clicks it. A hidden checkbox will also be ticked by the JavaScript function, which will enable the PHP script to detect which buttons have been clicked. There will be an array for each type of keyword, and on submission, each selected tag will be added to the relevant array. If there are no errors, the elements in each array will be converted into a single string using PHP's "implode" function, and added to the relevant column in the recipes table.



The user can select a variety of tags and keywords to improve the discoverability of their recipe on the site.

```

if(empty($cKws) AND empty($tKws) AND empty($sKws) AND empty($dKws)){
    $keywordsError = "Please select at least one keyword.";
}
//if
else {
    if(!empty($cKws)){
        $cString = implode(",",$cKws);
    }
    //if
    if(!empty($tKws)){
        $tString = implode(",",$tKws);
    }
    //if
    if(!empty($sKws)){
        $sString = implode(",",$sKws);
    }
    //if
    if(!empty($dKws)){
        $dString = implode(",",$dKws);
    }
    //if
}
//else

//Process preparation info
$difficulty = $_POST["proficiencyOption"];
if($difficulty == "-"){
    $diffError = "Please select a difficulty rating for your recipe.";
}
    
```

User must select at least one keyword to proceed.

The selected tags in each category are stored in separate arrays and converted in separate strings when the form is submitted using the "implode" function.

		recipe					
<input type="checkbox"/>	Edit Copy Delete	44	TestUser4	Warm beet, chorizo and pear salad	Make good use of succulent pear, colourful beetroot...	Lunch,Appetiser	Salad,Sides,Vegetable Sides
<input type="checkbox"/>	Edit Copy Delete	45	TestUser4	Salt beef club with Cajun fries	An indulgent solo supper of a layered sandwich wit...	Breakfast,Lunch,Dinner	Sandwiches and Wraps
<input type="checkbox"/>	Edit Copy Delete	46	TestUser4	Cheddar cheese & shallot pie with fennel seed ...	A hearty pie with a mustard and fennel crust, this...	Dinner	Vegetable Sides,Pie Bake,Traditional Vegetarian
<input type="checkbox"/>	Edit Copy Delete	47	TestUser4	Brandy (or rum) sauce	Add brandy or rum depending on personal preference...		Sauce and Condiments

The tags are stored in Full-Text fields in the recipes tables, which allows natural language searches to gather results for the recommendations and search pages.

The user can specify the difficulty of the recipe, estimated number of servings, time to prepare and time to cook from simple drop down menus which provide a range of potential options.

The user can add up to 20 ingredients for their recipes, and can specify a quantity in a separate field if they wish. Rather than displaying all ingredient text fields at once and

cluttering up the screen, the user can add or hide ingredient fields as they go using the two JavaScript enabled buttons.

```
//The IDS for each ingredient field is stored in an ordered array.
var ingredientSections = ["ingredientTwo", "ingredientThree", "ingredientFour", "ingredientFive",
    "ingredientSix", "ingredientSeven", "ingredientEight", "ingredientNine", "ingredientTen",
    "ingredientEleven", "ingredientTwelve", "ingredientThirteen", "ingredientFourteen", "ingredientFifteen",
    "ingredientSixteen", "ingredientSeventeen", "ingredientEighteen", "ingredientNineteen", "ingredientTwenty"];

//The pointer which will select the next field to show or hide is initialised.
var ingredientPointer = -1;
var next;

//Function that is called when user clicks the add button
function addIngredient(){
    //First, check that there are still ingredient sections left to display.
    if(ingredientPointer != 18){
        //If there is, increment the array pointer.
        ingredientPointer++;
        //Assign the id of the next ingredient field to a variable
        next = ingredientSections[ingredientPointer];
        //Change the class name so that the next field becomes visible
        document.getElementById(next).className = "ingredientSections";
    }//if
};//function

//Function that is called when user clicks the remove button.
function removeIngredient(){
    //First, check that Ingredient fields 2-20 are not already hidden.
    if(ingredientPointer != -1){
        //Assign the id of the last ingredient field to a variable
        next = ingredientSections[ingredientPointer];
        //Change the class name so that the ingredient field becomes invisible.
        document.getElementById(next).className = "hiddenIngredientSections";
        //Remove any values from the field, to avoid incorrect or empty values being added when the form submits.
        document.getElementById(next).value = "";
        //Decrement the ingredient pointer.
        ingredientPointer--;
    }//if
};//function
```

The JavaScript function for adding and removing ingredients (a similar function is used to add and remove instructions).

The user can then add up to ten instructions for their recipe. Like many other text inputs on the site, each instruction and ingredient will be fully tested for problematic characters before being added to the database.

```
//FUNCTION TO TEST TEXT INPUT
//Pass the data in question as a parameter
function test_input($data) {
    //Remove trailing spaces
    $data = trim($data);
    //Removes backslashes (which can be used to 'escape' strings)
    $data = stripslashes($data);
    //Converts certain characters to HTML entities (e.g. < becomes &lt;, & becomes &amp etc)
    $data = htmlspecialchars($data);
    //Remove semi-colons, which can cause issues with MySQL
    $data = str_replace(";", " ", $data);
    return $data;
};//function
```

```

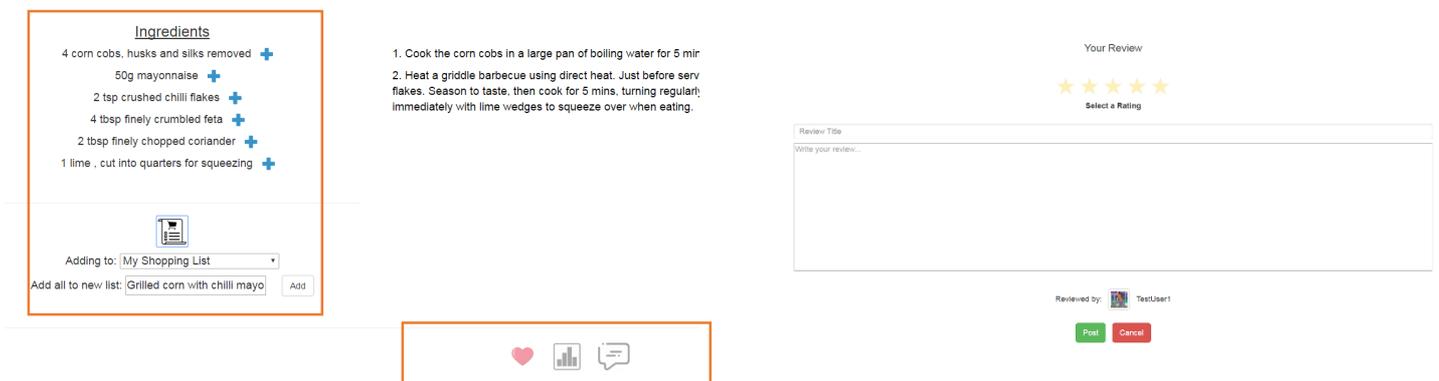
.se {
  //Sanitise inputs
  //(This is used to prevent SQL injection, which occurs when malicious users are able
  // to send SQL statements to a database via forms on websites.)
  $title = mysqli_real_escape_string($con, $title);
  $description = mysqli_real_escape_string($con, $description);

  foreach($qArray AS $quantity){
    $quantity = mysqli_real_escape_string($con, $quantity);
  }//foreach
  foreach($iArray AS $ingredient){
    $ingredient = mysqli_real_escape_string($con, $ingredient);
  }//foreach
  foreach($sArray AS $instruction){
    $instruction = mysqli_real_escape_string($con, $instruction);
  }//foreach
}

```

Some of the methods used to validate data.

If no errors have been found, the user's recipe will be added as new records in the recipes, ingredients and instructions tables. The recipe can then be viewed by others using the recipe viewer, which shares a similar layout to the create recipe form to promote a sense of consistency. A user can view a recipe regardless of whether they are signed in or not. However, additional features will be presented to users who have logged in, including adding the recipe to their favourites, writing a review, and adding ingredients to a shopping list. These features will be discussed further in the implementation section.



Signed in users will be able to access more advanced features when viewing a recipe, such as the shopping list and recipe review functions.

Browse and Category pages

If the user is not looking for a specific recipe and wants to browse the variety of recipes offered by users on Chef's Table, they can use the browse feature. The user will select a category and then will be brought to the specific page for that category. Recipes in that category will be subdivided according to the most recent, most popular and highest rated. A-Z listings of recipes in that category will also be presented in increments of five recipes.

Select a Category



Breakfast and Brunch

[Lunch and Salads](#)

Appetisers and Snacks

The user can select a category...

Lunch & Salad Recipes

Recently Added

 <p>Help-yourself tuna rice salad</p> <p>★★★★</p> <p>TestUser3</p> <p>Views Reviews</p>	 <p>Roasted sweet potato & carrot soup</p> <p>★★★★★</p> <p>TestUser4</p> <p>Views Reviews</p>	 <p>California quinoa & avocado salad</p> <p>★★★★</p> <p>TestUser4</p> <p>Views Reviews</p>
--	--	--

A-Z

	<p>Baked sea bass with lemon caper dressing</p> <p><i>"This elegant, gluten-free main is special enough for a dinner party, yet simple and quick to make for no fuss entertaining"</i></p> <p>★★★★</p> <p>TestUser5 2016-05-23</p> <p>Views Reviews</p>
	<p>California quinoa & avocado salad</p> <p><i>"The tahini and avocado combo in this dressing is totally addictive. If you have ripe avocados to use up, this is a winning recipe"</i></p> <p>★★★★</p> <p>TestUser4 2018-05-23</p>

... and are then presented with the specific page for that category.

The script which fetches recipes for each category page is quite similar to that used for recommendations, as categories such as recently added, most popular etc. will be displayed. The following screenshots explain how the A-Z panel for each category is generated:

```

<?php
//GET URL
$url = $_SERVER["REQUEST_URI"];

//GET POSITION OF CATEGORY ID IN HEADER
$position = strpos($url, "=") + 1;

//SET CATEGORY IN ORDER TO DISPLAY RELEVANT DETAILS FROM DATABASE
$category = substr($url, $position);

```

Rather than creating a separate page for each category, a single page can be used for each category using the following technique. When the user selects a category from the "Browse" page, the category will be appended to the "View Category" link. For example, if the user selects the "Breakfast/Brunch" category, the link will be "view_category.php?type=breakfast_brunch". The category can then be determined from the link using PHP's string manipulation function, as above.

```
switch($category) {
    case "breakfast_brunch":
```

A switch statement will be used to decide which series of queries should be run.

```
$sql = "
SELECT recipes.recipe_link, recipes.image, recipes.recipe_title, recipes.recipe_description, recipes.upload_date, recipes.overall_rating, recipe_views, num_reviews
    users.profile_link, users.username, users.avatar_reference
FROM recipes
    INNER JOIN users
        ON recipes.username=users.username
WHERE (
    MATCH(recipe_title) AGAINST('%Breakfast%' IN NATURAL LANGUAGE MODE)
    OR
    MATCH(recipe_title) AGAINST('%Brunch%' IN NATURAL LANGUAGE MODE)
    OR
    MATCH(recipe_description) AGAINST('%Breakfast%' IN NATURAL LANGUAGE MODE)
    OR
    MATCH(recipe_description) AGAINST('%Brunch%' IN NATURAL LANGUAGE MODE)
    OR c_keywords LIKE '%Breakfast%'
    OR c_keywords LIKE '%Brunch%'
)
GROUP BY recipe_title
ORDER BY recipe_title ASC";

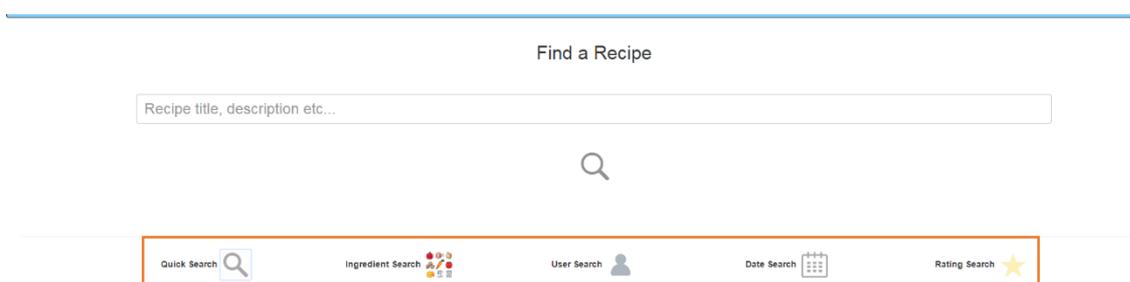
$all = mysqli_query($con, $sql);

```

Any recipes which fall under the category of "Breakfast/Brunch" will be selected using the query above, which again uses a Full Text Natural Language Search to find any relevant records. The results will then be grouped and sorted by their title, in alphabetic order.

Recipe Finder

As well as the quick search feature on the home page there will be an advanced search page which allows users to apply filters to their search queries. These will include recipes containing a specific ingredient, those uploaded recently and those with a minimum star rating specified by the user. A user can also find other users via their username, surname, email address etc.



The user should be able to search using a variety of criteria.

Appropriate results will be returned by an external PHP script which sends the relevant query to the database based on the type of search the user has selected. For example, in an ingredient search the ingredient tables will be searched for close matches and the recipe ID will be matched with the full details of the recipe record in the recipe table. In a date search,

results will be filtered to contain only recipes uploaded within the time frame specified by the user. The search script will be further detailed in the implementation section.

The following is an example of carrying out a general search:

Find a Recipe

Steak



User enters a search term.

```
$search = $_POST["gSearchBox"];
$search = mysqli_real_escape_string($con, $search);
```

The query is sent to the PHP script for searches

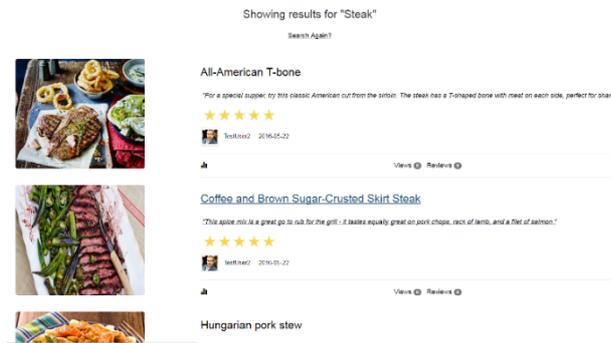
```
WHERE (
    MATCH(recipe_title) AGAINST('%$search%' IN NATURAL LANGUAGE MODE)
    OR
    MATCH(recipe_description) AGAINST('%$search%' IN NATURAL LANGUAGE MODE)
    OR
    MATCH(c_keywords) AGAINST('%$search%' IN NATURAL LANGUAGE MODE)
    OR
    MATCH(d_keywords) AGAINST('%$search%' IN NATURAL LANGUAGE MODE)
    OR
    MATCH(s_keywords) AGAINST('%$search%' IN NATURAL LANGUAGE MODE)
    OR
    MATCH(t_keywords) AGAINST('%$search%' IN NATURAL LANGUAGE MODE)
    OR
    MATCH(recipe_ingredients.description) AGAINST('%$search%' IN NATURAL LANGUAGE MODE)
)
GROUP BY recipe_title";
```

The relevant tables are searched comprehensively for any recipes which contain similarities to the search query in their title, description, keywords or ingredients.

```
GROUP BY recipe_title";
$results = mysqli_query($con, $sql);
$numResults = mysqli_num_rows($results);
$searchHeading = $search;
```

The query is executed and the number of results counted (which will be displayed to the user).

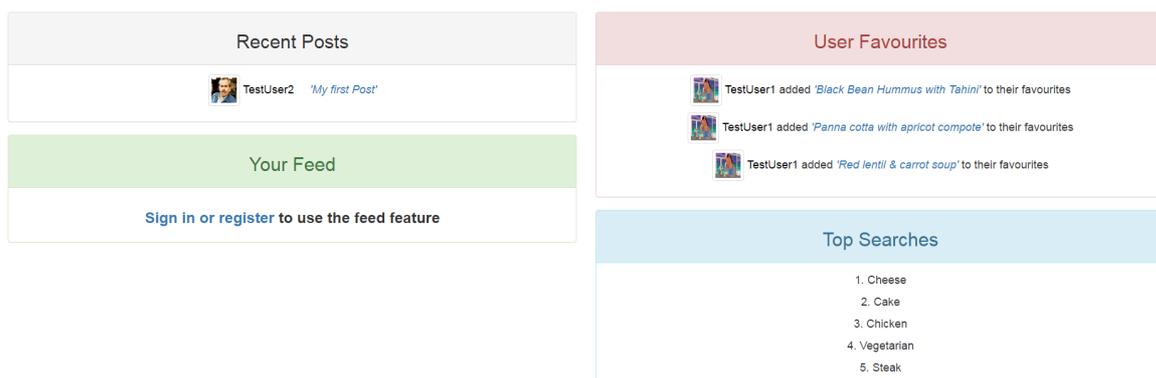
Results will be displayed on a separate page in a vertical list format. Clicking on any of these results will bring the user to the relevant recipe/user page. If no results have been found, an alert should be displayed.



Displaying search results to the user.

Community Page

This page will highlight recent activity on the site, and can be viewed whether the user is signed in or not. It will provide links to recently added profile posts, recent recipe reviews and recipes which other users have added to their favourites. A list of the most popular search terms will also be displayed and updated frequently. Each block of content will be stored in a separate colour coded panel. The content for each panel will be generated in an external PHP script. For example, the trending searches will be generated by counting the occurrences of matching values in the user search history, sorting in descending order, and then limiting results to five values. Similarly, the recent posts will be gathered by sorting the records in the user posts table by their date field and taking the top five results.

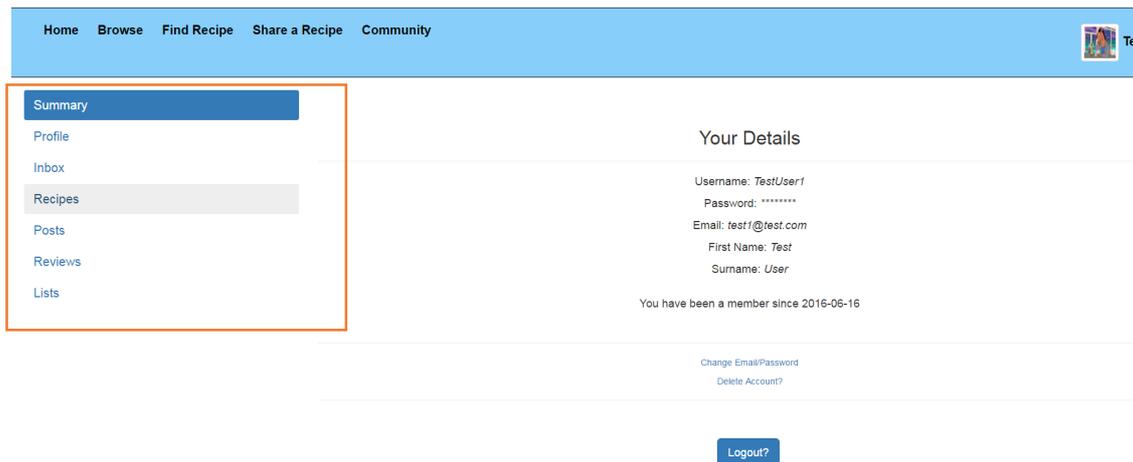


The community page will provide updates on recent activity taking place on the site.

If the user is signed in, an additional panel will provide a personalised feed which provides updates on users that a person has added to their "Followed" list (See *Profile* page below). This will include posts, comments, reviews and recipes uploaded by these followed users.

Account Page

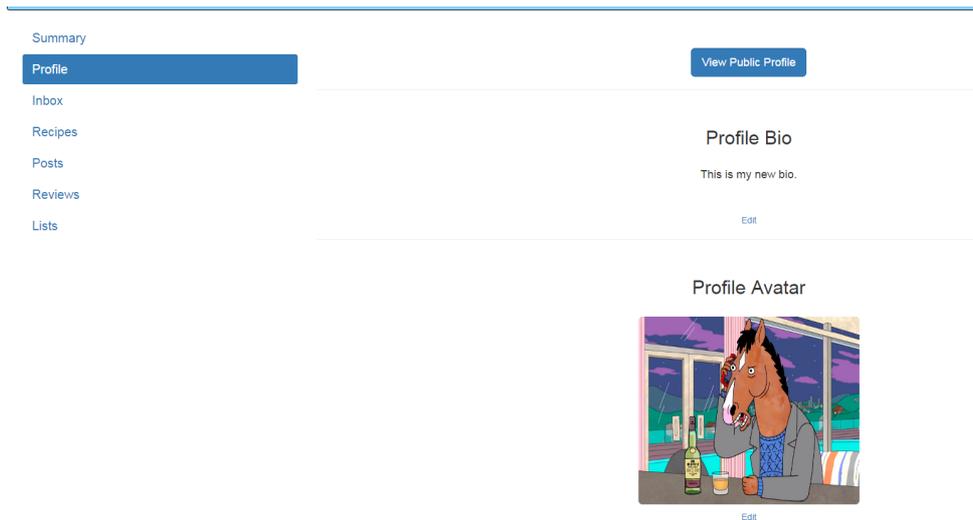
When the user is signed in, they will gain access to the account page which allows them to take any sort of action relating to their profile or published/saved content. This is also where the user can read and respond to messages from other users. The account summary page will contain links to each of the account pages in a vertical menu on the left side of the page. This menu will be consistent across each account page.



The account summary page.

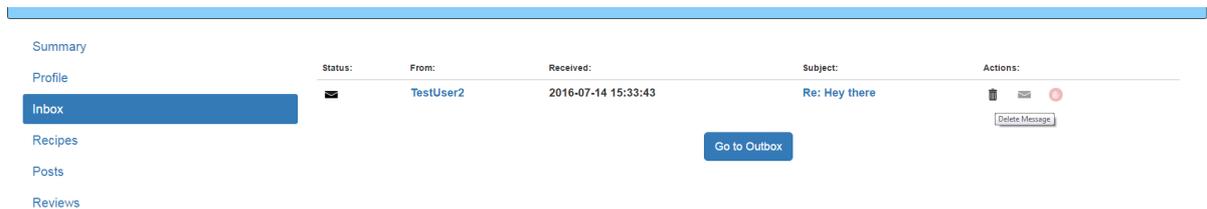
The account summary page will contain options for changing the email/password, deleting the account and manually logging off. If the user opts to delete their account, all content relating to that user must be deleted to reduce data redundancy in the database. This includes the user's profile, preferences, recipes, reviews and comments.

The profile page will allow the user to edit their bio, change their avatar, and set their privacy settings. The privacy settings will determine the amount of publically visible content on the user's profile page and the ability for other users to send them messages.



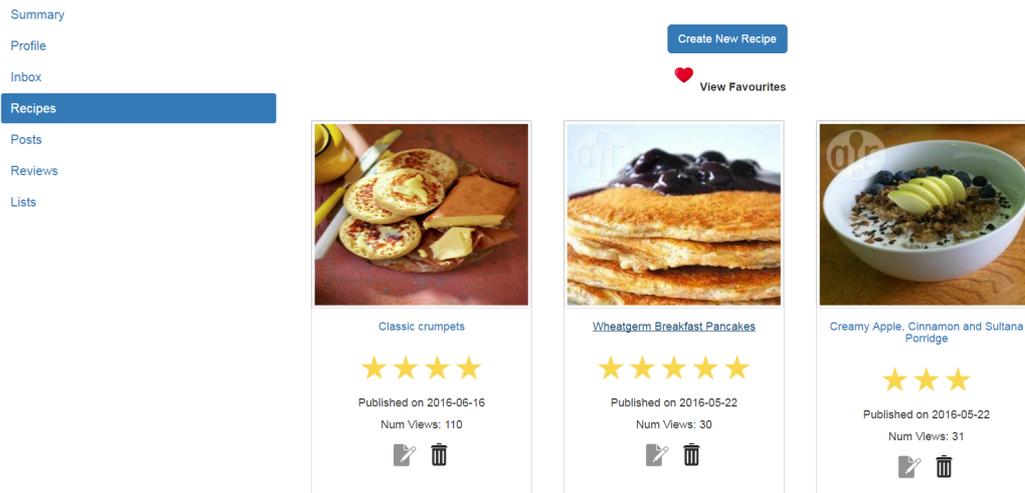
Profile page

The inbox will display any received or sent messages. Messages in the inbox will be formatted to show whether they have been read or not, and the user will be able to open the message in the message viewer, delete the message, or block the user from sending them any more messages. If there are no messages, an alert should inform the user.



Inbox page

The recipes section of the account will store all of the recipes a user has published, and allow them to make changes or delete a recipe. Recipes that a user has added to their favourites will also be stored here.



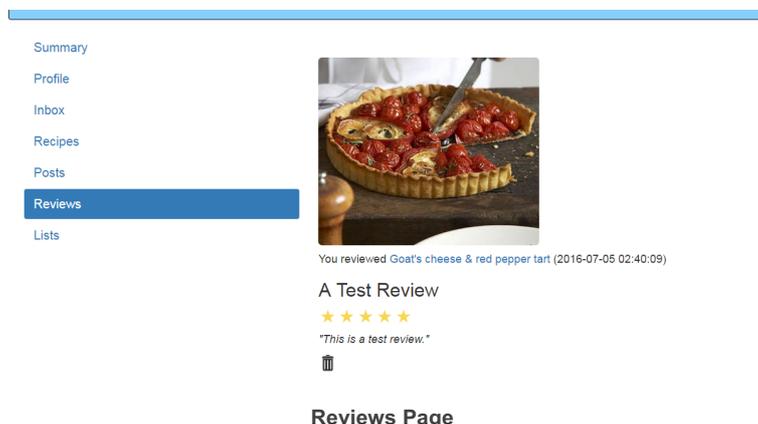
Published and saved recipes page

If the user wants to add a new post to their profile, or modify existing posts, this can also be performed using the account page.



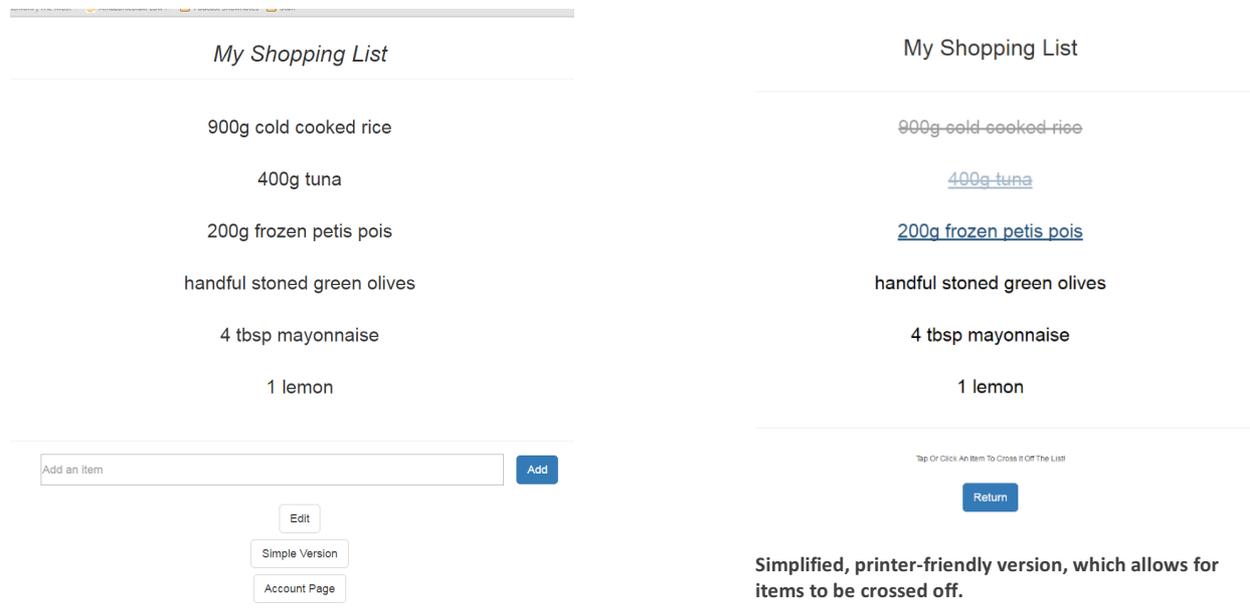
The posts section of the account page

Any reviews that a user has submitted for a recipe will be accessible from their account, and can be viewed or deleted.



Reviews Page

Finally, the account section contains the shopping list feature. The user will be able to manage their lists and create new ones. When a list is opened, the user can edit/remove existing items, add new items to the list, and open a simplified version of the page designed to be printer-friendly. On the simplified version, the user will be able to cross off items by clicking/tapping on them.



A saved shopping list

As described above, shopping list functionality will be integrated into the recipe viewer and will allow users to add one or ingredients from a recipe to a saved shopping list, or add all ingredients to a new list.

Profile Viewer

Each user will have a publically visible profile page, which will include details such as the user's bio and recent activity, as well as stats such as number of followers and profile views. The link to this page will be included beside any activity the user partakes in on the site, e.g. any recipes they create or comment that they write. Similarly, to the category page, one page can be used to display any user's profile, using the PHP string manipulation described above. Underneath the user's profile will be an action bar which allows a visiting user to add them to their "Followed" list, send a direct message and view all recipes or posts by that user. Another means of interaction is to leave a comment on the user's profile which can be seen by others.

Info about the user



TestUser5
Member since: 2016-05-22

Bio

I am test user 5!

Profile Views **11** Followers **0**



Action bar for interacting with the user and their content

Recent Activity

- 2016-05-23 TestUser5 added a new recipe - "Barley, chicken & mushroom risotto"!
- 2016-05-23 TestUser5 added a new recipe - "Low sugar chocolate sandwich cake"!
- 2016-05-23 TestUser5 added a new recipe - "Thai red duck with sticky pineapple rice"!
- 2016-05-23 TestUser5 added a new recipe - "Chef John's Cauliflower Pizza Crust"!
- 2016-05-23 TestUser5 added a new recipe - "Apple Cider Muffins (gluten & dairy free)"!
- 2016-05-23 TestUser5 added a new recipe - "Spatchcocked duck with posh peas and celeriac chips"!
- 2016-05-23 TestUser5 added a new recipe - "Goat's cheese & red pepper tart"!
- 2016-05-23 TestUser5 added a new recipe - "Panna cotta with apricot compote"!
- 2016-05-23 TestUser5 added a new recipe - "Harissa chicken traybake"!
- 2016-05-23 TestUser5 added a new recipe - "Low-fat roasties"!

A feed for all the public actions the user has recently taken.

Comments

No Comments!

Any comments will be displayed here.

 TestUser1

Any user can post a comment to a profile page

5. Implementation

5.1. Development tools and training undertaken

Before commencing development of the system, it was important for the developer to gain a theoretical understanding and practical experience of using the tools in order to deliver an efficient and intuitive application. To this end, the two main resources which were used at the training phase were the variety of tutorials provided by “W3 Schools” and the collaborative forum for developers from a variety of disciplines and backgrounds at “Stack Overflow”.

“W3 Schools” aims to help beginners rapidly learn the essentials of web development and programming. It has a range of tutorials covering each aspect of building a website, from HTML/CSS to PHP/SQL, which ‘focus on simplicity, practicing easy and straight-forward learning’ (W3 Schools, 2016). The developer completed all of the tutorials which were relevant to the aims of Chef’s Table and ensured that the best practices for modern web design outlined were used as a constant reference point throughout development.

“Stack Overflow” has a very large user base, receiving around ‘400 million visitors each month’, and a significant repository of ‘3.7M’ questions covering the majority of development issues (Stack Overflow, 2016). This resource was very useful for troubleshooting any specific issues which were encountered, as well as benefitting from a shared pool of public knowledge and experience.

5.2. Design prototype and objectives for implementation phase

On completion of the design phase, a simple working prototype for Chef’s Table has been developed which meets the minimum system requirements identified at the analysis stage. The majority of the page layouts and the general user interface have been finalised and implemented. Using the AGILE approach to development, three “sprints” have been completed which have resulted in the user account, recipe creation/recipe viewing and recipe recommendation features being fully functional, with preliminary development commenced on several other features.

The goal for the implementation section is to build on the work completed so far by integrating components (e.g. providing signed in users with personalised recommendations) and continuing to improve upon the usability and accessibility of the system for the end user. Action will be taken to remedy any issues identified in the feedback from the prototype demonstration, such as improving usability for mobile users.

The incremental steps taken to produce a final working system will now be described.

5.3. Further development and iteration

Home Page

The home page currently uses a PHP script to build a randomised series of recipe recommendations for the user when they access the site. If the user is signed in, a slightly different script will be used. As well as all of the categories for guest users, a personalised set of recommendations for the user will be generated. These will be based on:

1. Recipes the user has previously viewed.
2. Recipes the user has added to their favourites.
3. Recipes which the user has submitted a positive review about.
4. The user's search history.
5. The preferences the user has specified in their account settings.

The SQL queries for each of these will differ slightly, however most will take advantage of MySQL's "Full-text search" feature. By specifying columns in a table as "FullText", comparisons can be made between the values contained in the column and user-specified string queries. Such queries will return any records that 'contain at least one match', with a match occurring when the specified column 'contains all of the terms specified in the full-text query', as well as other criteria 'such as the distance between the matching terms' (Microsoft Developer Network, 2016). This type of query has the advantage of finding many more potential recipe suggestions, as it will include records that are not exact matches, but very similar.

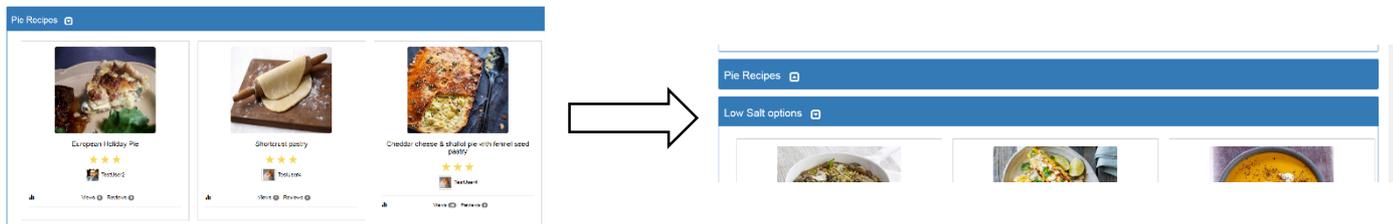
The following is an example of such a query, one in which suggestions are made based on previous recipes a user has viewed:

```
//LOAD LAST VIEWED RECIPES

$sql = "SELECT user_view_history.recipe_id, recipes.recipe_title, recipe_description, c_keywords, t_keywords, s_keywords, d_keywords, recipe_link
FROM user_view_history
INNER JOIN recipes
ON user_view_history.recipe_id=recipes.recipe_id
WHERE user_view_history.username='$username'
ORDER BY RAND()
LIMIT 1";
$query = mysqli_query($con, $sql);
```

Firstly, the ID for a recipe a user has recently viewed is selected from the "View History" table. This is then used to get the full details of the relevant recipe from the recipes table. The line "ORDER BY RAND()" is used to get a random entry from the user's view history.

smartphone/tablet users. To alleviate this the recommendations will be divided into increments of six, and the user can choose to load more if they wish. For ease of navigation and to allow the user to remove suggestions they are not interested in, each panel should be capable of being minimised.



For ease of navigation, users should be able minimise the amount of scrolling they need to perform to get to the recommendations they are interested in.

```

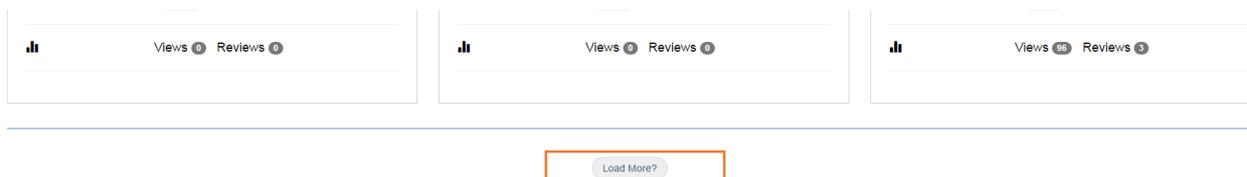
var panel = "panel_"; var up = "up_"; var down = "down_";

function hide_this(num){
    document.getElementById(panel.concat(num)).className = "hidden";
    document.getElementById(up.concat(num)).className = "glyphicon glyphicon-collapse-up";
    document.getElementById(down.concat(num)).className = "hidden";
}

function show_this(num){
    document.getElementById(panel.concat(num)).className = "panel-body";
    document.getElementById(up.concat(num)).className = "hidden";
    document.getElementById(down.concat(num)).className = "glyphicon glyphicon-collapse-down";
}

```

JavaScript function for hiding panels.



Rather than overwhelming the user with every suggestion at once, they should be presented in batches of 6 with an option to load more. The load more button will fire a simple JavaScript function which tracks the current number of results which have been displayed, and then loads more as appropriate.

Search Page

In addition to the general search option, additional search options are provided for users who are looking for more specific results. Each search option is contained within separate divider elements, and will be displayed or hidden according to which the user has selected. Again, this will be achieved by a JavaScript function which uses CSS classes to hide or show elements.

The search options provided to the user are:

1. Ingredients – This could be useful if a user has some left over ingredients at home that they do not want to waste. For a more exact search, the user can also optionally specify a quantity for the ingredient.

2. User search – This would allow a user to find any family or friends who might also be using the site, as they can search using an email address or first name/surname.
3. Date search – This could be used if the user only wants to see recent recipes for example.
4. Rating search – This could be used if the user is only interested in recipes that have obtained a seal of quality from the community.

The following is an example of an ingredient search:

Ingredient Search

User selects ingredient search from menu and enters their search term, in this case also specifying a quantity.

```
PROCESS INGREDIENT SEARCH
if(!empty($_POST["iSearchBox"])){

    $con = new mysqli("127.0.0.1", "root", "", "chefstable");

    if($con->connect_error) {
        die("connection failed: " . $con->connect_error);
    }

    $ingredient = $_POST["iSearchBox"];
    $ingredient = mysqli_real_escape_string($con, $ingredient);

    if(!empty($_POST["iQSearchBox"])){
        $quantity = $_POST["iQSearchBox"];
        $quantity = mysqli_real_escape_string($con, $quantity);
    }
}
```

The data is passed to the search results PHP script for processing.

```

$sql = "SELECT recipes.recipe_link, recipes.image, recipes.recipe_title, recipes.recipe_description, recipes.upload_date, recipes.overall_rating , reci
users.profile_link, users.username, users.avatar_reference,
recipe_ingredients.quantity, recipe_ingredients.description
FROM recipes
INNER JOIN users
ON recipes.username=users.username
INNER JOIN recipe_ingredients
ON recipes.recipe_id=recipe_ingredients.recipe_id
WHERE (
recipe_ingredients.quantity LIKE '%$quantity%'
AND
MATCH(recipe_ingredients.description) AGAINST('%$ingredient%' IN NATURAL LANGUAGE MODE)
)";
$results = mysqli_query($con, $sql);
$numResults = mysqli_num_rows($results);
$searchHeading = $quantity." ".$ingredient;
    
```

An attempt is made to match the search with any records in the recipe ingredients table. If matches are found, the full details of the recipe can be obtained via an "Inner Join" procedure.

Showing results for "2 red peppers"

[Search Again?](#)

The search results page returns any matching recipes...




Goat's cheese & red pepper tart

There is a little work involved in this, but there is so much satisfaction in bringing it to the table to share with family or friends

★★★★★

 TestUser5 2016-05-23

Views 1 Reviews 1

Help-yourself tuna rice salad

Ideal for busy households, this hearty salad keeps well in the fridge for up to three days, ready to be spooned into a bowl whenever you're peckish

★★★

 TestUser3 2016-06-16

Views 1 Reviews 0

Difficulty: Difficult

25 Minute

Ingredients

- 2 large red onions
- 2 tbsp olive oil
- 1 tbsp balsamic vinegar
- 12 pitted black olives
- 2 red peppers
- 200g ready made shortcrust pastry
- 250g cherry tomato

- 1. C
- furt
- 2. H
- coo
- 3. H
- Ren
- 4. S
- goa
- 5. B
- tem

Ingredients

- 900g cold cooked rice
- 400g tuna
- 200g frozen petis pois
- 2 red peppers
- 3 tomatoes
- 5 spring onions
- Bunch flat-leaf parsley
- handful stoned green olives
- 4 tbsp mayonnaise
- 1 lemon
- 2 tbsp extra virgin olive oil

- 1. Th
- toma
- 2. Sti
- your

```

if($signedIn){
    //Set correct time zone for server
    date_default_timezone_set('Europe/London');
    //Get current data and time
    $dateTime = date('Y-m-d H:i:s');

    $sql = "INSERT INTO user_search_history
            VALUES ('$signedInUser','$search','$dateTime)";
    mysqli_query($con, $sql);
} //if
mysqli_close($con);

```

Any time a signed in user performs a search, the details of that search will be recorded so that it can be used to generate recipe suggestions.

The following is an example of searching for another user on the site.

Find a User

🔍

Users can search for other users via a username, email address, first name or surname.

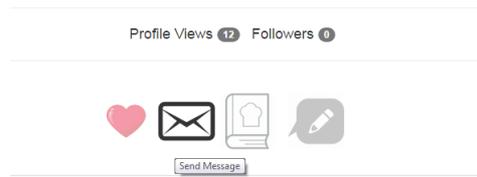
Showing results for "User"

	<p>NewUser <small>Joined on 2016-07-14</small></p> <hr/> <p style="text-align: right;"><small>Profile Views 0 Followers 0</small></p>
	<p>TestUser1 <small>Joined on 2016-06-16</small></p> <hr/> <p style="text-align: right;"><small>Profile Views 25 Followers 0</small></p>
	<p>TestUser10 <small>Joined on 2016-06-16</small></p>

The user results page differs in layout from the recipe results page

Profile Page

The profile page will provide methods for registered users to interact with each other. The small control panel underneath the user's bio gives the option of "following a user", sending them a direct message, or viewing the full list of recipes and profile posts related to that user. Each button (and most buttons on the site) makes use of CSS hover effects for responsiveness.



```

}.icons {
    height:70px; width:70px;
    border:none;
    opacity:0.4;
}
}.icons:hover {
    opacity:1.0;
}
}.activeIcons {
    height:70px; width:70px;
    border:none;
    opacity:1.0;
}
}.activeIcons:hover {
    opacity:0.4;
}

```

User interaction on the profile page. Most buttons on the site user CSS hover effects to make buttons appear brighter and more defined when the user hovers over them.

When the user clicks on the follow button, a script inserts a new record in the “User Follows” table, specifying the followed user, the follower, and the date it has been added. When the user views this profile page again the follow button will already be highlighted and the function attached to the button will have changed so that it will remove the person from the user’s followed list instead.

```

<script>
var recipient = "<?php echo $user; ?>";
var redir;

function send_pm(){

    redir = "send_message.php?=".concat(recipient);
    location.replace(redir);

}

function follow_btn(){

    redir = "php/follow_user.php?=".concat(recipient);
    location.replace(redir);

}

```

```

$url = $_SERVER["REQUEST_URI"];
$position = strpos($url, "=") + 1;
$toBeFollowed = substr($url, $position);

$user = $_SESSION["username"];

$con = new mysqli("127.0.0.1", "root", "", "chefstable");

//Set correct time zone for server
date_default_timezone_set('Europe/London');
//Get current data and time
$dateTime = date('Y-m-d H:i:s');

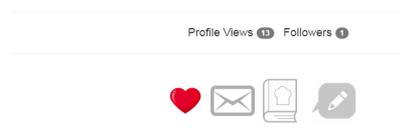
$sql = "INSERT INTO user_follows
VALUES ('$user', '$toBeFollowed', '$dateTime)";
mysqli_query($con, $sql);

$sql = "UPDATE users
SET num_follows = num_follows + 1
WHERE username = '$toBeFollowed'";
mysqli_query($con, $sql);

mysqli_close($con);
header("Location:view_profile.php?=".$toBeFollowed);

```

When the user clicks the follow button, the “Follow User” script is loaded with the name of the person to be added. A new record is added into the relevant table, and the total number of followers for that user is incremented by one.



Finally, the user is redirected back to the profile page, which has been updated to reflect that they are now following this user. The function for the button will also have changed so that it will delete the record from the “Following” table.

Clicking on the message icon will open up the new message page. This allows the sending of private messages with a subject line and text content. Sent and received messages can be viewed from a user's inbox/outbox in the account menu.

Send Message

To:  TestUser5

Hello

This is a test message!

Sending a message.

```

if(!empty($_POST["sendMessage"])){

    //Check no errors/problematic characters in subject line
    $subject = test_input($_POST["subject"]);

    //Subject line must not be empty
    if(empty($subject)){
        $errorMsg = "Please enter a subject for your message.";
    }

    //Subject line must not be too long...
    if(strlen($subject) > 150){
        $errorMsg = "Subject line must be less than 150 characters!";
    }

    //Check no errors/problematic characters in message
    $message = test_input($_POST["message"]);
    if(empty($message)){
        $errorMsg = "Message is empty!";
    }

    //Set correct time zone for server
    date_default_timezone_set('Europe/London');
    //Get current date and time
    $dateTime = date('Y-m-d H:i:s');

    //If no errors found, add new record to 'User Inbox' table
    if(empty($errorMsg)){

        //Initialise SQL connection
        $con = new mysqli("127.0.0.1", "root", "", "chefstable");

        //Escape strings to prevent SQL injection
        $subject = mysqli_real_escape_string($con, $subject);
        $message = mysqli_real_escape_string($con, $message);

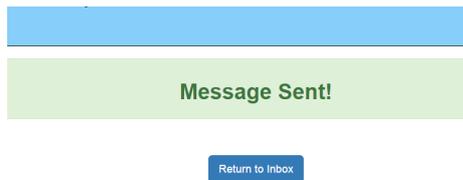
        //Save sender's username to a variable
        $activeUser = $_SESSION["username"];

        //Insert new record into table.
        $sql = "INSERT INTO user_inbox
            VALUES (default, '$recipient', '$activeUser', '$dateTime', '$subject', '$message', default, default)";
        mysqli_query($con, $sql);

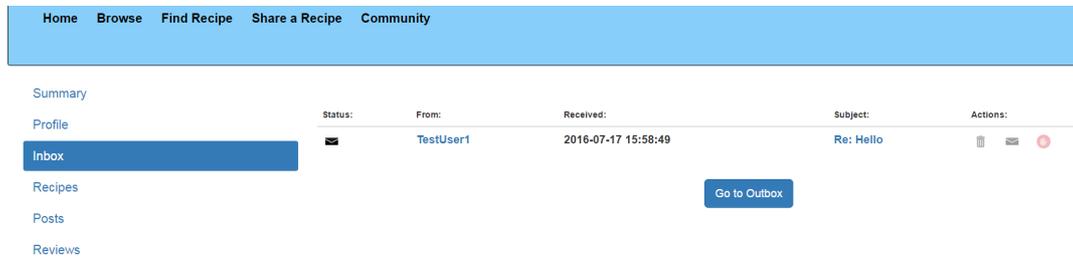
        //Close connection to database
        mysqli_close($con);
        //Redirect user to message success page
        header("Location: message_success.php");
    }
}

```

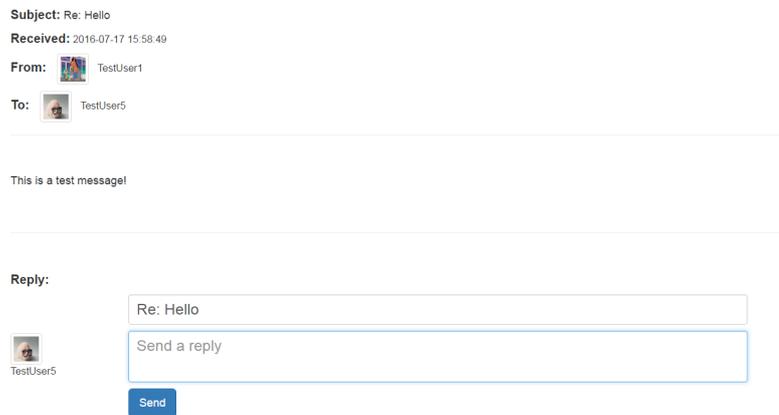
Form handling for sending a message.



Positive feedback to the user is provided when the message has been successfully sent.



The message will appear in the recipient's inbox and the sender's outbox.



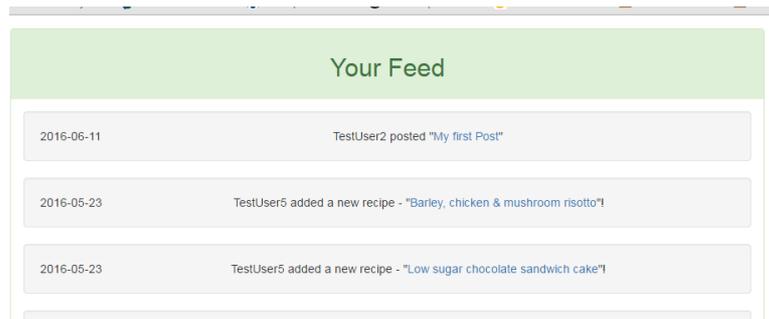
It can be viewed in the message viewer and replied to directly using the same subject line.

Personalised feeds on community page

The primary function of following a user is to facilitate a personalised feed being generated on the community page. This will display updates on followed users, sorted by the most recent. When building the feed, the PHP script will include recently uploaded recipes, profile posts, users that the followed user is also following, and reviews.



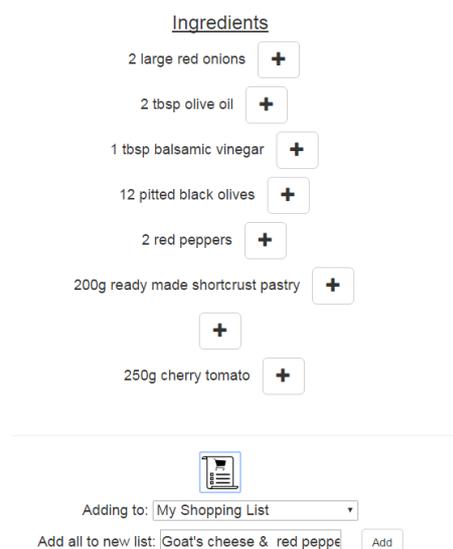
The entries in the "User Follows" table indicate that TestUser1 is following two others – TestUser2 and TestUser5.



Updates on these users will be provided in a dedicated feed on the community page.

Recipe Viewer – Extra features

As well as viewing and maintaining shopping lists of ingredients in the user's account page, the feature has been integrated into the recipe viewer to enable the ingredients for a recipe to be added to a list conveniently. This feature will only be visible to signed in users, and can be accessed by clicking on the list icon underneath the ingredients.



The shopping list feature has been integrated into the recipe viewer page.

Ingredients can be added one at a time by clicking on the relevant button. These buttons will be attached to a function which checks a hidden checkbox, and this can be used by the PHP form to determine which ingredients have been selected. A drop down menu will contain all of the user's stored lists which allows the user to specify which list they want to store the ingredient(s) to. Alternatively, they can save all of the ingredients as a new list.

Ingredients

2 large red onions

2 tbsp olive oil

1 tbsp balsamic vinegar

12 pitted black olives

2 red peppers

200g ready made shortcrust pastry

250g cherry tomato



Adding to:

Add all to new list:

Ingredients can be added individually

```
//DETERMINE WHICH LIST THE INGREDIENTS ARE TO BE ADDED TO
$listName = $_POST["allLists"];

//LOOP THROUGH ALL POTENTIAL INGREDIENT CHECKBOXES
for($index = 0; $index < 20; $index++){
    //IF THE CHECKBOX IS TICKED, ADD INGREDIENT TO ARRAY
    if(isset($_POST["ing".$index])){
        $item = $quantities[$index]." ".$ingredients[$index];
        array_push($toBeAdded, $item);
    }
}

//FOR
//ESTABLISH CONNECTION TO DB
$con = new mysqli("127.0.0.1", "root", "", "chefstable");

//COUNT THE NUMBER OF ITEMS ALREADY IN THIS LIST,
//IN ORDER TO DETERMIN AT WHAT POSITION IN THE NEW LIST THE
//NEXT ITEM SHOULD BE ADDED IN.
$sql = "SELECT COUNT(item_id)
FROM shopping_list_items
WHERE list_id = '$listName'";
$query = mysqli_query($con, $sql);
$result = mysqli_fetch_array($query);
$nextPointer = $result[0];
$nextPointer++;

//LOOP THROUGH THE ARRAY OF INGREDIENTS TO BE ADDED
foreach($toBeAdded AS $nextItem){
    //INSERT INGREDIENT AS A LIST ITEM.
    $sql = "INSERT INTO shopping_list_items
VALUES ('$listName', '$nextPointer', '$nextItem')";
    mysqli_query($con, $sql);

    //INCREMENT THE POINTER TO THE NEXT POSITION IN THE LIST
    $nextPointer++;
}

//DISPLAY A SUCCESS MESSAGE TO THE USER
$successMsg = "Items have been successfully added!";
//CLOSE DB CONNECTION
mysqli_close($con);
```

The script for adding ingredients from the recipe to a shopping list.

The user can also view the stats and reviews for a recipe using the control panel at the bottom of the page. If they are signed in, the user can also write their own review.

The screenshot displays a recipe page with the following elements:

- Recipe Stats:**
 - Overall Rating: 5 stars (3 reviews)
 - Number of Views: 96
- User Reviews:**
 - Review by TestUser1, posted on 2016-07-05 02:49:03.
 - Rating: 5 stars.
 - Review text: "This is a test review".
 - A "Write Review?" button is visible below the review.

Below the screenshot, the text "Viewing a recipe's stats" and "View a recipe's reviews." are centered.

When a user is writing a review, they will have to select a star rating and write a title for their review. If the review has been successfully added to the “User Reviews” table, it will become publically visible on the recipe’s page.

The screenshot shows a star rating interface with five stars. The fourth star is selected, and the text "Recommended" is displayed below it. To the right, the JavaScript code for the `four_star` function is shown:

```
function four_star() {
    document.getElementById("1s").checked = false;
    document.getElementById("2s").checked = false;
    document.getElementById("3s").checked = false;
    document.getElementById("4s").checked = true;
    document.getElementById("5s").checked = false;

    document.getElementById("oneStar").style.opacity = "1.0";
    document.getElementById("twoStar").style.opacity = "1.0";
    document.getElementById("threeStar").style.opacity = "1.0";
    document.getElementById("fourStar").style.opacity = "1.0";
    document.getElementById("fiveStar").style.opacity = "0.4";

    document.getElementById("ratingText").innerHTML = "Recommended";

    return false;
} //function
```

The text prompt underneath the rating will be changed via a JavaScript function depending on the rating selected.

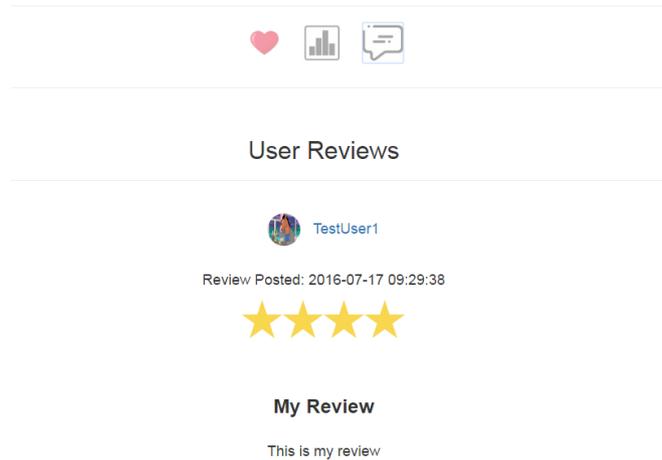
```
//CALCULATE NEW AVERAGE AND UPDATE IN DATABASE
$newRating = ($numOne + $numTwo * 2 + $numThree * 3 + $numFour * 4 + $numFive * 5) / $numReviews;

$newRating = round($newRating);

$sql = "UPDATE recipes
        SET overall_rating = '$newRating'
        WHERE recipe_id = '$id'";
    mysqli_query($con, $sql);

mysqli_close($con);
header("Location:". $url);
//:f
```

When a new review and rating is submitted, the average rating must be recalculated using a weighted average formula. The count of the total number of reviews will also be incremented and updated in the recipe’s record.



If the recipe has been successfully submitted, it will be publicly visible on the recipe's page.

Account Page – Adding posts

A user can post short blog posts to their profile page, which other users can then read and comment on. A new post can be added via the relevant button on the “Posts” tab in the account section. This will open up the new post form which asks for a post title, text content, and up to three images (optional). When a post has been published, other users can view it and comment using the form at the bottom of the post.

Title for your post

What's on your mind?

Add Image(s)?

Post
Cancel

The form for creating a new post



Users can add up to three images to their post, which can also be previewed before submitting the form.

```

LOW A USER TO CLEAR AN IMAGE BY CLICKING ON IT
function clear_image(theImage) {

//IF THE USER CLICKS ON AN UPLOADED IMAGE, THE ID FOR THE IMAGE
//WILL BE PASSED AS A PARAMETER TO THIS FUNCTION
switch(theImage) {

//CLEAR BOTH THE VALUE OF THE FILE UPLOAD FIELD, AND THE SRC
//FOR THE IMAGE PREVIEW.
case 1:
//CLEAR THE FILE UPLOAD TARGET, TO PREVENT THE UNDESIREED IMAGE BEING SAVED.
document.getElementById("image1").value = "";
//REVERT THE IMAGE PREVIEW TO THE PLACEHOLDER IMAGE
document.getElementById("previewOne").src = "images/image_icon.png";
break;
case 2:
document.getElementById("image2").value = "";
document.getElementById("previewTwo").src = "images/image_icon.png";
break;
case 3:
document.getElementById("image3").value = "";
document.getElementById("previewThree").src = "images/image_icon.png";
break;
}
}
  
```

JavaScript functions for adding and removing images to posts.

```

//A VARIABLE WHICH HOLDS NUMBER OF IMAGES THAT HAVE ALREADY BEEN UPLOADED (MAX 3)
var num = 1;

//THE FUNCTION THAT IS CALLED WHEN THE USER SELECTS FILE UPLOAD
var loadFile = function(event) {

//CONSTRUCT A FILE READER AND ASSIGN IT TO A VARIABLE
//THIS WILL BE USED TO READ THE CONTENT OF THE UPLOADED FILE
var reader = new FileReader();

//CREATE AN EVENT HANDLER WHICH IS EXECUTED WHEN THE LOAD EVENT IS FIRED
reader.onload = function() {

//UPDATE THE IMAGE PREVIEW, DEPENDING ON HOW MANY FILES HAVE BEEN UPLOADED
switch(num) {
case 1:
var output = document.getElementById("previewOne");
output.src = reader.result;
num++;
break;
case 2:
var output = document.getElementById("previewTwo");
output.src = reader.result;
num++;
break;
case 3:
var output = document.getElementById("previewThree");
output.src = reader.result;
num++;
break;
}
}
};

//STORE THE TARGET OF THE FILE UPLOAD AS A URL STRING
reader.readAsDataURL(event.target.files[0]);
}
  
```

A Test Post

This is my post.

Posted by TestUser1 on 2016-07-18



Views  Comments 

Images for this post



6. Testing and Evaluation

6.1. Testing introduction

The development for this project has been based on the AGILE method. As such, thorough testing of each component has been performed at the end of each “sprint”. The purpose of the testing section therefore is to both document these tests and ensure that no new issues have arisen with interconnected components.

6.2. Test Cases

A series of test cases outlining which feature will be tested, the desired outcome and, if an issue has been found, the steps which were taken to remedy the problem. The test cases will mirror closely the objectives and functional requirements of the system which were outlined during the analysis section.

Test	Method	Success?	Explanation
<i>Test #1 – User should be able to create an account and set their preferences.</i>	Create a new user, using the registration form, and then set their preferences.	Yes.	When the user successfully completes the new user form, a new record containing the user’s details is inserted into the users table and a record containing any preferences the user has set is inserted into the user preferences table. If the user has uploaded an avatar, the image is renamed to the user’s username and stored in the avatar directory.
<i>Test #2 – User should be able to edit their preferences and delete their account if they wish</i>	Sign in as the new test user, update their preferences, and then delete the account.	Partial.	<p>If the user confirms that they want to delete their account, their record is deleted from the users table and any records in related tables that share a foreign key with the username are also deleted.</p> <p>A problem was found in the SQL query for updating a user’s preferences. The query used was an “INSERT” statement when it should have been an “UPDATE” statement. As the username is the primary key for the preferences table, MySQL would not allow another record to be inserted with the same username. This has now been rectified.</p>

<p><i>Test #3 – A registered user should be able to publish a recipe to the site.</i></p>	<p>Create a new recipe using the recipe form.</p>	<p>Yes.</p>	<p>When the user successfully completes the new recipe form, new records are added to the recipe, ingredients and instructions table. These records are linked by the shared key in the recipe ID field. The user will be redirected to another page which informs them that the recipe has been successfully published to the site and are given the option of viewing it, editing it or creating another new recipe.</p>
<p><i>Test #4 – The user should then be able to edit or delete their recipe.</i></p>	<p>Change certain details of the recipe, and save these changes.</p>	<p>Partial.</p>	<p>When the user navigates to their recipes page, they can make changes to their recipe or delete it entirely. When editing a recipe, the same form for creating a recipe is used and the current values are loaded and displayed inside the input boxes. Submitting the form then sends SQL “UPDATE” statements to the relevant tables. On testing, the only issue found was with the instructions section. If the user removed an instruction using the associated button, the changes were not saved. This was because the JavaScript function for the button would only hide the input, without also clearing the value from the textbox. This has now been rectified.</p> <p>When the user wants to delete the recipe, a confirmation dialog is presented. Once the user confirms the operation, the record for the recipe is deleted and any linked records in other relevant tables are also deleted (e.g. ingredients, instructions, user reviews, user view history etc).</p>
<p><i>Test #5 – The recommendations page should provide a variety of accurate and relevant recipe suggestions.</i></p>	<p>Launch the recommendations page, as a guest user. Refresh page a number of times.</p>	<p>Yes.</p>	<p>When the user first accesses the site, they will be presented with a series of recommendations, the order of which will change every time. Clicking on any of the recipes inside each panel will bring the user to the page for that recipe. The recipes returned under each heading are relevant to the description.</p>

<i>Test #6 – The recommendations page should adapt to users if they are signed in, and present some more personalised suggestions.</i>	Launch the recommendations page as an active user. Refresh page a number of times.	Yes.	The user will be redirected to a variation on the regular index page if they are signed in. This page will use a different PHP script to provide recommendations based on the user's search history, viewing history, preferences etc. It will also not suggest recipes that the user has uploaded.
<i>Test #7 – The general search function should return relevant results.</i>	Perform a general search and check the relevancy of the results obtained.	Yes.	The general search will try to match recipes by matching a recipe's title, description, ingredients, or keywords to the user's search query, via MySQL's full-text matching. If a user searches for "Curry" for example, the top result will be a recipe titled "Prawn laksa curry bowl".
<i>Test #8 – The ingredient search function should return relevant results.</i>	Perform an ingredient search and check the relevancy of the results obtained.	Yes.	The ingredient search takes an ingredient description and a quantity (optional). It then attempts to match these with any records in the recipe ingredients table using MySQL's full-text matching. A search query for "4 Spring Onions" returns the top result of "California quinoa & avocado salad", which contains exactly this ingredient.
<i>Test #9 – The user search function should return relevant results.</i>	Perform a user search and check the relevancy of results obtained.	Yes.	Searching for a user by their username, email address etc. will attempt to match any records using a full text search, but will not return users that have set their privacy settings to "Private". A search for "TestUser" will return the profiles of TestUser1, 3, 4 and 5, but not TestUser2 as this user is private.
<i>Test #10 – The date search function should return relevant results.</i>	Perform a date search and check the relevancy of results obtained.	Yes.	A search for all recipes uploaded within the last 90 days for example will return only recipes uploaded during that period, sorted by the most recent first.
<i>Test #11 – The rating search function should return relevant results.</i>	Perform a rating search and check the relevancy of results obtained.	Yes.	A search for all recipes uploaded with a rating of 3 stars or above will return a list of all recipes, as at the moment each recipe has at least 3 stars. If the user searches for only those with a 5-star rating however, only 5 results will be returned representing those recipes which have 5 star ratings.
<i>Test #12 – The personalised feed on the community page should provide updates on any</i>	Using the test user, add two other users to their followed list, and then check the	Yes.	If, for example, "TestUser1" is following "TestUser2", "TestUser3" and "TestUser4", the feed on the community page will display updates on any activities those users have performed on

<i>followed users, ordered by date.</i>	feed on the community page.		the site. These feed items are sorted by date.
<i>Test #13 – The user should be able to change their email/password.</i>	Go to account details section, change the email and password, and check that the record has been updated in the users table.	Yes.	The user must supply their current password to perform either of these actions. If the current password is incorrect, the new email address is the incorrect format or if the confirm password field does not match the new password field then the form will not be able to submit and a suitable error message will be displayed to the user. Otherwise, an SQL "UPDATE" statement will be sent to the users table for either the email or password values.
<i>Test #14 – The user should be able to send, receive, view and delete messages.</i>	Send a message to another user, login as the recipient and check that the message has been received, and then delete the message.	Yes.	If, for example, "TestUser1" sends a message to "TestUser3" using the "New Message" form, a new record is inserted into the inbox table. The message is then viewable in the sent messages for "TestUser1" and the inbox for "TestUser3". When TestUser3 next logs on they can view the message and send a reply, or delete the message and block the user.
<i>Test #15 – The user should be able to post updates to their profile page.</i>	Create a new post, check that the post is publicly visible and that a new record has been added to the posts table.	Yes.	When the user completes the "New Post" form successfully, a new record is inserted into the "User Posts" table. The post then becomes publicly visible to other users via the profile page or community page.
<i>Test #16 – The user should be able to review recipes, and delete the reviews later if they wish.</i>	Write a review for a recipe, check that it displays correctly on the recipe page, and then delete the review.	Yes.	If "TestUser1" adds a review for the recipe "Zingy chicken stir-fry", a new record is inserted into the user reviews table, the number of reviews for that recipe is incremented, and the overall rating is recalculated using a weighted average of all reviews so far. The review then becomes visible to any users viewing that recipe. If the user navigates to the reviews section of their account page, a new entry will have been added with a summary of what they have written in their review. If they click on the delete button and confirm that they want to delete the review, a "DELETE" query is

			sent to the reviews table to remove that record.
<i>Test #17 – The user should be able to create, view and manage shopping lists of ingredients.</i>	Create a new shopping list, add a series of ingredients, delete a few ingredients, and then delete the list entirely.	Yes.	A default shopping list is created for every user when they first register. The lists can be accessed via the account page and can be opened or deleted. When a shopping list is deleted, the list record and all related records in the list items table are removed. The user can add, edit and remove items from the list quickly and easily.
<i>Test #18 – The shopping list feature should be integrated into the recipe page so that one, several, or all recipes from an ingredient can be easily added to a chosen shopping list.</i>	Add a number of ingredients from a recipe to a saved shopping list. Then, add all ingredients from a recipe as an entirely new list.	Yes.	When the user navigates to a recipe such as “Caramel Chicken”, they can select several ingredients and add them to the selected list. Alternatively, a user can save all ingredients from a recipe (e.g. “Thai Chicken Saute”) as a new list with the name of that recipe.
<i>Test #19 – The user should be able to post comments on other’s profiles or posts.</i>	Write a comment on another user’s profile page or a post they have published. Check that a new comment record has been created and that it displays correctly on the profile/post page.	Partial.	When the user successfully submits a comment on the forms, a new comment record is inserted into either the profile or post comment tables. The comment then becomes publicly visible on the profile/post page. There was a minor issue with the formatting and layout of comments when they have been submitted, with some text misaligned and linked content being unclear. This has been remedied via the CSS script for these pages.

6.3. Evaluation

When considering the finished application in the context of the specified system requirements in the analysis section, Chef’s Table is successful overall. It is a web application which has been designed using modern development tools and design principles. It has attempted to combine best practices from similar recipe websites, social media platforms and user-driven content providers in order to deliver a personalised, community driven resource for food and cooking enthusiasts.

As described in the analysis section, a series of minimum functional and non-functional requirements were identified and have now been implemented into the end system. Users can discover new recipes via the personalised and randomised recommendations, and find recipes specific to their needs via the search or browsing tools. Recipes hosted on the site are presented in a simple, visually appealing manner with each prominent piece of information clearly defined. Users can upload their own creations quickly and easily using the relevant form, and are prevented from entering problematic input using adequate form control and error handling. The social and user account features are integrated into the application via a user's public profile, which allows them to interact in a variety of ways with other users and content, as well as convenient features such as adding ingredients from a recipe to a saved shopping list.

During development, efforts were made to ensure that Chef's Table would be accessible to all potential users, e.g. by making the HTML code screen reader friendly and avoiding the overuse of colours and images to provide user feedback. The use of AGILE "sprints" has contributed to the overall quality and stability of the application, as features have been added incrementally and fully tested at each stage. An additional period of testing and refinement was also undertaken during the final phase of development. The "Bootstrap" mobile framework has provided a good level of compatibility for Chef's Table across devices and screen size, with content adjusting and rearranging appropriately via the "Grid" system. Finally, each stage of development has been fully documented and illustrated with diagrams and code explanations.

7. Conclusions and Recommendations

7.1. Concluding thoughts

The original problem that Chef's Table was designed to solve can be summarised from the analysis section as:

“To provide an online recipe resource which caters to the requirements of a variety of users and can be accessed via desktops or mobile devices. The user should be able to find and discover recipes relevant to their tastes and preferences, publish their own creations, review those submitted by other users, store favourites and connect socially with those who share a similar love of food and cooking.”

The extent to which the finished system achieves the aims set out in the problem statement can be assessed critically under the following headings:

1. Accessibility.

Chef's Table is intended to appeal to a broad spectrum of potential users, who will have differing levels of knowledge about food and cooking, as well as the ease and familiarity they might have with a typical software system.

As such it was important that the final user interface, forms and interactions were intuitive, simple and streamlined so as to avoid alienating any less experienced or more casual users. To this end, each potential user interaction was identified using storyboards at the analysis stage, carefully prototyped and iterated upon during the design phase, and finally implemented and tested thoroughly for any potential issues which might cause confusion or irritation for the end users. Practical steps taken included appropriate error handling and error messages on any forms which required user input, clearly distinguishing different elements using borders, boxes, colours etc., and providing immediate feedback when a user successfully or unsuccessfully completes an action.

The needs of users with visual impairments were also considered. The HTML code was formatted and spaced for maximum compatibility with screen readers, and any images were provided with informative alternative text descriptions.

2. Responsiveness

It is important that any modern web application is capable of responding and adapting to a range of different screen sizes, devices and operating systems, and to do so without negatively affecting the experience for a particular group of users. To this end, Chef's Table made use of the Bootstrap mobile development framework to ensure that

content would resize and scale appropriately and maintain a standardised layout across devices. This was achieved by wrapping all page content inside Bootstrap's grid-based system which sets parameters for which items should be most prominently presented and allows the navigation bar to minimise into a drop down menu on mobile devices. Use was also made of several Bootstrap styles and icons, which contributed to a sense of consistency and quick familiarisation for users of the site.

3. Discoverability

Chef's Table was intended to be a resource for people to both find a particular type of recipe and to discover new favourites. This is delivered via the randomised and personalised recommendations on the home page, which draws from all of the recipes on the site to provide a variety of potential choices for the user. The search function allows users to enter customised, specific search queries which will be matched with relevant recipes. The community page will also keep users informed of all the recent activity on the site, which also acts as a content discovery tool.

4. User content and customisation

User content is a crucial element of the application, and as such it is important that users are encouraged to engage with the content and feel comfortable with adding their own. Users are encouraged to create an account on the site so that they can personalise their experience and access some of the more advanced features. The user can then update any of their details or access their published content via their account settings.

Recipes can be easily added to the site using the relevant form, which clearly outlines all of the essential details which a user should be including for their recipe and mimics the layout of the recipe viewer for consistency.

5. Social and community

To encourage recurring use amongst visitors to the site, efforts have been made to encourage a sense of community and friendly collaboration. Users can communicate directly by sending direct messages, or indirectly by writing posts on their profile pages or leaving comments on other user's profiles/posts. Feedback can also be obtained on shared recipes via the user reviews and associated statistics such as number of visits and average user score.

7.2. Recommendations for further development

The following have been identified as potential revisions or additional features for Chef's Table:

- A feature which allows users to create short video clips and share them on the site. These videos could serve as practical guides to cooking a recipe, and would be linked to on the recipe's page.
- A bespoke app which would be specifically optimised for mobile devices and would be separate from the basic web application.
- The search function would be further developed to allow more criteria and filters to be applied to potential results.
- The ability to create a photo gallery of different images for a created recipe.
- The ability for a user to change the privacy settings for their profile and content has been included but not developed fully due to a lack of time.

7.3. Evaluation of development process

In addition to delivering a final software solution, a desired outcome of the development process was for the developer to display a familiarity with modern software design principles and to successfully manage the delivery of a long-term, work-intensive project. The process of building the system involved the following distinct phases:

1. *Identified purpose, scope and desired outcome of project. Also considered the most suitable tools for achieving these outcomes.*
2. *Undertook any required training, familiarisation and resource gathering.*
3. *Thoroughly documented the Analysis process for the system and decided upon a suitable design process*
4. *Delivered first milestone report.*
5. *Designed, implemented and tested new features whilst adhering to the AGILE methodology.*
6. *Achieved a working prototype which included the most essential components and provided a realistic impression of the quality of the final system.*
7. *Delivered second milestone report.*
8. *Tested overall system thoroughly and refined any components as necessary.*
9. *Completed documentation.*
10. *Delivered final system.*

The project could be considered a success from a management and organisational perspective due to a combination of careful planning and task prioritisation. The AGILE method was a very useful tool as it encouraged short bursts of highly focused development on a single feature, and this resulted in a higher level of quality as each component was tested throughout and issues could be caught and rectified at an early stage.

Key milestone targets such as the first and second progress reports, as well as the final system were delivered on time and were considered to be at a good standard of quality.

A high degree of competence and familiarity with the chosen development tools has also been achieved as a result of this project. The developer has made full use of the new elements introduced in HTML5 such as the updated semantic and structural tags, form elements, and input types. CSS3, the most up to date version of Cascading Style Sheets was used to format and style each page. A MySQL database was used to store all necessary user data, and efforts were made to reduce data redundancy and enforce relational integrity between linked entities. PHP scripts were used to execute relatively complex queries which manipulated and displayed data from the database as efficiently as possible. Finally, an understanding of the importance of responsive design and catering for mobile devices was illustrated by the use of several Bootstrap elements such as grid layouts, wells, alerts, panels and image classes. Additional user interaction and feedback was provided by JavaScript and JQuery functions.

7.4. Reflections and lessons learned

When reflecting on the development process for Chef's Table, the developer has learned the following lessons which would be key considerations for any future projects.

- The AGILE approach is a highly effective tool for rapidly developing and iterating upon a new system, however it may not be the most suitable choice for all types of projects and development teams. For this project, the bulk of development was performed during the first few months. This had a negative impact on development as when the developer is focusing intently on a single feature it is possible to become mentally exhausted and therefore lose motivation. It is important therefore that sprints and milestones are carefully planned so that a sustainable pace can be maintained.
- If the system will be available on smartphones and tablets, it is important that the system is designed from the ground up as a mobile application, and then expanded

to include more powerful devices. This “mobile first” mind-set will lead to a higher overall quality, as there is a minimum benchmark identified early on in development.

- When considering the approach to delivering a software solution, the focus should be placed on developing a small number of features to a high level of quality, rather than a large number of features which will not benefit from an appropriate focus of attention and development time. The smallest possible system that would deliver on the functional requirements should be the target of early prototyping, and the system can then be expanded appropriately depending on the remaining time and other resources.
- It is important for the developer to make full use of all resources and support available to them during the project. In this case, valuable feedback was obtained from each milestone report and this was used to help steer the direction of the project and identify key priorities. However more advice and guidance could have been sought from faculty and staff at the University which might have more quickly resolved troublesome issues and avoided time wasted on ill-considered ideas.

References

- Agile Methodology (2008). *The Agile Movement* [Online] agilemethodology.org. Available at: <http://agilemethodology.org/> [Accessed 12th July 2016].
- Alexa (1996-2016). *Top Sites in: All Categories > Home > Cooking > Recipe Collections*. [Online]. Available at: http://www.alexa.com/topsites/category/Top/Home/Cooking/Recipe_Collections [Accessed 13th January, 2016].
- *Allrecipes.co.uk*. Available at: <http://allrecipes.co.uk/> [Accessed 13th January, 2016].
- Amherst College (n.d.) *Make your Webpages Accessible to the Visually Impaired* [Online] Available at: <https://www.amherst.edu/mm/48284> [Accessed 13th July 2016].
- Barker, S (2015). *Actionable Ways To Improve Website Usability* [Online] Business 2 Community. Available at: <http://www.business2community.com/seo/actionable-ways-improve-website-usability-01309308#FrkPoihsRLUUmH8.97> [Accessed 13th July, 2016].
- *BBC Food*. Available at: <http://www.bbc.co.uk/food/> [Accessed 13th January, 2016].
- BBC News (2010). *Jamie Oliver's 30-Minute Meals breaks sales record*. [Online] BBC News. Available at: <http://www.bbc.co.uk/news/uk-11957627> [Accessed 13th January, 2016].
- Beal, V (2016). *Normalization*. [Online] Webopedia. Available at: <http://www.webopedia.com/TERM/N/normalization.html> [Accessed 17th July, 2016].
- Bittman, M (2014). *Rethinking the Word 'Foodie'*. [Online] The New York Times. Available at: http://www.nytimes.com/2014/06/25/opinion/mark-bittman-rethinking-the-word-foodie.html?_r=1 [Accessed 13th January, 2016].
- Boehm, B (1989). *Software Risk Management*. Washington, DC, USA: IEEE Computer Society (CS) Press.
- Bowes, J (2014). *Agile vs Waterfall: Comparing project management methods*. [Online]. Manifesto.co.uk. Available at: <https://manifesto.co.uk/agile-vs-waterfall-comparing-project-management-methodologies/> [Accessed 12th July 2016].

- Chapple, M (2007) *Database Normalization Basics*. [Online] About Tech. Available at: <http://databases.about.com/od/specificproducts/a/normalization.htm> [Accessed 13th January, 2016].
- *Cooks.com*. Available at: <http://www.cooks.com/> [Accessed 13th January, 2016].
- Database Publishing Consultants (n.d.). *HTML5 Features and Benefits*. [Online] DPCI. Available at: <http://www.dpci.com/html5-features-and-benefits> [Accessed on 13th January, 2016].
- European Commission (2016). "Web Usability" [Online] Available at: http://ec.europa.eu/ipg/design/usability/index_en.htm [Accessed on 12th July, 2016].
- Fitzpatrick, D (n.d.). "Importance of HCI" [Online] School of Computing, Dublin City University. Available at: <http://www.computing.dcu.ie/~dfitzpat/node/2> [Accessed on 12th July, 2016].
- *GoDaddy.com*. Available at: <https://uk.godaddy.com/> [Accessed 13th January, 2016].
- Haunts, S (2014). *Advantages and Disadvantages of Agile Software Development* [Online]. Available at: <https://stephenhaunts.com/2014/12/19/advantages-and-disadvantages-of-agile-software-development/> [Accessed on 12th July 2016].
- Hock-Chuan, C (2010). *A Quick-Start Tutorial on Relational Database Design* [Online] Nanyang Technological University, Singapore. Available at: https://www.ntu.edu.sg/home/ehchua/programming/sql/Relational_Database_Design.html [Accessed 17th July, 2016].
- *Idov, M (2012). When Did Young People Start Spending 25% of Their Paychecks on Pickled Lamb's Tongues?* [Online] New York Magazine. Available at: <http://nymag.com/restaurants/features/foodies-2012-4/> [Accessed on 12th July, 2016]
- INCORE Digital Agency (2013). *HTML5 popularity Among Fortune 500 Companies*. [Online] Available at: <https://www.incore.com/Fortune500HTML5/> [Accessed on 13th January 2016].
- Koetsier, J (2013). *HTML5 vs native vs hybrid mobile apps: 3,500 developers say all three, please*. [Online] VentureBeat. Available at: <http://venturebeat.com/2013/11/20/html5-vs-native-vs-hybrid-mobile-apps-3500-developers-say-all-three-please/> [Accessed on 13th January, 2016].

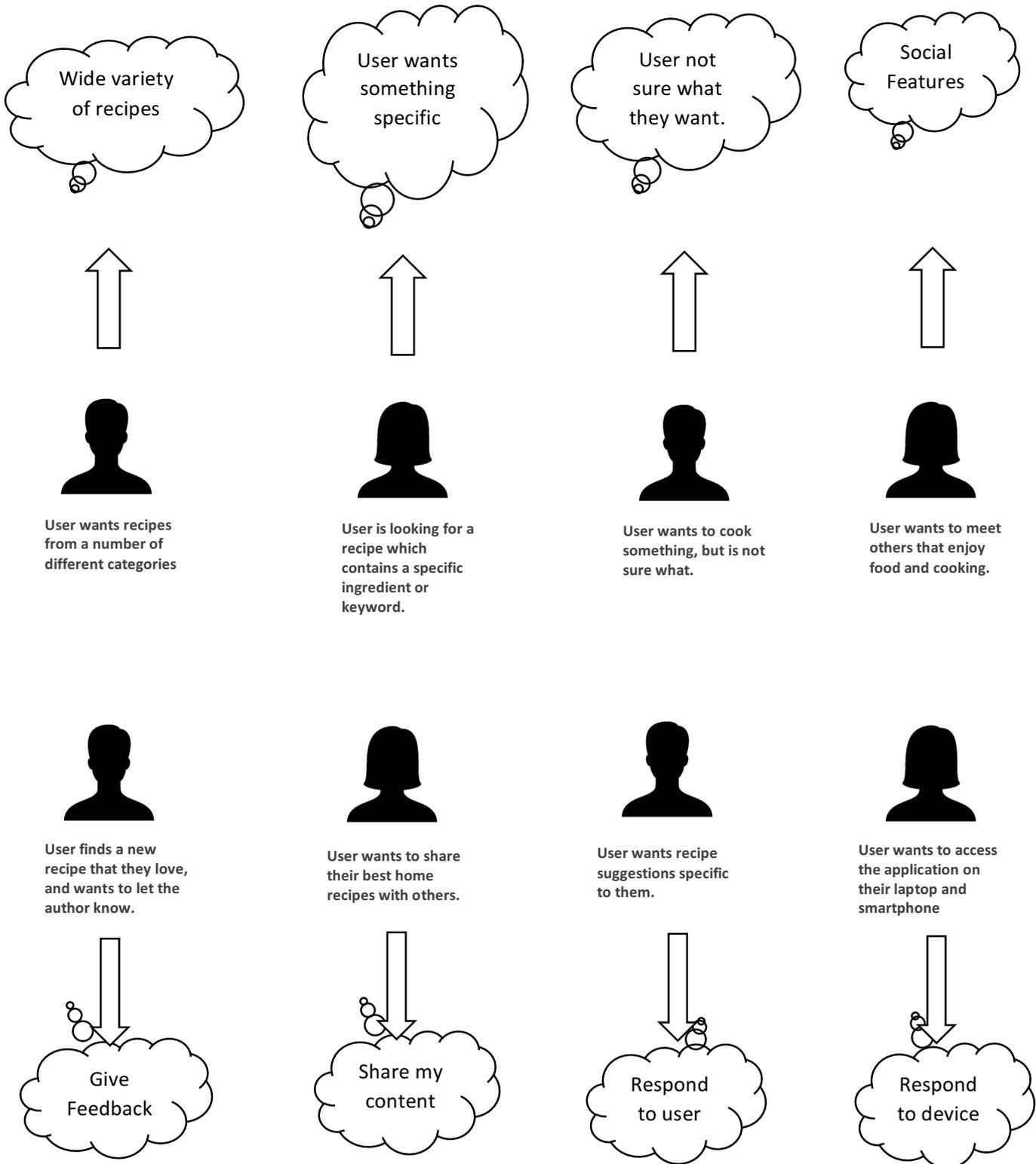
- Mah, P (2010). *Review: Notepad++ text editor*. [Online] TechRepublic. Available at: <http://www.techrepublic.com/blog/product-spotlight/review-notepad-plus-plus-text-editor/> [Accessed on 12th July 2016].
- Merriam Webster Online Dictionary (n.d) *Definition of Toque*. [Online] Available at: <http://www.merriam-webster.com/dictionary/toque> [Accessed on 16th January, 2016].
- Microsoft Developer Network (2016), "Full-Text Search" [Online]. Microsoft. Available at: <https://msdn.microsoft.com/en-us/library/ms142571.aspx> [Accessed on 14th July, 2016].
- Mikoluk, K (2013). *Agile vs Waterfall: Evaluating the Pros and Cons*. [Online] Udemy Blog. Available at: <https://blog.udemy.com/agile-vs-waterfall/> [Accessed 12th July 2016].
- MySQLTutorial (2016). *MySQL Natural Language Search* [Online]. Available at: <http://www.mysqltutorial.org/mysql-natural-language-search.aspx> [Accessed on 14th July, 2016].
- Ofcom (2015). *UK now a smartphone society*. [Online] Ofcom. Available at: <http://consumers.ofcom.org.uk/news/uk-now-a-smartphone-society/> [Accessed on 13th January, 2016].
- Orr, G (2012). *Sweet taste of sales success: Why are cookbooks selling better than ever?* [Online] Independent. Available at: <http://www.independent.co.uk/life-style/food-and-drink/features/sweet-taste-of-sales-success-why-are-cookbooks-selling-better-than-ever-8113937.html> [Accessed 13th January, 2016].
- Plunkett, J (2013). *MasterChef nets more than 4 million viewers but is no match for Messi*. [Online] The Guardian. Available at: <http://www.theguardian.com/media/2013/mar/13/masterchef-4-million-viewers-barcelona-messi> [Accessed 13th January, 2016].
- Poolet, M (1999). *SQL by Design: How to Choose a Primary Key*. [Online] Available at: <http://sqlmag.com/database-administration/sql-design-how-choose-primary-key> [Accessed 17th July, 2016].
- Radigan, D (n.d). *Epics, stories, versions and sprints*. [Online] Available at: <https://www.atlassian.com/agile/delivery-vehicles/> [Accessed 13th January, 2016].

- Scottish Qualifications Authority (2007). *Composite Key*. [Online] Available at: http://www.sqa.org.uk/e-learning/MDBS01CD/page_19.htm [Accessed 17th July, 2016].
- Scottish Qualifications Authority (2008). *Second Normal Form*. [Online] Available at: http://www.sqa.org.uk/e-learning/SoftDevRDS02CD/page_15.htm [Accessed 17th July, 2016].
- Scottish Qualifications Authority (2007). *Systems Development: Structured – Functional and Non-Functional Requirements*. [Online] Available at: http://www.sqa.org.uk/e-learning/SDM03CD/page_02.htm [Accessed 13th January, 2016].
- Sherman, A (N.D). *User Stories and Agile Development* [Online]. StoryboardTHAT. Available at: <http://www.storyboardthat.com/articles/software-development/agile-user-stories> [Accessed on 15th July, 2016].
- Software Engineering Institute (2012). *A Framework for Software Product Line Practice, Version 5.0*. [Online] Carnegie Mellon. Available at: http://www.sei.cmu.edu/productlines/frame_report/technicalRM.htm [Accessed 16th January, 2016].
- Stack Overflow (2016). *About* [Online]. Available at: <http://stackoverflow.com/company/about> [Accessed 17th July, 2016].
- TechNet (n.d). *Databases – Designing Databases – Data Integrity*. [Online] Microsoft Available at: [https://technet.microsoft.com/en-us/library/ms184276\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms184276(v=sql.105).aspx) [Accessed 13th January, 2016].
- Techopedia (2016). *Data Integrity*. [Online] Available at: <https://www.techopedia.com/definition/811/data-integrity-databases> [Accessed 13th January, 2016].
- Techopedia (2016). *Data Redundancy*. [Online] Available at: <https://www.techopedia.com/definition/18707/data-redundancy> [Accessed 13th January, 2016].
- Techopedia (2016). *Scope Creep*. [Online] Available at: <https://www.techopedia.com/definition/24779/scope-creep> [Accessed 12th July, 2016].

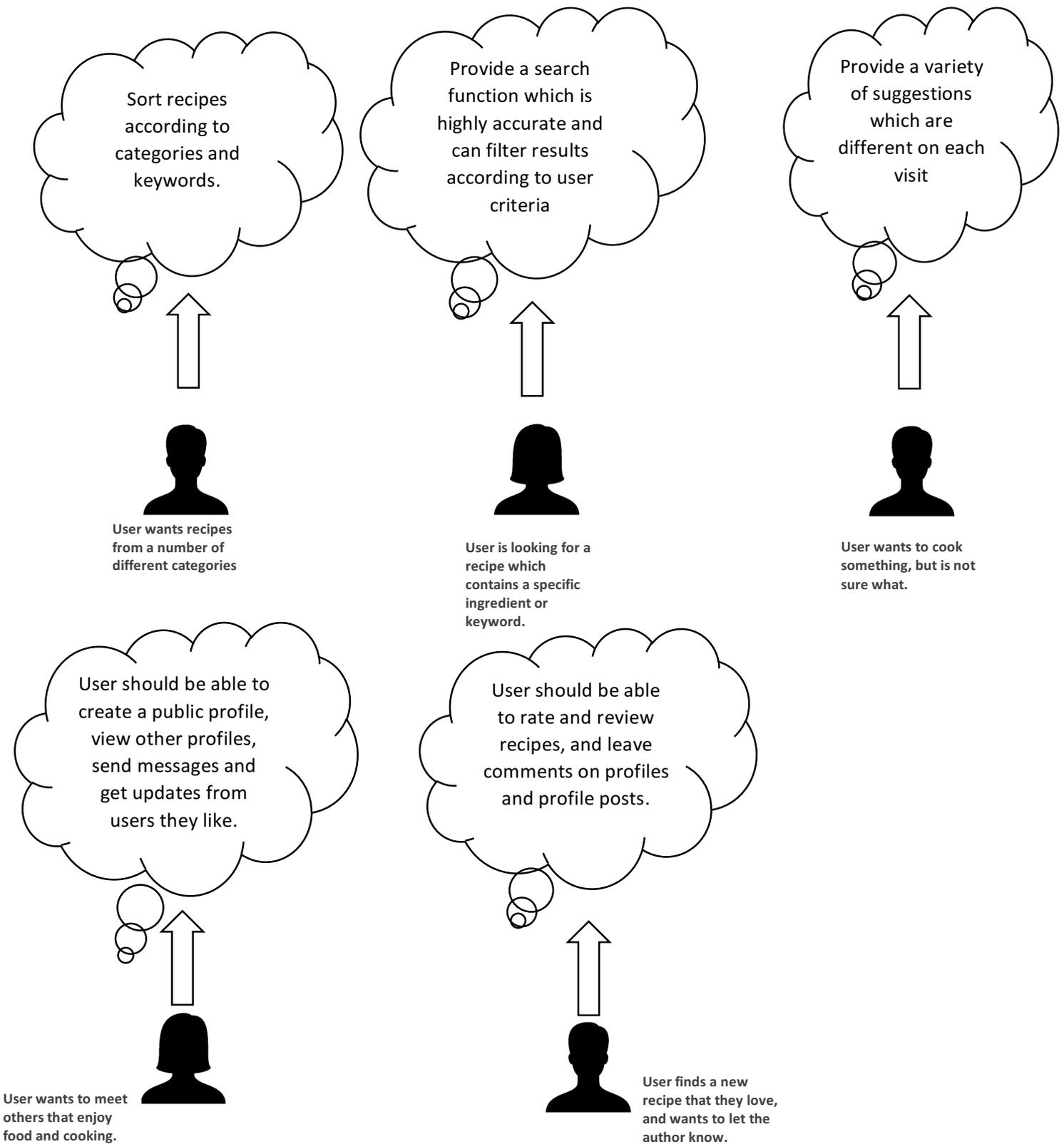
- Tee, J (2012). *Web application framework and java middleware: Pros and cons of HTML5 for business*. [Online] TheServerSide.com. Available at: <http://www.theserverside.com/tip/Pros-and-cons-of-HTML5-for-business> [Accessed on 13th January, 2016].
- Trivuncic, S (2009-2016). *Maison Cupcake*. [Online] Available at: <http://maisoncupcake.com/> [Accessed 13th January, 2016].
- Tutorialspoint (No author given) (2016). *Database – First Normal Form (1NF)*. [Online]. Available at: <http://www.tutorialspoint.com/sql/first-normal-form.htm> [Accessed 17th July, 2016].
- University of Alabama, Birmingham (2014). *The future of mobile application development (Infographics)*. [Online] Available at: <http://businessdegrees.uab.edu/resources/infographic/the-future-of-mobile-application/> [Accessed 13th January, 2016].
- W3 Schools (2016). *About W3Schools* [Online]. Available at: <http://www.w3schools.com/about/default.asp> [Accessed 17th July, 2016].
- W3 Schools (2016). *Bootstrap Grid System* [Online] Available at: http://www.w3schools.com/bootstrap/bootstrap_grid_system.asp [Accessed 13th July, 2016].
- WebAIM (2013) *Color-blindness*. [Online] Available at: <http://webaim.org/articles/visual/colorblind> [Accessed 13th July 2016].
- World Wide Web Consortium (2010). "Accessibility, Usability, and Inclusion: Related Aspects of a Web for All". [Online] Available at: <https://www.w3.org/WAI/intro/usable> [Accessed 12th July 2016].
- Wigmore, Ivy (2014). *How to write a business case*. [Online] Available at: <http://whatis.techtarget.com/reference/How-to-write-a-business-case> [Accessed 13th January, 2016].
- Woodward, E (2016). *Deliciously Ella*. [Online] Available at: <http://deliciouslyella.com/> [Accessed 13th January, 2016].
- Zara, C (2015). *Broadcast And Cable TV Ratings Keep Declining: Viacom, A&E Struggles Underscore Industry In Flux*. [Online] International Business Times.

Available at: <http://www.ibtimes.com/broadcast-cable-tv-ratings-keep-declining-viacom-ae-struggles-underscore-industry-1969568> [Accessed 13th January, 2016].

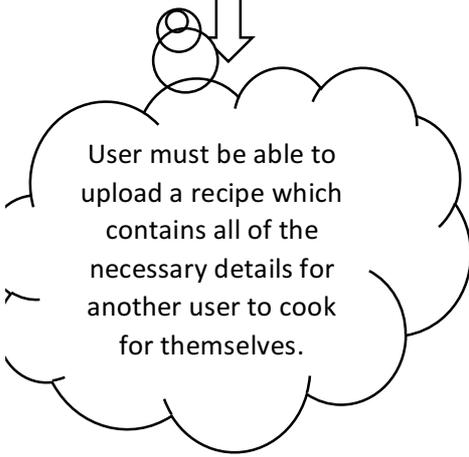
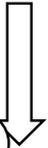
Appendix 1 - Problem Rich Picture



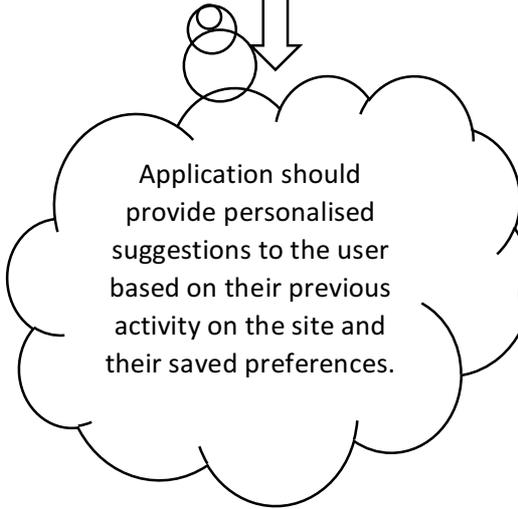
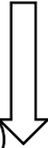
Appendix 2 - Solution Rich Picture



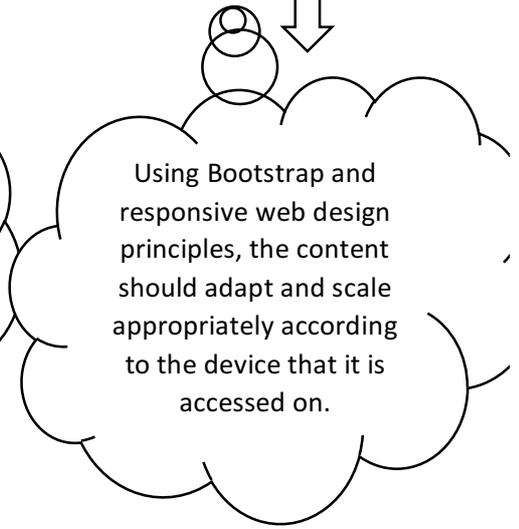
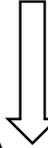
User wants to share their best home recipes with others.



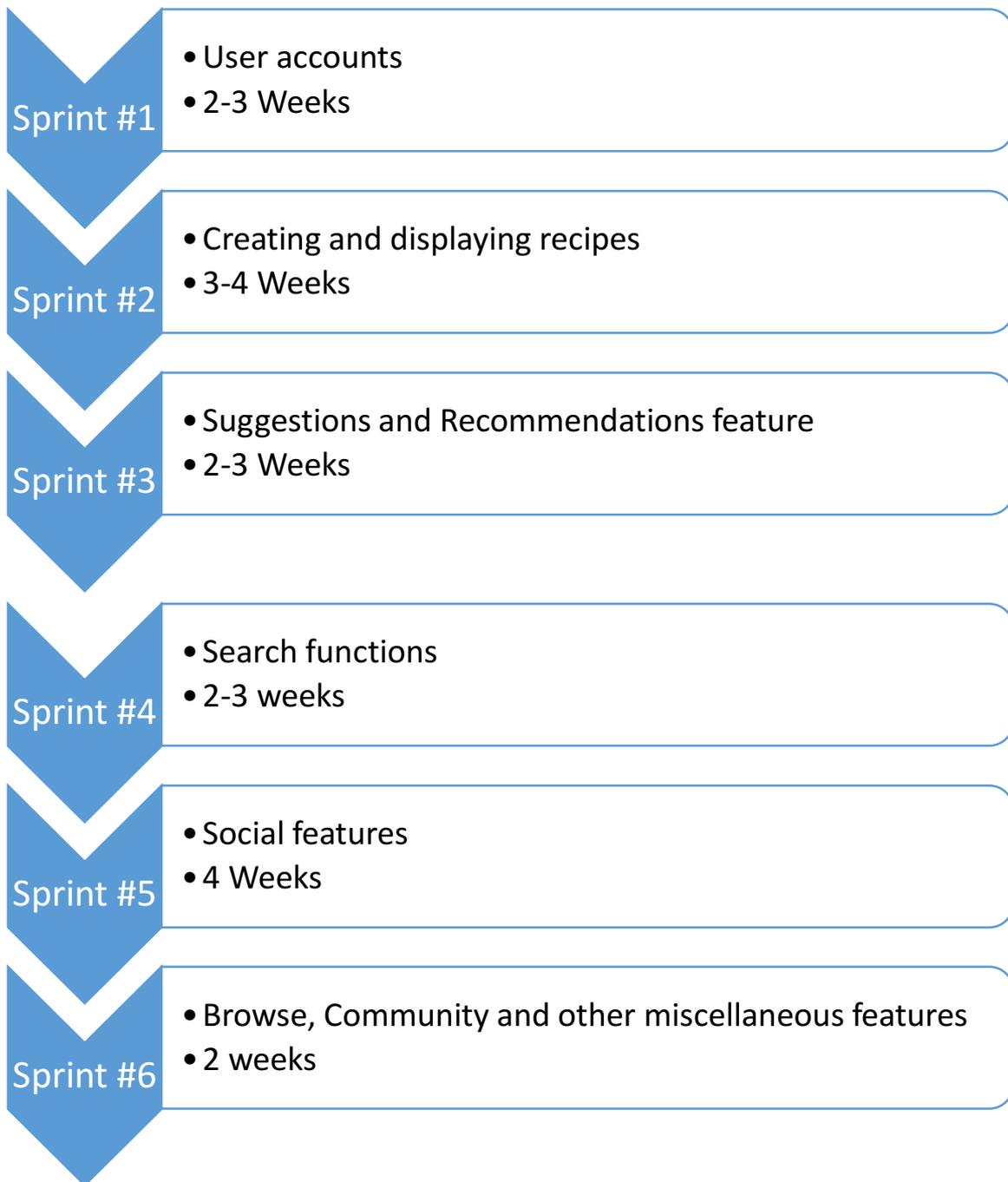
User wants recipe suggestions specific to them.



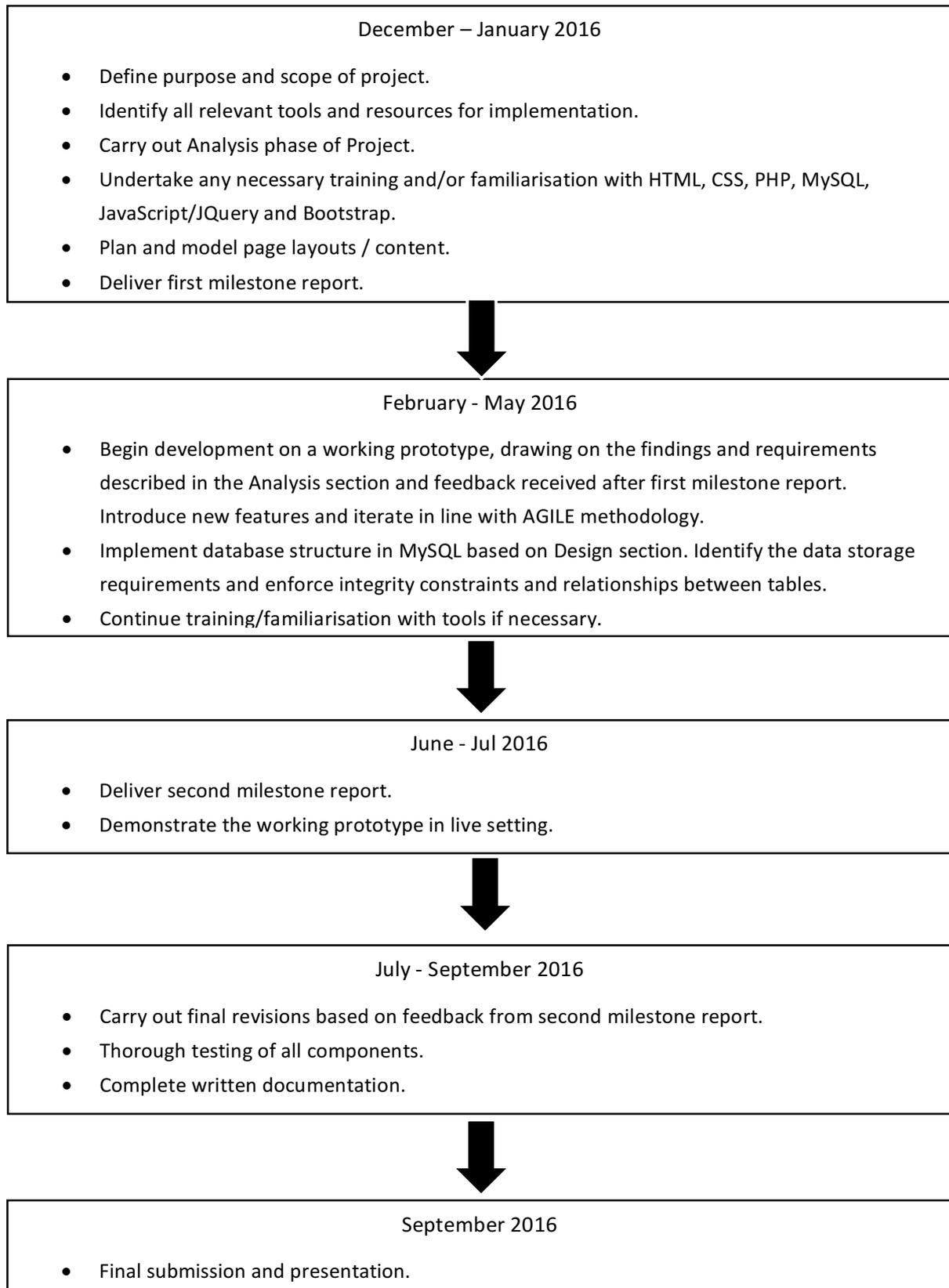
User wants to access the application on their laptop and smartphone



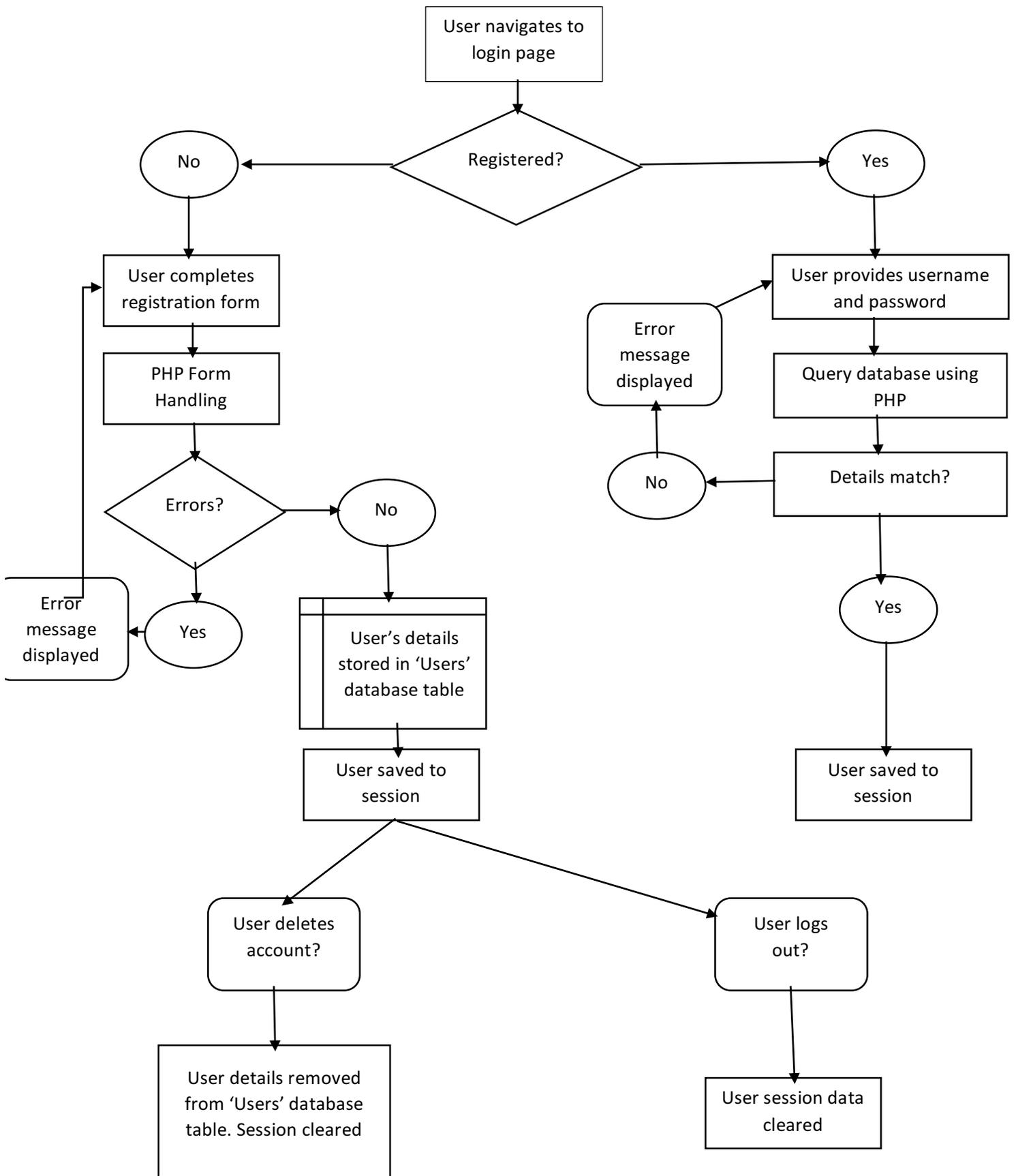
Appendix 3 – Agile Sprints



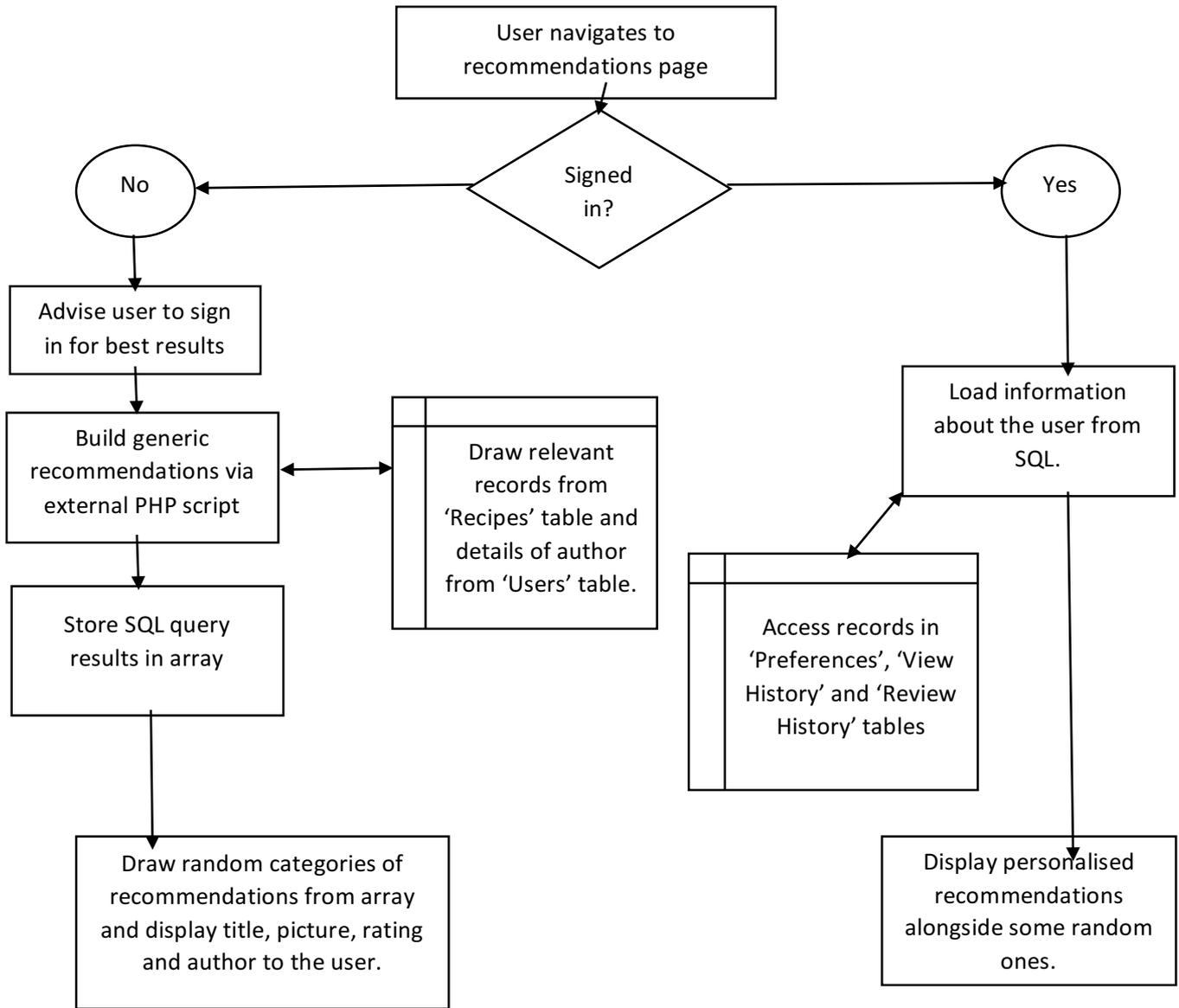
Appendix 4 – Development Schedule



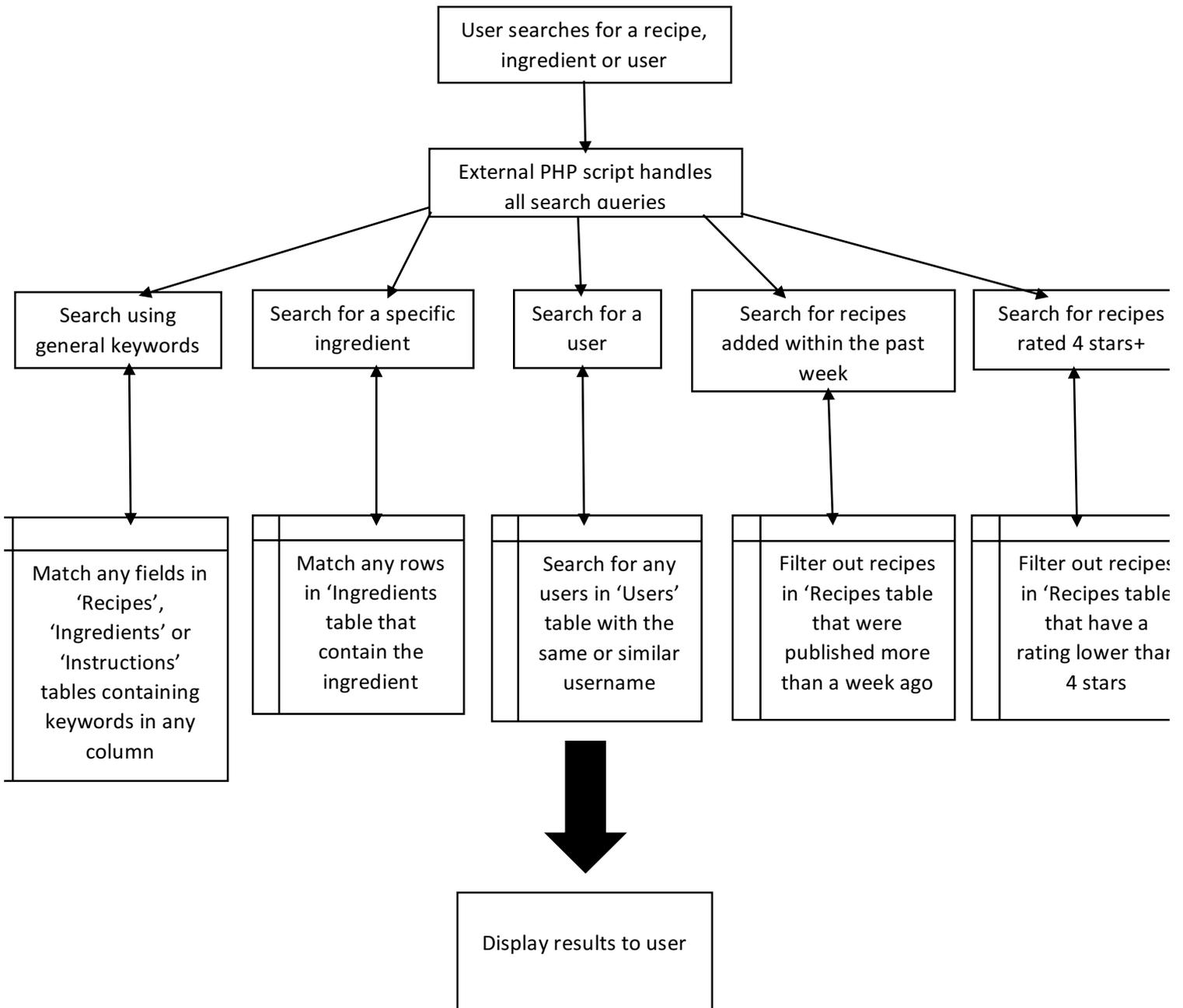
Appendix 5 – Storyboards



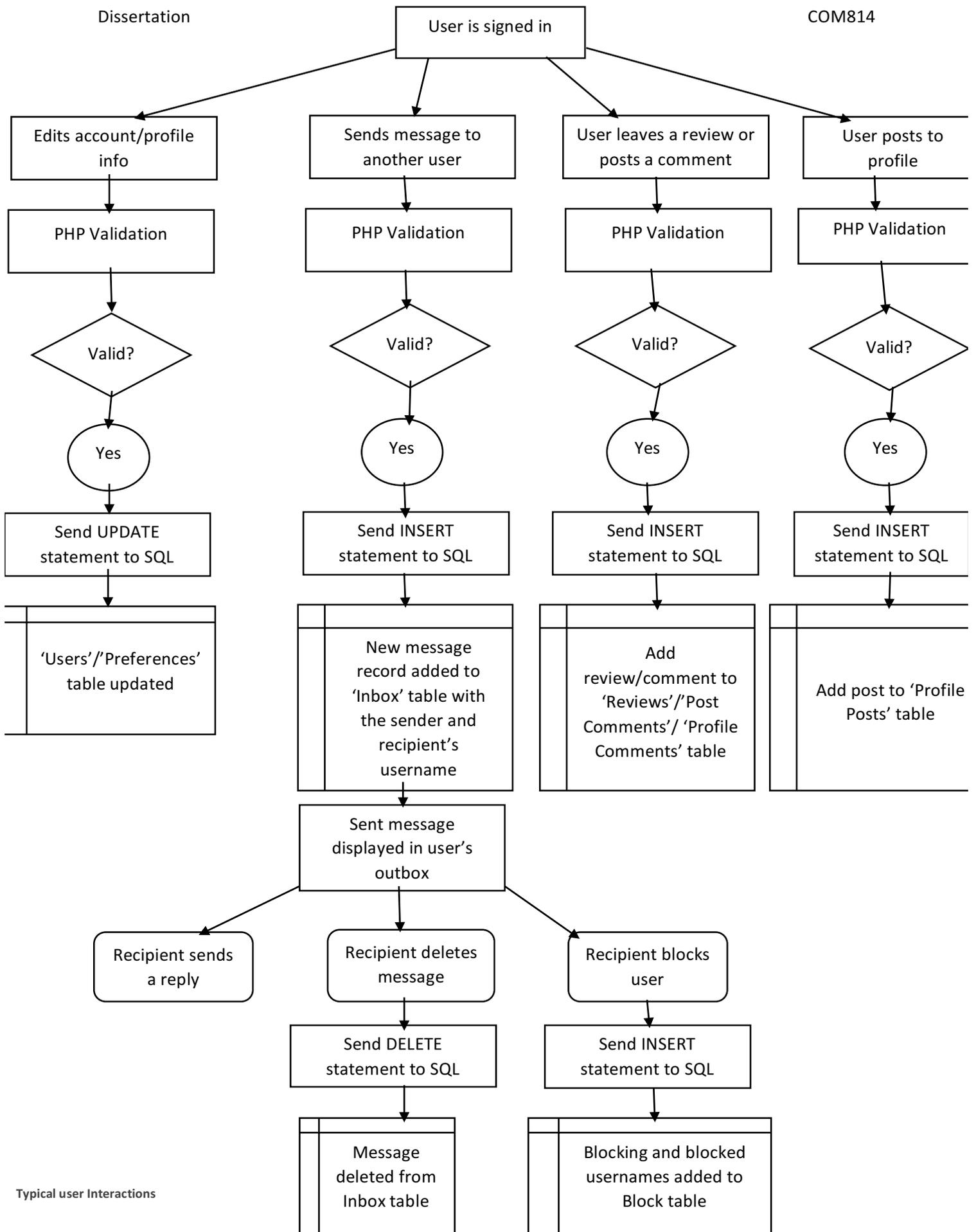
User login/registration



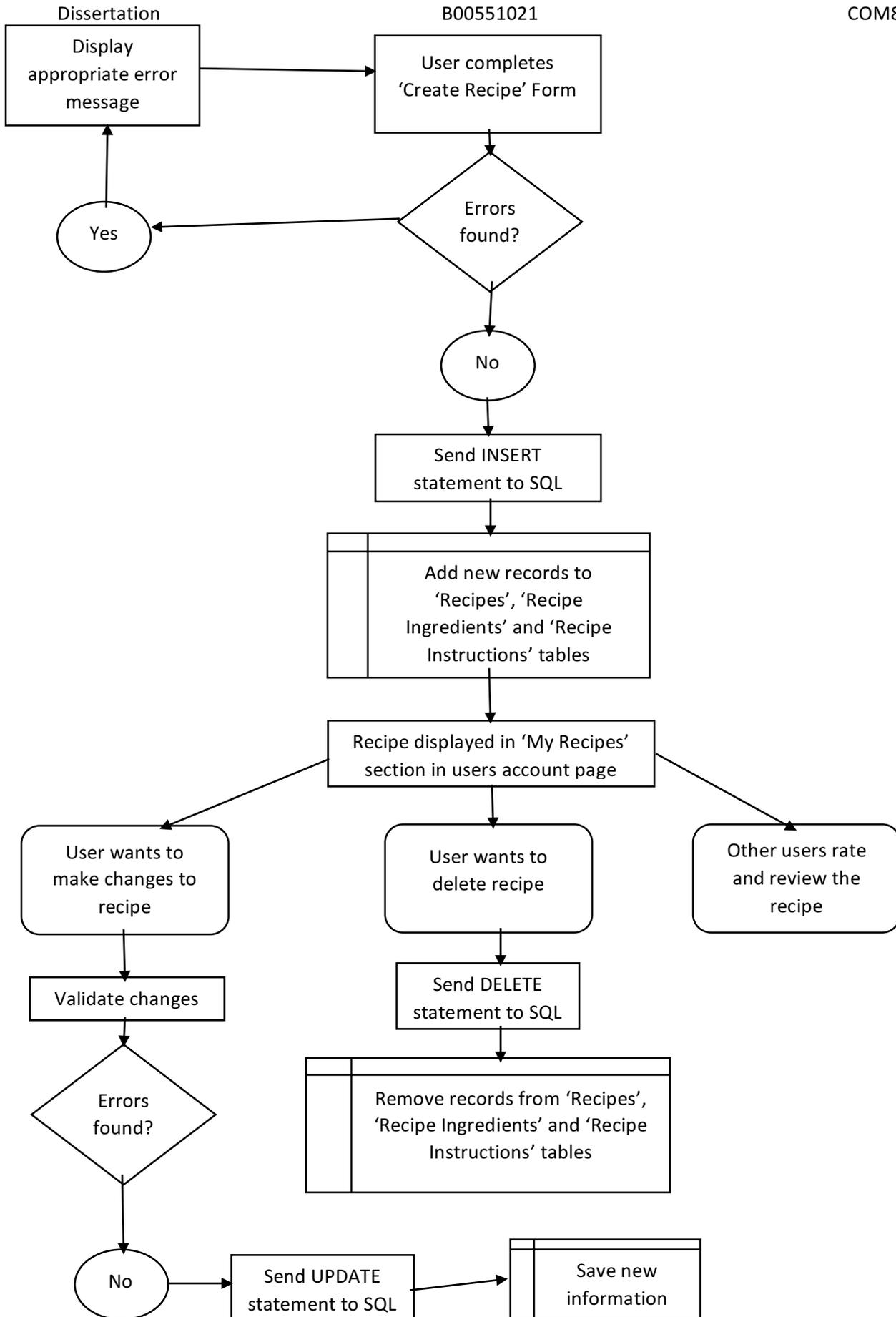
Recommendations/Home page



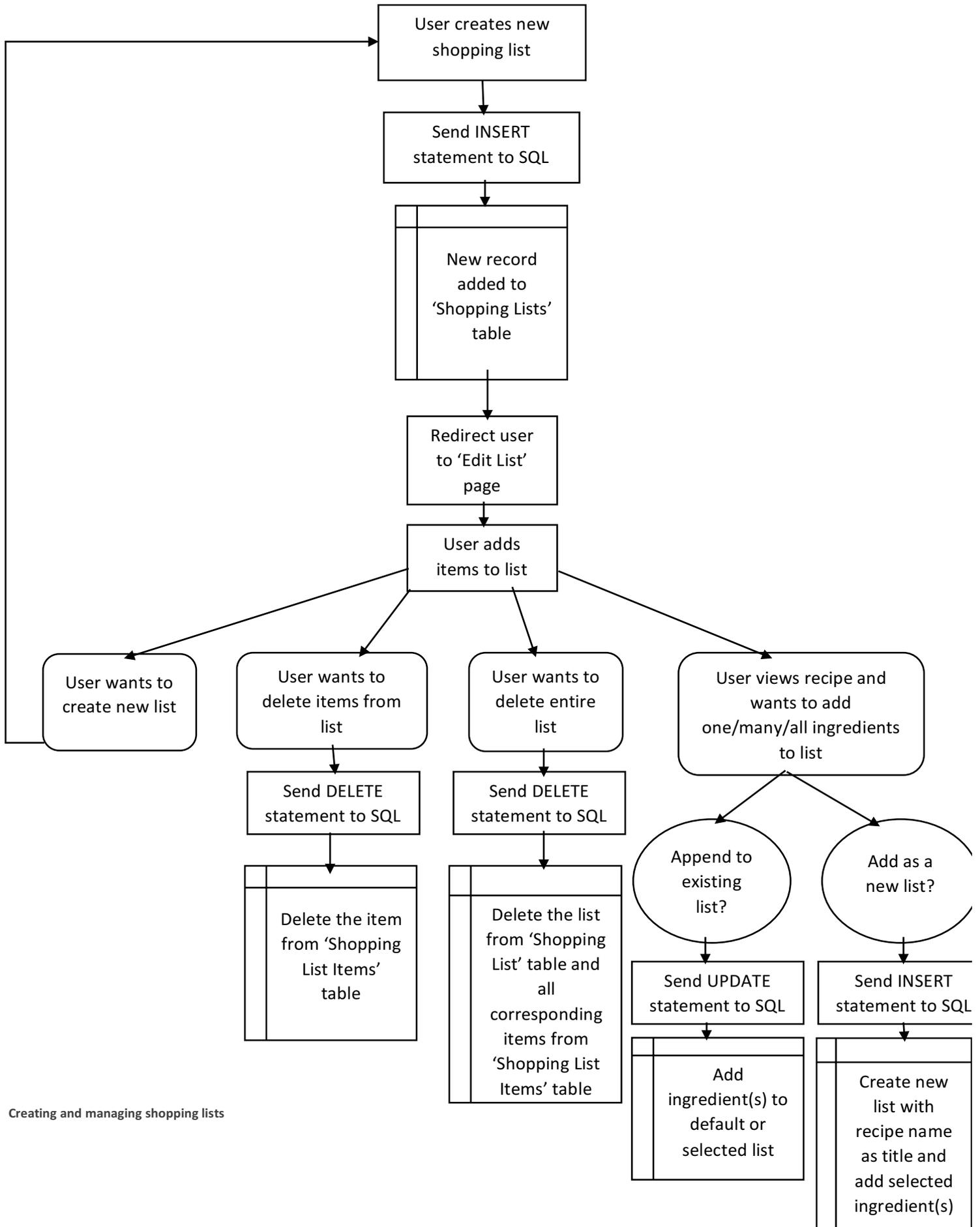
Searching



Typical user Interactions

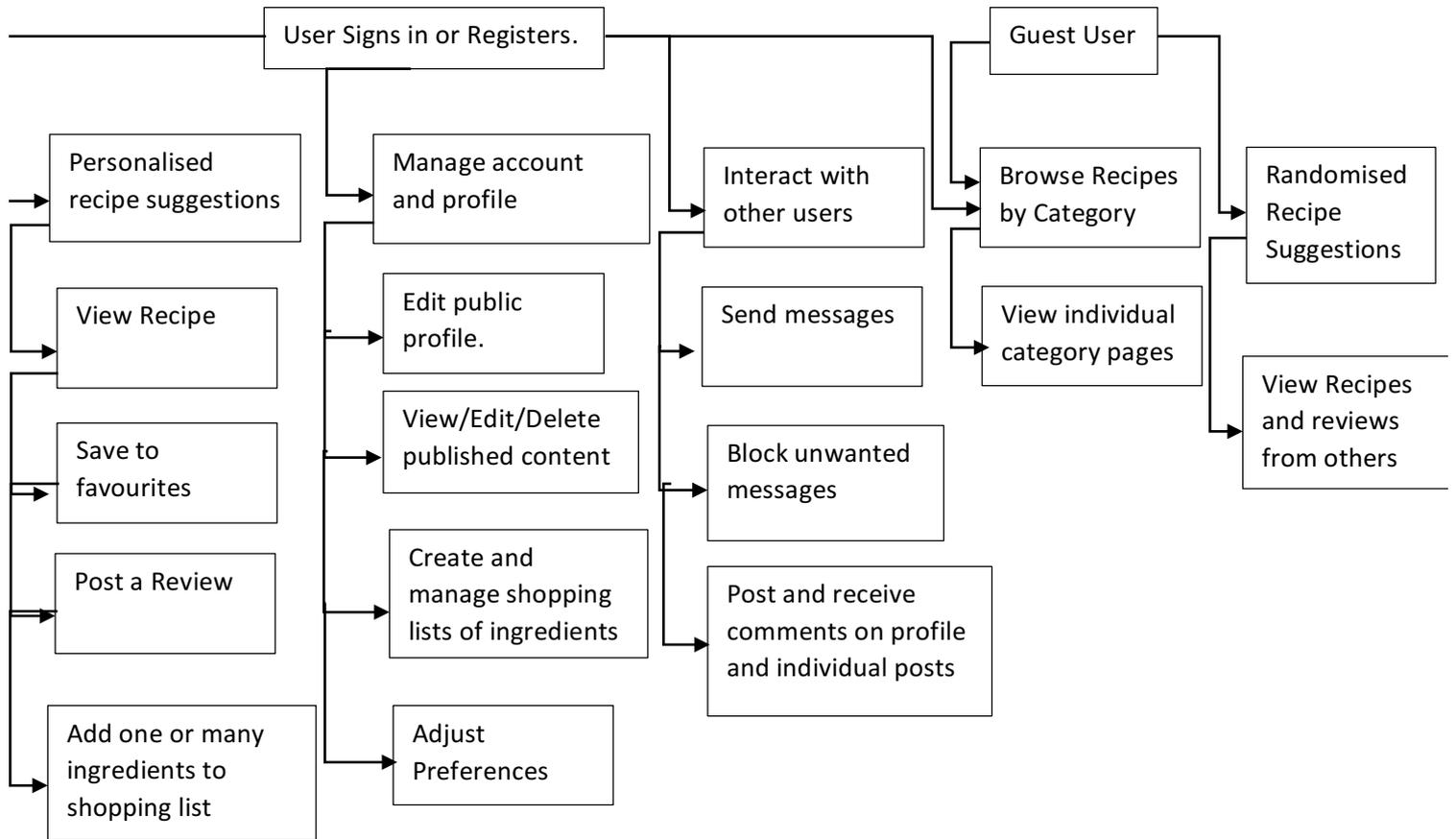


Creating and publishing a recipe

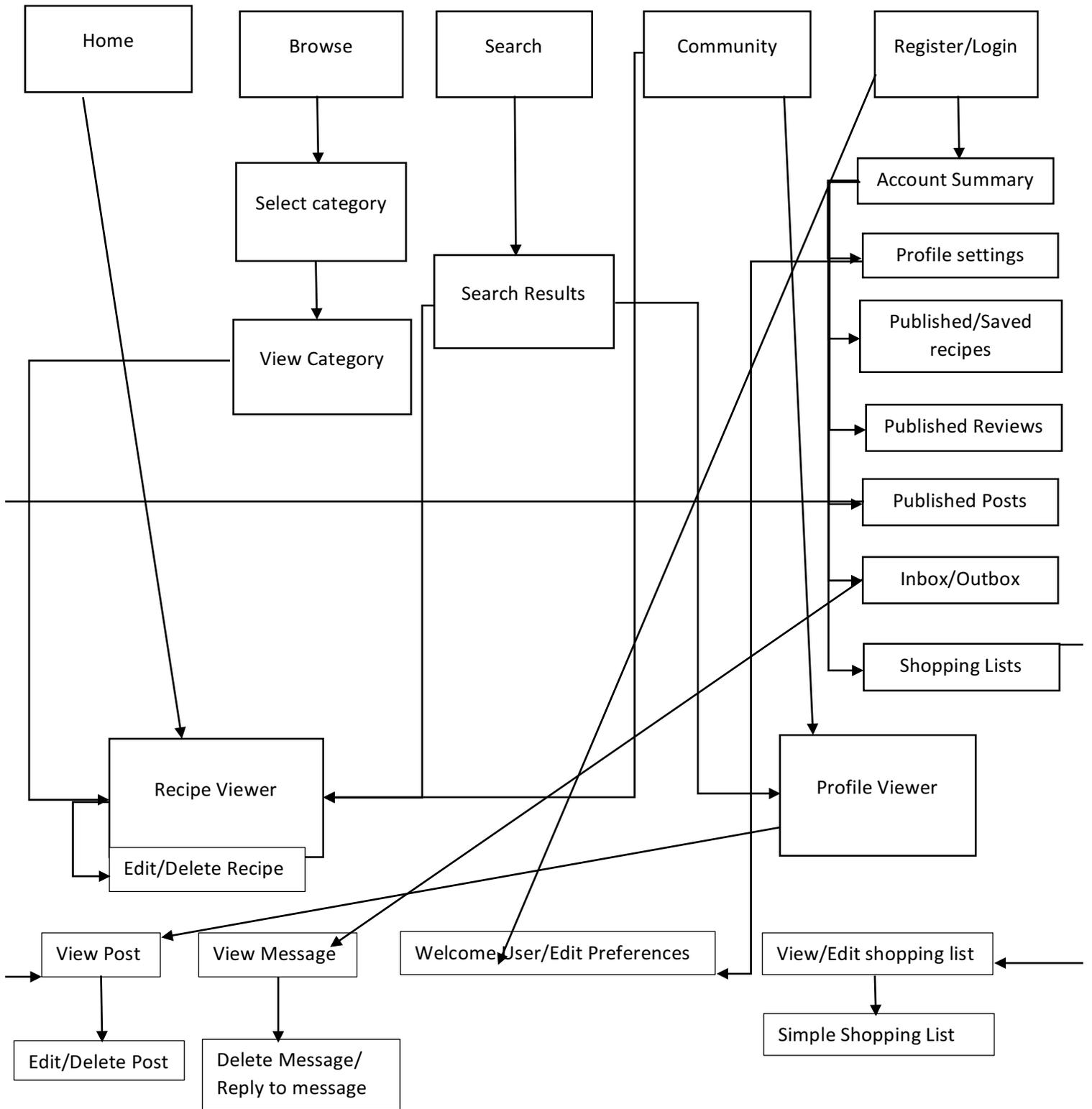


Creating and managing shopping lists

Appendix 6 – Architecture Diagram



Appendix 7 – Site Map



Appendix 8 – Database Design

Entity Listings

USERS (Username#, password, email, first name, surname, avatar reference, bio, join date, profile link, profile views, privacy setting)

USER PREFERENCES (Username#, likes, dietary preferences)

USER REVIEWS (Review ID, username#, recipe ID#, rating, review title, review text, review date)

USER SEARCH HISTORY (Username#, searched for, date of search)

USER VIEW HISTORY (Username#, recipe ID#, view date)

USER INBOX (Message ID, Username#, Sent By, Date Sent, Subject, Message, Viewed?)

USER BLOCK LIST (Username#, Blocked Username#)

USER FAVOURITES (Username#, Recipe ID#, Date Added)

USER FOLLOWING LIST (Username#, Following Username#, Date Added)

PROFILE POSTS (Post ID#, Username#, Post Title, Post Text, Image 1, Image 2, Image 3, Post Date, Post Link, Num Views, Num Comments)

POST COMMENTS (Post ID#, Poster Username#, Comment Text, Comment Time)

PROFILE COMMENTS (Username#, Poster Username#, Comment Text, Comment Time)

RECIPES (Recipe ID#, Username#, Title, Description, Course Keywords, Type Keywords, Style Keywords, Dietary Keywords, Difficulty, Preparation, Cooking Time, Servings, Image, Upload Date, Recipe Link, Edit Recipe Link, Recipe Views, Num Reviews, Num One Star Reviews, Num Two Star Reviews, Num Three Star Reviews, Num Four Star Reviews, Num Five Star Reviews, Overall Rating)

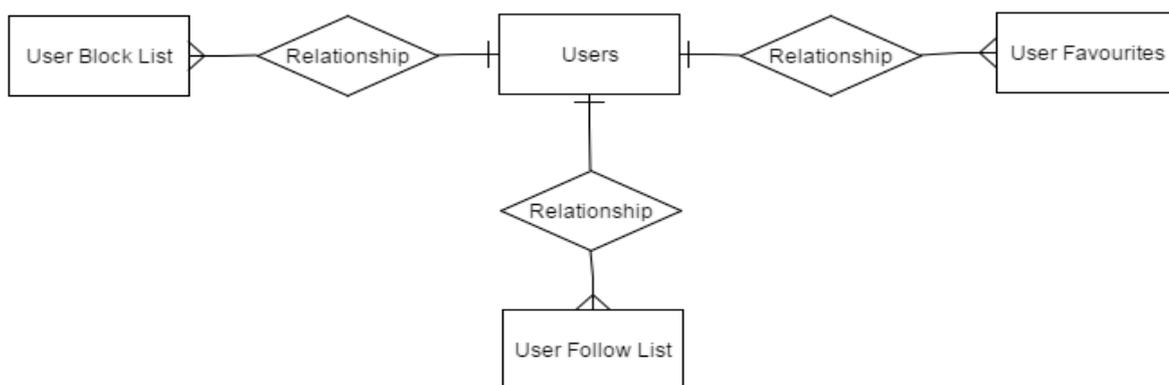
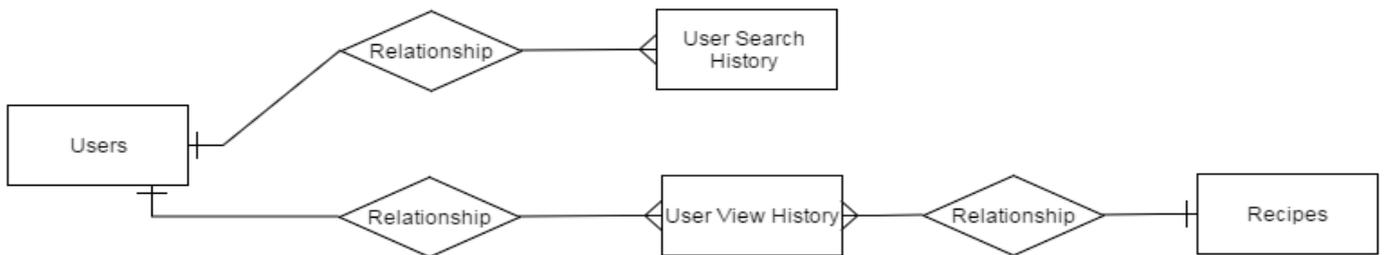
RECIPE INGREDIENTS (Recipe ID#, Ingredient Number, Quantity, Description)

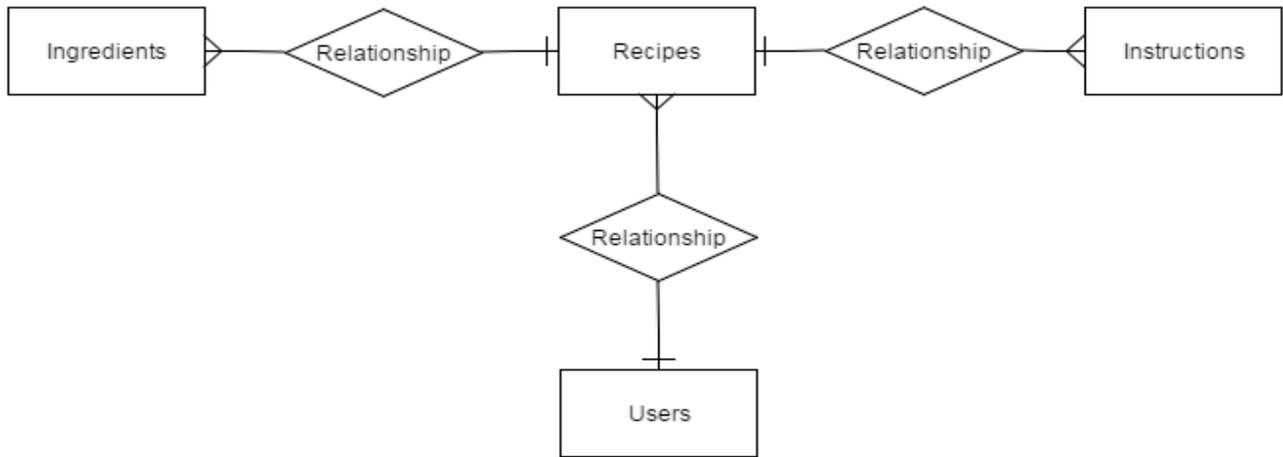
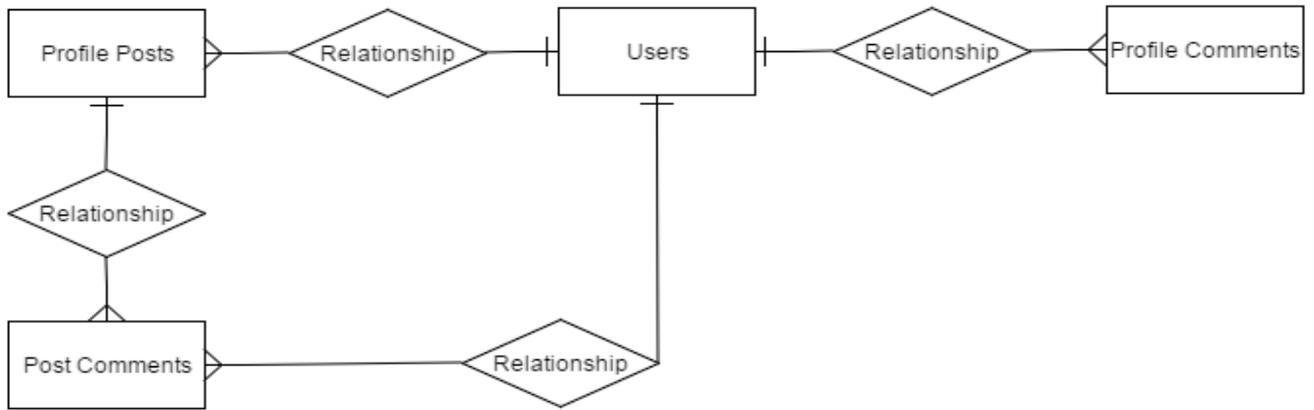
RECIPE INSTRUCTIONS (Recipe ID#, Instruction Number, Instruction Text)

SHOPPING LISTS (List ID#, Username#, List Name)

SHOPPING LIST ITEMS (List ID#, Item Number, List Item)

ER DIAGRAMS





Appendix 9 – Forms and Error Handling

FORM	FIELD	ERROR HANDLING
New User	Username	<ul style="list-style-type: none"> Username field must not be empty. Username must only contain letters or numbers. Username must be between 5 and 50 characters. Username must not already be associated with another user record.
	Password	<ul style="list-style-type: none"> Password field must not be empty. Password must only contain letters or numbers. Password must be between 5 and 20 characters. Password must match value entered in the “Confirm Password” field.
	Email	<ul style="list-style-type: none"> Email field must not be empty. Email address must be in a valid format (i.e. username@provider.com). Email address must not already be in use by another user.
	First Name	<ul style="list-style-type: none"> First name field must not be empty. First name must only contain letters or spaces. First name must be between 3 and 50 characters.
	Surname	<ul style="list-style-type: none"> Surname field must not be empty. Surname must only contain letters or spaces. Surname must be between 3 and 50 characters.
	Avatar	<ul style="list-style-type: none"> File type must be JPG, JPEG or PNG. File size must not exceed 1MB.
Existing user login	Username	<ul style="list-style-type: none"> User must enter a value. Username must match saved value in user’s record.
	Password	<ul style="list-style-type: none"> User must enter a value. Password must match saved value in user’s record. Password must be in the same case as the saved value in the user’s record.
Create a Recipe	Title	<ul style="list-style-type: none"> User must enter a value. Title must be between 5 and 200 characters.
	Description	<ul style="list-style-type: none"> User must enter a value. Description must be between 5 and 500 characters.
	Recipe Image	<ul style="list-style-type: none"> File type must be JPG, JPEG or PNG. File size must not exceed 1MB
	Keywords	<ul style="list-style-type: none"> User must pick at least one keyword.
	Difficulty Preparation Cooking Time Servings	<ul style="list-style-type: none"> User must select a value from the drop down options.
	Ingredients	<ul style="list-style-type: none"> User must add at least one ingredient.

		<ul style="list-style-type: none"> The quantity of ingredients cannot exceed 40 characters. The details of an ingredient cannot exceed 200 characters.
	Instructions	<ul style="list-style-type: none"> User must add at least one instruction. An instruction cannot exceed 5000 characters.
Change login details	Change Email field	<ul style="list-style-type: none"> User must enter a value. The new email address must not be identical to the previously stored value. The new email must be in an appropriate format (i.e. username@provider.com). Email address must not be already associated with another user. User must also confirm their password.
	Change Password field	<ul style="list-style-type: none"> User must enter their old password. User must enter a value for their new password. New password must not be identical to their old password. New password must contain only letters or numbers. New password must be between 5 and 20 characters. User must confirm the new password.
New profile post	Post title	<ul style="list-style-type: none"> User must enter a title for their post. Title must be between 3 and 200 characters.
	Post content	<ul style="list-style-type: none"> Post must not be empty.
	Post images	<ul style="list-style-type: none"> File type must be JPG, JPEG or PNG. File size must not exceed 1 MB.
New shopping list	List name	<ul style="list-style-type: none"> New list name must not exceed 100 characters.
Update profile	Bio	<ul style="list-style-type: none"> Bio must not be over 1000 characters.
	Avatar	<ul style="list-style-type: none"> File type must be JPG, JPEG or PNG. File size must not exceed 1 MB.
Send message	Subject line	<ul style="list-style-type: none"> Subject line must not be empty. Subject line must not be more than 150 characters.
	Message text	<ul style="list-style-type: none"> Message text must not be empty.
Add item to shopping list	New item	<ul style="list-style-type: none"> New item must not be more than 75 characters.
Recipe User Reviews	Review Title	<ul style="list-style-type: none"> Title field must not be empty. Title must be between 3 and 100 characters.
	Review Text	<ul style="list-style-type: none"> Text must be included with review.
	Review Rating	<ul style="list-style-type: none"> User must select a rating.

Welcome User form	Bio	<ul style="list-style-type: none">• Input for the bio should not exceed 1000 characters in length.
	User's likes	<ul style="list-style-type: none">• Input should include only letters and/or spaces.• Must not exceed 250 characters and be more than 3 characters.