Interactive Application Programming Techniques (IAPT)

Open Group Assessment - Group 6

Interactive System Quality Reporting (Part B)

This document presents a series of quality assessments of Group 6's implementation of the interactive cooking application (part A of the assessment). Appendix A contains all the website screenshots and diagrams, that will be referenced throughout this document.

1. Reliability and Maintainability

Group 6's cooking website uses the open source *Codeigniter* [1] web application framework as its core framework. Codeigniter is based on the Model View Controller (MVC) pattern, ensuring separation between the logic and presentation of the application. It is a framework with a small footprint that requires very little configuration, yet it features a wide range of libraries that are useful in building PHP applications.

Using Codeigniter for our application has ensured consistency in managing the different classes and components that have been produced. The framework provides an easy mechanism to register and load a specific Controller for a given URL pattern. Configuration is centralised: the loading of required libraries and dependencies (including additionally added helper methods) is automated. In addition, database interaction with Codeigniter is simplified - a feature, which has been particularly useful for the cooking website, is the generation of full object instances from SQL result rows, in a Data Mapper fashion. The latter has helped in separating the database structure from the Model components.

All of the views of Group 6's cooking website have been developed with the aid of the popular *Bootstrap* [2] front-end framework. Bootstrap ensures consistency in our application's layouts and web components, through providing commonly used interface components and CSS rules in one package. In addition, Bootstrap is highly customizable and has saved a lot of development time.

With respect to source code **versioning and management**, we chose to use Git because of a common familiarity and preference among group members. Hence, the source code has been maintained in a private GitHub [3] repository, which also has handled all of the website's code revision history. GitHub also offers some useful project management features, such as milestone and issue creation. The latter were used extensively in order to keep track of the development progress through task assignment to group members and to highlight code bugs.

Generally, there are two development environments for the cooking application. Each developer's workstation has been used as a **local development** and **testing** environment. Every developer of our team is responsible for performing **tests** on our application prior to every major release (this is discussed further in the test plan section). Every major release (or milestone) of the application

also involved deploying the application's code on our **production** environment - the Computer Science (CS) server. The latter is equivalent to merging the Git code base with the Computer Science SVN repository. This process ensured that only fully working and verified code is present on the CS server.

The main tool, which has been used for code **profiling** by all of the developers of this project, is Codeigniter's built-in profiler [4] This feature has been very useful in development, because of the various debugging information that is displayed, including the exact SQL queries which have been executed by the Model, the $_POST data that was received, and others. The profiler has helped in maintaining the quality of the application.

All the code of the cooking website has been developed with the Codeigniter **code style** guidance [5] in consideration, to ensure that the code is readable. This includes correct commenting of PHP function definitions, using DocBlock, and correct naming of function definitions, variables, and others. In addition, Eclipse's automatic formatting feature has also been applied to the code base, to ensure consistent spacing and layout of all of the code.

The cooking website is fully written in PHP - no templating language has been used to create the views. This is because the team decided to avoid unnecessarily complicating the development process; all of our team members, even non-developers, would be able to read the code without having to learn a new language. This also means that there wouldn't be a steep learning curve for a new developer coming on to the team.

Model of data

Consider *Figure 1*, which contains a UML diagram of the data represented by the cooking application, demonstrating its reliability. There are 3 main classes, namely Recipe_object, Category_object and Ingredient_pool, that represent and contain all of the website's information. The objects are created by the single model of the application, called Recipe_model. Recipe_model queries the MySQL database of the website and automatically builds and returns said objects, which a controller will then load for visualization by the views.

The functionality of object building is greatly simplified for Recipe_model, due to a number of MySQL view algorithms that have been constructed to support the process. This further decouples database logic from the application, and simplifies queries.

Model of processes

A constructed business process model and notation (BPMN) diagram, shown in *Figure 2* illustrates the flow of process semantics of a typical user goal. In this example, we structured the model where the user wishes to make 'Vanilla Slice'. The diagram demonstrates the reliability and maintainability of the application through the choice of actions on this user journey. We took this graphical notation into consideration while developing the system, ensuring that the application's process flow reflected the BPMN diagram.

Test plan

The test plan of the cooking application involves performing the test suites, which have been designed. The outcome of the tests determine whether a major release of the application is ready to be deployed on the CS server. Every group member is responsible for performing the test and communicating the results to the group. Test failures delay deployment and require actions from developers to fix any issues with the application.

The test suites are performed on each group member's personal environment manually. Each member executes every test in turn, recording the outcome (pass or fail) and including further comments if necessary. Finally, once a new version of the website is deployed on the CS server, the tests are executed once again in order to verify that deployment has been successful.

Below, the core test suite has been provided, which tests general key aspects of our application, across all of the website pages.

Example test suite - General website functionality

The below table summarizes the test cases for this test suite.

ID	Test case steps	Expectation	Pass/Fail
1	Click "Advanced" on the welcome page. Pick any recipe from the home page. Change the currently selected user style from the top drop-down menu of the website to "Novice".	The preparation panel should change to display the recipe cooking steps.	
2	Click "Advanced" on the welcome page. Open any recipe from the home page.	You should reach the recipe's individual page.	
3	Click "Advanced" on the welcome page. Click on the "Surprise me" button.	You should be taken to a random recipe's individual page.	
4	Click "Advanced" on the welcome page. Inspect every recipe on the home page.	Recipe should include information about the recipe's category, number of servings, cooking time, and list of categories.	
5	Click "Advanced" on the welcome page. Search for "burgers" from the search bar at the top page.	Verify that a burger recipe is present.	
6	Click "Novice" on the welcome page. Select a recipe from the home page. Click the "Next" button below the recipe preparation panel.	The current preparation step should change to the next step.	
7	Click "Advanced" on the welcome page. Navigate to	You should see a list	

	the "Main dish" category via the categories drop down menu. Hover over a recipe's ingredients information icon.	summarizing a recipe's ingredients.	

2. Design Rationale

This section aims to describe the general design of the system, initially introducing how it addresses the goals of users. Two pieces of functionality are also explained in terms of design choices, relating to existing design principles such as Shneiderman's, Don Norman's [6] and Tog's [7], and also the usability.gov [8] guidelines. We addressed most of the design principles in our early prototype but for illustration purposes, we describe the final system in this section too.

The interactive system is designed as a service that provides a variety of recipes that users of any cooking level can easily follow at home. On immediate entry of the website, the welcome page, shown in *Figure 3*, asks the user what display style they would prefer, according to their skill level of cooking. The presentation styles were initially termed 'Step-by-step', 'Segmented' and 'Narrative', but after our collaborative heuristic evaluation (CHE), we renamed them 'Novice', 'Intermediate', and 'Advanced' respectively. Reasons for this redesign are explained in the *Heuristic Evaluation* section. Nonetheless, this choice already defines the website around the user's preference, making them feel more comfortable when it comes to following instructions. This choice is related to *Tog's principle of Autonomy* and *Shneiderman's rule to "support internal locus of control"*, whereby the users are the initiators of actions and therefore feel like they are in charge of the system.

When users want to look for a recipe, they have several ways of doing this. If they already have a meal in mind, they can simply search for it using the search bar, using either a recipe name or an ingredient. Users can also look for meals under different courses, for example main dishes or desserts, using the categories drop-down menu (*Figure 17*). This was also designed around the goals of the users as many people may only wish to cook a certain course, or they may even have ingredients that they want to use before going off. A final touch for finding recipes is the "Surprise Me!" function, as shown in *Figure 26*. This provides the user with a random recipe, for example if they are struggling to think of a meal to cook. Another reason that inspired this inclusion is that most users tend to follow a recipe because they do not know how to cook that certain meal, so using this function instantly allows them to try something new.

Alternatively, the user can select one of the recently added recipes from the home page of the application (*Figure 27*), which is available after the user selects his initial presentation style. It enlists the 5 most recent recipes in a panel, describing useful information about every recipe. Hovering over each of the individual recipes displays a popup with a short description of the recipe (*Figure 28*).

The navigation system holds links to the homepage, categories, the different presentation style choices and the search function. The navigation bar is consistent as it is fixed on every page and holds the same elements.

A detailed description of two substantial pieces of functionality of our design is provided next.

The Search Function

The interactive system has several functionalities that vary in complexity, but are all designed to support the usability of the application. One substantial piece of the system that has fairly complex functionality is the search function. It provides a quick alternative to general application navigation, which has the potential to be very effective for users.

The search function (*Figure 4*) allows the user to type anything into the 'search box', ranging from ingredients to categories, and suggests the meals according to the given search. The search box is situated at the top right of the website header and remains there on every page. This is linked to *Shneiderman's first rule of consistency*, *Don Norman's principle of visibility* and the *usability.gov 17:4 guideline* which suggests that the search option should be provided on each page. The user can search for a different recipe at any point, and can always refer to the same place to do so. The magnifying glass is also a common metaphor used to indicate to the user that it is a function for searching. When the user hovers their mouse over the magnifying glass, it highlights, indicating that it is a working function. This is linked to *Don Norman's design principle of affordance* and also *Tog's "human interface objects"* as the magnifying glass is also a button allowing execution of the search function.

Users would typically have two different goals when they search. First, a user could look for recipes containing a certain ingredient that they like, for example "lemon". For users who have no idea what a search bar does, a hint has been included as stated in the *usability.gov 17:8 guideline*. This is done by setting the value to "Search meal/ingredient" within the search bar, which disappears once the user types in their request, and a tooltip is provided to further explain this (*Figure 5*). This notifies the user of the search choices they can make. When the user enters "beans" into the search bar, it returns recipes that includes "beans" in its ingredients, as well those with it in its recipe name. An example of the search results for this is shown in *Figure 6*. The application has a consistent layout as typical search results have a similar layout to the category list. Further to that, upper and lowercase searches are equivalent, as stated in the *usability.gov 17:3 guideline.* In the case where a user types in a search request of 2 characters or less, the function can handle such errors by notifying the users of the character limitations, as shown in *Figure 7.* Also, when it attempts to search for an ingredient/recipe that does not exist, it displays a simple message (*Figure 8*). This function therefore complies with *Shneiderman's rule to offer simple error handling*.

Another goal would be for the user to find a specific recipe that they have been looking for. For example, if the user were looking to cook a quiche, then searching for "quiche" would return them with a list of quiche recipes. Sometimes the user may be looking for a specific recipe but can only remember part of the name. An instance of this could be if they could only remember it had the word "maritime", but not exactly what the recipe was called. The search function would look for recipes with "maritime" in its name or ingredient list, ultimately returning recipes such as "Quiche Maritime".

In the final stage of development of our system, we have included filters (as in *Figure 6*) with the

search results as a result of the redesign after the task-based user evaluation. This layout remains consistent with the category list page and it enables users to refine their search further. Users can further filter search results in order to discover a specific recipe, or to downselect the available recipes according to the criteria that's available. This gives users more control over the system (*Shneiderman's rule to support internal locus of control*).

For users who end up searching very often, or for regular users, we've added a shortcut to reduce their number of interactions, which increases their pace of interaction. The use of shortcuts can be found under *Shneiderman's second rule* stating to *"enable frequent users to use shortcuts"*. The user would simply have to press the 's' key, and the search bar will be selected and ready for entry. This instruction is also given as a tip in the Help section which is found in the footer of the website (*Figure 23*).

Step view of individual recipe page

Another piece of functionality with interesting design implications is the method of following instructions on a recipe page. On any recipe page, the set of instructions can be found underneath the ingredients. Consider the Beef Burger recipe page, shown in *Figure 9*, with the chosen presentation style as 'Novice'. Immediately you can tell that the first instruction is highlighted in blue, depicting the current step. Users can navigate through the set of instructions with the provided buttons below, to either move to the previous step, next step, or go back to the start.

Though the display for the instructions seems small, as the user scrolls down the set of instructions, the current highlighted step is kept at the centre of this box (*Figure 10*). This is potentially very effective for users as it helps them keep track of their progress; they do not have to remember the step they are on in between carrying out instructions, due to the highlighting. These design implications effectively provide feedback (*Shneiderman's and Don Norman's etc.*) as the step panel scrolls down and highlights accordingly with every user action. Users can also cross out steps (*Figure 11*) once they are complete by simply clicking on the instruction, automatically scrolling if the current step is crossed out. The cursor changes into a hand pointer, conveying affordance as it provides a hint to the user that it is interactible. Additionally, the ingredients box above can be minimised by clicking on "Ingredients" with the arrow icon that indicates it can be hidden, as illustrated in *Figure 12*. This can be useful for users once they have collected all the ingredients and want to start following the recipe. Most of these interactions comply with *Shneiderman's rule to allow easy reversal of actions* as the users can scroll back up the set of instructions, re-expand the ingredients list, and uncheck any steps they have crossed out.

The navigation through the instructions is not limited to the buttons provided. A small 'i' icon is presented next to these buttons, indicating that the user can find out more information as they hover over, depicting affordance (*Don Norman*). The tooltip (*Figure 13*) describes the alternative method of navigation as users can also use arrow keys as shortcuts to go through the steps. This was inspired by *Shneiderman's rule to enable shortcuts*. Furthermore, a voice control function is accessible to users who wish to follow the recipe handsfree. This follows *Shneiderman's rule to support internal locus of control* as users can feel as though they are in charge of the system, due to it responding to their commands. When clicking on the "Enable voice control" button, simple

instructions are provided to make users feel comfortable with this experimental feature (*Figure 14*). This is ideal for users as they can use the application without getting their device dirty whilst cooking.

In the case that the instruction list is too long, the number of completed steps, as well as the total number of steps, is provided above the instructions, which can be used by users to monitor their progress (*Figure 15*).

Where the presentation style is 'Intermediate' instead of 'Novice', the system displays the recipe instructions in a layout identical to the one described above. Each recipe also uses the same operations and elements, conveying consistency, as suggested by *Don Norman, etc.* The 'Advanced' setting simply displays the instructions as a block of text, so general functionality described above for skipping steps is not applicable here.

3. Heuristic Evaluation

A Collaborative Heuristic Evaluation (CHE) was performed on a low-fidelity prototype of the cooking website, employing the Petrie and Power Heuristics for Interactive Web Applications. The prototype is effectively a wireframe that was created through the Lucidchart web application. It represents the different pages of the desired system to a level of sufficient detail.

For the heuristic evaluation, the goal is to allocate a group member to "drive" the application, performing various tasks believed to be common for the end users of the website. The tasks were devised by considering several previously created personas of imagined users of the website, that vary in age, skill and background. The application "driver" then undertook tasks in the shoes of the different personas. The personas used for this evaluation, on top of the two provided in the specification, are introduced below:

Persona 1: Kyrie
Kyrie is a 27-year-old businessman. Ever since he met his fiancée, he has always been trying to expand his variety in the meals that he can cook. Kyrie is also a little bit of a gym freak, and whenever he has free time to himself, he tends to spend it there. Being a businessman, Kyrie is very capable when it comes to computers but he definitely doesn't want something that will waste his time.

Persona 2: Sarah
Sarah is a 36-year-old single mother to Stephen and George. Looking after the 11-year-old twins is a real pain for Sarah sometimes, especially when cooking, so she always prefers to cook something quick (and painless). Most of the time it's quite difficult for Sarah to keep an eye on her kids as well as the cooking. When her parents are over to babysit, she loves to watch the cooking channel and tries to follow, but sometimes it's too quick for her.

Persona 3: Sergio
Sergio is 42 and originally from Italy. He moved to the UK about a year ago to pursue his career in

automobile engineering. Sergio's father owns two restaurants back in Italy, so he loves to cook, especially Italian food. He isn't that fluent in English since it's not his first language but he enjoys reading novels in his spare time to help him with his vocabulary. Sergio regularly uses his laptop to learn more about the UK and ever since moving there he has been trying to cook English meals.

For each of the tasks, usability problems that violate any of the Petrie and Power heuristics were recorded, along with the exact page location of the problem. Each group member (GM) privately rated the severity of the problem after the "driver" had finished. *Table 1* below enlists the full results of the evaluation:

Issue #	Usability Problem Description	Persona Used	Page	Heuristic	GM 1	GM 2	GM 3	GM 4	Mean
1	Cannot decipher what the presentation styles mean and the differences between them.	Sarah	Welcome screen	7,8,9,12,13	3	2	3	3	2.75
2	No clear labels and instructions when choosing a presentation style on the welcome screen.	Dale	Welcome screen	7,9,10	2	2	3	2	2.25
3	There are 2 ways to change the presentation style, both with different descriptions.	Jennifer	Recipe	7,9,11	2	3	2	2	2.25
4	Currently no help section.	Sergio	Any	6	2	2	3	2	2.25
5	"Surprise Me!" function is not clear - opens a new page without clearly stating what it is.	Kyrie	Homepage	7,9,10,18	2	2	2	2	2
6	Search could be for either recipes or categories - unclear what part of the website it searches.	Sergio	Any	7	2	2	2	1	1.75
7	Difference between "Cook" and "Take a peek" is not clear.	Jennifer	Category	9,12,19	1	1	2	2	1.5
8	Potential duplication of information and category navigation on the "Take a peek" page.	Kyrie	Take a peek	11	1	2	2	1	1.5
9	"Go back" is not clearly defined (go back in instructions or go back a	Jennifer	Recipe	10	1	1	2	1	1.25

	page?)								
10	Scroll bars and pagination may confuse users.	Jennifer	Category	16	1	1	2	1	1.25
11	Nothing states that the instructions are the method for preparation.	Sarah	Recipe	10	1	2	1	1	1.25
12	Vertical bar on the latest 5 recipes looks like it can be dragged but it cannot.	Jennifer	Homepage	19	1	1	1	1	1
13	Don't know which category each of the recipes belongs to.	Dale	Recipe	17	1	1	1	1	1

Table 1: Usability problems described with their corresponding heuristics violated.

The severity ratings recorded range from 0 to 4, as described below:

- 0 – not a usability problem in the opinion of the expert
- 1 – a cosmetic problem which is an irritant to the user but will not interrupt their progress much
- 2 – a minor problem which will interrupt the user task, but they will be able to recover; also for problems which will occur once and then likely will not happen again
- 3 – a major problem which will interrupt the user task for a longer period of time; the user will have to fall back on some problem solving skills to move on in the system
- 4 – a usability catastrophe (very major problem) that will stop the user from proceeding in their task; used for times where a user is unable to continue (e.g. trapped in a dialogue) or where they are likely to give up (e.g. lost in the information architecture of the website).

The severity ratings are included in the above table, and the problems are ordered by the mean severity rating.

The piece of functionality, which had the highest mean severity rating, and also violated the largest number of heuristics, is the choice of user presentation styles of the application. Interestingly, the top three most-severe problems highlighted by the evaluation, were all related to the presentation style buttons. *Figure 19* and *Figure 16* illustrate the problem area of the application, and will be referenced in further discussions below.

Problem description, criticality and redesign

The presentation and mechanism that is used for managing the three presentation styles of the cooking application was identified as the most critical problem from the heuristic evaluation. We will now discuss the specifics of the problems, referencing the heuristics that have been violated. Below, each heuristic is enlisted, along with an explanation of the problem areas. Descriptions of solutions are provided, with reference to redesign (which is the actual implementation, rather than a wireframe).

Content 7: *Provide clear terms, abbreviations, avoid jargon.*

The evaluation indicated that the textual descriptions originally defined to refer to the three different presentation styles do not convey the differences in presentation styles to users in a sufficiently clear and unambiguous manner. The welcome page had cooking levels presented as 'Novice, Medium and Pro', as shown in *Figure 16*. This definition provides no correlation with the conceptual meaning of the presentation styles (they allow the user to change the presentation of recipes), hence the potential to confuse users. Our redesign has addressed this by renaming the three terms to 'Advanced, Intermediate and Novice' (as depicted by *Figure 3*), which, in our opinion, bears a clearer resemblance to the actual implementation of presentation change.

Information Architecture 8: *Provide clear, well-organised information structures.*

The presentation styles dropdown button (see *Figure 18*) also could confuse users to assume that the element is somehow closely connected to the navigation of the website. There is also no information of the exact function of this menu. The welcome page (*Figure 16*) also isn't helpful in explaining what the three presentation styles are, how they function, and can be used to control different aspects of the website.

To address these issues, first, we added a "Current style: " text prior to the value of the currently active style (*Figure 20*) to provide a clearer meaning to the dropdown button. In addition, tooltips were added to each individual element of the dropdown, which contain valuable descriptive information about each of the presentation styles (*Figure 21*). The same tooltips are also present on the welcome page, as discussed later. This functionality clarifies the meaning and purpose of the presentation button, which is also present in all pages of the application.

Interactivity 9: *How and why - provide clear explanations of how the interactivity works.*

An additional problem, as discovered through the evaluation, is the lack of clarity in describing the function of the three presentation styles of the application. This is applicable to the welcome page (*Figure 16*) and the presentation drop down (*Figure 18*). To address this issue, we expanded the textual description available on the welcome page (*Figure 3*) and added tooltips for each presentation style (triggered on hover), as in the presentation toggle button (*Figure 21*). A help section is also provided, with a link to it present in the footer of the application (which is always available for users to consult with - *Figure 23*). The help section describes some common concepts regarding the application, which may be useful to some users.

In addition, it was identified that there's a lack of feedback to users when switching between the different presentation styles of the application. Although the currently selected style is present in the header (as discussed previously), we also introduced simple notifications on style change (*Figure 22*). These provide users with clear feedback that their interaction with the application has been successful. We felt that all the adjustments in the redesign sufficiently address the issues outlined here.

Interactivity 12: *Make input formats clear and easy.*

The wireframe violated this heuristic, because it provided two different controls for controlling the presentation style at the same time on the recipe page. This is demonstrated by *Figure 19* - observe

that besides the presentation dropdown in the header, there's also a set of radio buttons below the recipe title, which are also targeting presentation styles. The evaluation indicated that this has the potential to confuse users, especially since two distinct types of representations refer to the same functionality. A similar problem was also discovered on the welcome page of the application (*Figure 16*), in which there are also two options for changing the presentation styles - in both the welcome box and the header.

Those problems were addressed by our team in our redesign, by first, removing the presentation radio buttons from the individual recipe page. We decided that it is a redundant feature, because users can still change their presentation from the dropdown in the header. Second, we decided to hide the presentation dropdown on the welcome page *only*. We decided that, on entering our website, users should be focused on the welcome page, which introduces them to the website. The welcome page would also be the place to choose the initial presentation of the cooking application (or skip it, which is also an option). Upon presentation selection, the dropdown is re-displayed, allowing users to change their presentation at anytime. Those changes address heuristic 12 through providing a single method for presentation control across all of the pages of the application.

In addition to the heuristics, which have been discussed in detail so far, the parameter styles presentation and mechanism was identified to violate 2 additional heuristics. The first, "Interactivity 13: Provide feedback on user actions and system progress." relates to the feedback that lacked with presentation style changes, and was discussed in previous heuristics (the presentation style change bubble, along with the "Current style" text address this). The second, Interactivity 11 - "Avoid duplication/excessive effort by users", which was also violated, has also been addressed in the discussion of the Interactivity 12 heuristic above.

4. Task Based User Evaluation

This section describes the task-based user evaluation, which was performed on the cooking application. It required every group member to conduct evaluation with the aid of two users, who were each assigned a different task to accomplish on the website. Each user was new to the web application.

The tasks, that were devised for the user evaluation, are each aimed at testing different important aspects of the website. Some tasks are more critical than others, but they are all based on functions that can be seen as potentially complicating or confusing to new users of our application, as determined by group members. We also attempted to make the tasks as natural as possible - tasks represent activities that people may generally decide to do when using our cooking application, or any similar cooking application.

The tasks were performed on a variety of environments (MS Windows, Linux, Mac), on systems with different screen resolutions (varying from 1366x768 to 1920x1080). Firefox, Chrome and Safari were all used. This ensured that we cover a wide range of browser settings, which would also

allow us to get a better overview of the performance and behaviour of the cooking application on different environments.

The process of performing a user task is as follows. First, users are left to get familiar with the website for a few minutes. Then, they are returned to the starting page of the application, afterwards that they are briefed for the task that they should aim to accomplish. Some of the tasks include a small scenario of the mindset with which the user should approach the task. Users are encouraged to comment as they undertake the task, especially if they hesitate or come accross a problem. The users then proceed with performing the task, while the evaluator (group member) notes down any usability problems, comments or suggestions provided by the user. Finally, once the task is accomplished, the users are asked to rate the severity of each of the usability problems that had been recorded from 1 to 4, similar to the ratings used in the collaborative heuristic evaluation.

Usability evaluation results

The full tasks, results and recorded usability problems are enlisted below. The results are outlined by stating the task number and summary (in italics), following which a short description of the user's approach to the task is provided. Next, any usability problems are listed, along with their severity as judged by the users (severity ratings are in the same scale, as used in the IAPT practicals).

Task 1
Find a meal that takes 15 minutes or less to prepare.

The user read the welcome description, hovered over the buttons and decided to click on "Intermediate". After going on the welcome screen, the user saw that "Beef burgers" was under recent recipes and saw that it said "15 min." next to it.

No usability problems were highlighted in this task.

Task 2
There's a dish called "Quiche Maritime", find out how many eggs are used to make it.

After reading the brief description, the user decided to click on "Novice" thinking that the instructions might have been a little complicated. After scanning the welcome page, thinking that they were the only recipes, the user clicked on "Quiche Maritime", scrolled down on the ingredients list and saw "3 eggs beaten" and so deduced that it needed 3 eggs.

No usability problems were highlighted in this task.

Task 3
You're having a small get-together with a few of your vegetarian friends; find a vegetarian meal that can serve up to 4 people.

The user read the description, pointing out that it should have said "follow the recipes your *own* preferred way" and decided to choose the "Intermediate" setting. She then looked at the category list after scanning the homepage and realised that there were no vegetarian meals available. After being unsure which category to select, the user decided to go with "Main dishes" anyway and scrolled down along the meals. She thought that the "Quiche Maritime" would have been a suitable vegetarian meal but hovered over ingredients to check anyway. The user saw that it contained fish, so went back to the category drop-down list. After checking the allergen free list, the user looked at the Salad category list and hovered over the ingredients again to ensure the same thing didn't happen. The ingredients tooltip was a little big for the screen as the top got clipped off, so the user clicked on the recipe, scrolled down on the ingredients list and assumed it was safe to cook after seeing that it served 4.

Problems:
- No real category for vegetarians/vegans, therefore making the user unsure which category to select.
  - Severity 3
- Ingredients tooltip a little big for the screen, with the top being clipped off, causing the user to actually view the recipe.
  - Severity 2
- User had to scan the page multiple times on several recipes, including the salad and the quiche to ensure it was suitable for vegetarians. Not enough information on category page to support this process.
  - Severity 2

Task 4
You and your friend are stuck between cooking beef burgers and Tourtiere, but decide that the quickest meal will be the one you make. Find out which is quicker to cook.

The user read the welcome screen description and decided to go with "Intermediate". The user clicked on the search bar to search for "Beef burgers" but realised it was on the homepage. As it said it took 15 mins to prepare, the user then went on to search for "Tourtiere" but ended up spelling it "Toutiere", returning the user with no results. After not knowing which category it belonged to, the user guessed it was a Main Dish and so clicked on the Main Dishes category. The user then clicked on "Tourtiere" and knew where to look for the duration, though it didn't have it on the category page list, and decided that the Beef Burgers were quicker to cook.

Problems:
- The durations being on the homepage next to the recipes but not next to recipes on individual category pages.
  - Severity 2

Task 5

Your friend with a nut allergy is coming over for dinner; find a nut-free meal to cook.

User started the evaluation from the welcome screen with all the user style options. He wasn't sure what the options meant, and didn't really care about them at this stage - so he clicked "decide later". From the home page, he clicked a random recipe, hoping to see if it is nut free, but there was nothing about it on the matter. Then, he searched nut free, but there were no results. After that, he clicked on the Categories drop-down list and saw the "Allergen free" category and clicked the link to go to the category page. The list of recipes belonging to "Allergen free" contained no information about the specifics of the allergy. The user clicked a recipe - Beef burgers - and saw from the top of it that it has a label "Nut free" present next to the title. Task accomplished.

Problems:
- User didn't assume nut free or allergy free would be a category - so he didn't go to the category drop down straight away. He thought that it would be a label of some sort.
  - Severity 2
- In the "Allergen free" category, there's no indication of recipes being "Nut free", which confused the user and made him pick a recipe at random to see if it's nut free. This problem can be paraphrased "Not having sufficient details in the recipe list pages, which displays recipes as lists."
  - Severity 3
- User was a bit confused with the welcome screen. He would have preferred the "Decide later" button to be called "Skip" instead.
  - Severity 1

Task 6
You decide to cook Quiche Maritime but would rather follow the recipe in steps. You realise your hands will get messy making this meal so you decide to follow the steps using voice control. Do so (up to step 3).

The user read the welcome screen description and decided to go with "Intermediate". He selected "Main Dish" from the Categories drop down and saw the Quiche Maritime recipe. He selected the recipe. He spotted the "Enable voice control" feature straight away, and enabled it. He saw from the pop-up that appeared that he could use the "next" voice command to switch steps, and managed to successfully call it a number of times. His voice was recognised without any major problems.

Problems:
- Lack of consistency between "go to start" button and "go to top" voice command - they are different.
  - Severity 3
- User would have preferred the steps to be left aligned.
  - Severity 1

Task 7

You and your friend are having a dispute about how the lemongrass should be cut when making beef burgers. Find out the proper way to do it.

User started by selected "intermediate" on the welcome screen, and then to the beef burgers on the home screen. From there she read the method and found that only the white part of the lemongrass should be used. Task accomplished.

Problems:
- Could not select text in the preparation panel.
  - Severity 1
- Could not scroll in the preparation panel so had to use the arrow keys or click next to be able to find the step that discussed lemongrass.
  - Severity 2

Task 8
It's a Sunday afternoon and all the shops are closed. You're really feeling either the Vanilla Slice or Chocolate Raspberry Brownies but know that your housemate, who's out, only has 2 eggs left. Find out which uses less.

User was greeted by the welcome screen and read the tooltips on each of the viewing preferences before choosing "intermediate". Both the *Brownies* and the *Vanilla slice* were present on the home screen so the user first clicked on the *Brownies*. From the recipes page, the user scrolled through the ingredients list and found that 4 eggs are needed for *Brownies*. The user seemed slightly confused by the buttons under the preparation panel and attempted to use the *previous* button to go back to the home screen. After clicking a few times the user then used the logo to return to the homepage and then click on the *Vanilla Slice* recipe link to confirm that the *Vanilla Slice* uses fewer eggs.

Problems:
- Used Preparation navigation buttons as buttons to navigate through the site
  - Severity 2

Problems summary and criticality
The results presented above highlight several important problems, related to the usability of the cooking application. Further to that, some general cosmetic issues with the presentation of the website were also identified. The issue, which was identified as most critical and which will be described next, in detail, relates to the clarity and level of detail presented on the individual categories and search results page (both of which share the same view).

Specifically: the third problem from task 3; the single problem from task 4; the second problem from task 5 all suggest that an individual category page lacks sufficient detail and flexibility in presentation adjustment. This lead to our test users often having to visit individual recipe pages in order to gather further information about a recipe, or gain further confidence in their recipe choice, which both confused them and delayed the completion of their tasks. As these problems were

related and averaged at a severe rating, we collaboratively decided that they were critical enough to address.

Consider *Figure 24*, which depicts the category page of the category "Main dish" prior to redesign. On a closer examination, the recipes enlisted lack some of the details that are contained on the individual recipe page and the home page - such as the recipe cooking time and the subcategories for the recipe. This information may be critical to the users in their decision making of what to prepare, and has lead to confusion among users in the usability evaluation. This problem was judged to be critical by the group, since it has significantly delayed task completion in some cases.

A further problem, as identified by the results, is that recipes in a category page are simply represented as a list, which is perhaps too plain for the needs of users. The current presentation may be sufficient for a category with a small number of recipes, but it would get very complex as more recipes are added. In particular, search results, which re-use the view of the categories, have the potential to become very complex and hard to navigate - especially as important information about recipes could be missing, such as the specifics of their category. This problem has also been identified critical by our group, and suggests that users may need additional functionality, allowing for refinement of those recipe lists. Some of the users even suggested directly that they wished such functionality were available.

Any redesign would need to address the problems below:

Problem 1: Not sufficient information/functionality on individual category page to support recipe discovery and exploration.
Problem 2: Recipe information available on other pages across the cooking application is lacking on the individual category page - there is a lack of consistency of recipe information across pages.

Redesign

This led to a redesign of the individual category page of the cooking application. Consider *Figure 25*, which presents the new design of the category page (the search results - *Figure 6* - share the same design and filtering capability). Two major improvements have been made in order to specifically address the two problems, which were identified in the previous section

To address **Problem 1**, the redesign has been adjusted to include much more detailed information about each individual recipe, that is enlisted on a category page. Specifically, all the categories and subcategories of each recipe have been added (missing from the previous design - *Figure 24*). This is especially useful, as a recipe on our website can be in multiple categories - "Beef burgers" are in "Main dish" and "Nut free", for example, as shown by the figure. Further to that, the cooking time of the recipe was also added. This information is also essential for the filtering functionality, which is discussed next. It ensures consistency across all of the pages of the cooking website - users can effectively find out all information about a recipe, as it is presented on the home page. Users also do not need to open individual recipe pages from a category page to find out the cooking time, for example (this was necessary in the previous design).

To address **Problem 2**, the redesign also includes a filtering panel (as illustrated in *Figure 25*), which allows users to refine the lists of recipes, present on a category page. The filtering panel alters the presentation of the recipes by hiding any recipes that don't match the currently selected filter. Available filters are: subcategories or other categories of recipes; cooking time; number of servings. Multiple filters can be applied simultaneously.

Filtering is an aid to website users, through allowing them to further downselect from the list of recipes, depending on their preferences and needs. Users do not have to scan each recipe's information, for example, to find out whether the recipe is nut free, or is under 30 minutes to cook; they can apply a filter. This allows users to focus on their main goal - selecting a recipe to cook - and makes the category page more generally usable.

5. Accessibility Evaluation

This section discusses the accessibility evaluation that was performed on the cooking application.

To start with, extensive **HTML5 validation** was performed on all of the views of the application, utilising the W3C HTML validator [9]. Some minor errors, mostly relating to incorrect placement of tags (such as <div> tags within an <a> tag), as well as extraneous closure tags and starting tags, were identified. These errors are most likely due to a developer error, and can be easily removed upon closer inspection of the code. **Semantic elements** are also used throughout the site. This removes the necessity of role attributes to aid screen readers in navigating the page and allows website crawlers to access the navigation bar for more informative results in search browsers.

Further to that, an **automated accessibility test** was executed on the site, through the Google Accessibility Developer tools, that are freely available as a Google Chrome Extension [10]. Although the guidelines that this extension follows are not clear. Its main component is the accessibility audit: a collection of audit rules checking for common accessibility problems. This extension identified some oversights on our part, mainly the omission of labels on some of the components of our website. The extension also found that the contrast on some of the text throughout the website does not have a high enough contrast ratio. This potentially means that some users, especially users with an eye impairment, may find the site difficult to read, which would in turn have a negative effect on the users' experience on our website.

The site was also tested without CSS. This was achieved by disabling the CSS from within the browser and can be done on most modern internet browser. It allowed us to examine the layout of the page and see if it was logical. There are two issues that were identified by this test. The images on the site are quite large; this means that when CSS is removed, the layout of the page is compromised, as a large amount of scrolling is required to move past the images. However, this can be easily rectified by limiting the size of newly submitted images of recipes on the website. The other issue is that the tooltips, which are used throughout site (*Figure 21* for example) in an attempt to make information available but not obtrusive, do not render properly without CSS.

Instead of appearing around the element that the tooltip is attached to, the text is rendered at the bottom of the page, with no clear link to the correct element.

Regarding **keyboard navigation** on our website, the layout allows for the user to be able to navigate through almost the entire site in a logical order. This was tested by pressing *tab* to navigate through the site and taking note of the order that elements were selected. An additional issue with the keyboard navigation, which was discovered, is that on navigation through the ingredients of a recipe on its individual page, one needs to press *tab* twice for every ingredient. This is caused by the check box being in focus and the text being a link to allow the user to cross the ingredient off the list. As previously mentioned in the report we also allow users to navigate through the method list using the arrow keys *(Figure 13)* and to put the search bar into focus by pressing the 's' key. This allows users to navigate through the entire site without having to tab to a specific section of page.

While there were issues in the accessibility of the site, there are many positive aspects that should be documented. All recipe images on the site contain alternative text and description that are informative and do not repeat information available elsewhere. This provides users that cannot see the image with a reasonable replacement for the picture that allows them to have a similar experience on the site as a user that can see the image.

There are examples of **feedforward** and **feedback** throughout the site. On the homepage, each image has a halo effect (as in *Figure 27*) to indicate that it is a link and therefore can be clicked to reach that recipe's individual page. This same effect has been used in the category and search page to indicate when the user is hovering on a particular recipe (as in *Figure 25* - Beef Burgers is highlighted). While the same effect has been used for slightly different reasons the feed forward has been maintained as the user has clear navigation buttons in the category and search pages. All links on the site will become a darker colour on hover, however this fails on previously visited pages.

Feedback is used in a few areas of the site. One example of this is the notification that appears when the user changes their viewing preference (*Figure 22*). It does not appear for long, but it confirms that the change has taken place and alongside the **Current Style** text in the header (*Figure 20*) it assures of the change. This is especially important when the user is not on the recipe page as there are no modifications to the view on any page except the preparation page. The current viewing preference of the user is shown both in the **Current Styles** header text and marked by a tick in the drop down menu (*Figure 21*) so the user is confident about the viewing preference that they are currently using.

Works Cited

[1] EllisLab. (2014) Codeigniter / EllisLab. [Online]. http://ellislab.com/codeigniter

[2] Twitter. (2014, Feb) Bootstrap. [Online]. http://getbootstrap.com

[3] GitHub. (2014) GitHub. [Online]. https://github.com

[4] CodeIgniter. (2012) Profiling Your Application: CodeIgniter User Guide. [Online].

http://ellislab.com/codeigniter/user-guide/general/profiling.html

[5] CodeIgniter. (2012) Style Guide: CodeIgniter User Guide. [Online].
http://ellislab.com/codeigniter/user-guide/general/styleguide.html

[6] Ulf Krautmacher. (2008, June) rastplatznotizen. [Online].
http://twobenches.wordpress.com/2008/06/05/don-normans-design-principles/

[7] Bruce Tognazzini. (2014, March) First Principles of Interaction Design | askTog. [Online].
http://asktog.com/atc/principles-of-interaction-design/

[8] U.S. Department of Health & Human Service. (2014, April) Guidelines | Usability.gov.
[Online]. http://guidelines.usability.gov

[9] W3C. (2009) The W3C Markup Validation Service. [Online]. http://validator.w3.org/

[10] Google Accessibility. (2014, April) Github. [Online].
https://github.com/GoogleChrome/accessibility-developer-tools

[11] Ben Shneiderman. Shneiderman's "Eight Golden Rules of Interface Design". [Online].
http://faculty.washington.edu/jtenenbg/courses/360/f04/sessions/schneidermanGoldenRules.html

Appendix A - Screenshots and diagrams
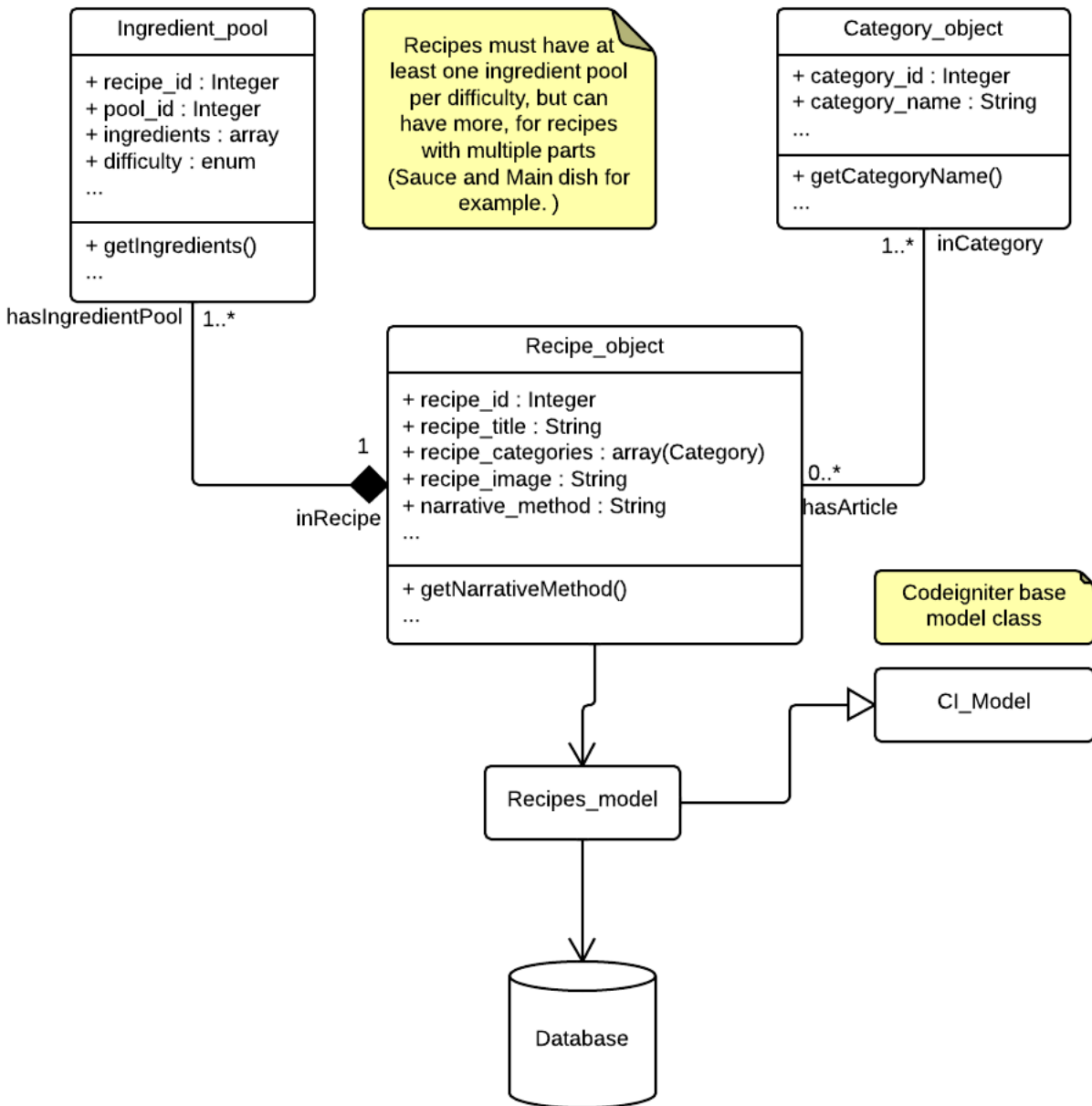
Figure 1: Data model diagram of cooking application.
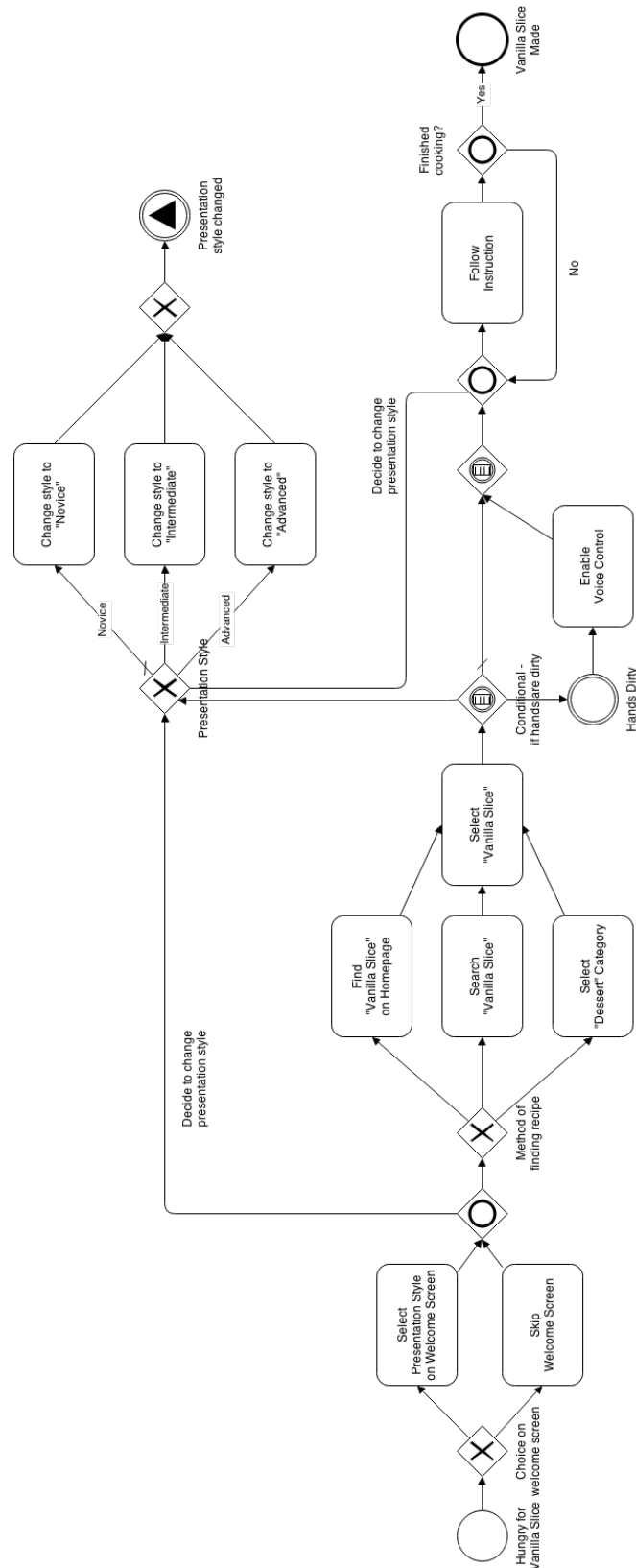
Figure 2: Vanilla Slice BPMN Diagram. Also available as a separate file.

Figure 3: Welcome page of cooking website.
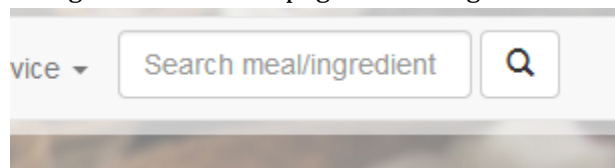
Figure 4: Search bar.
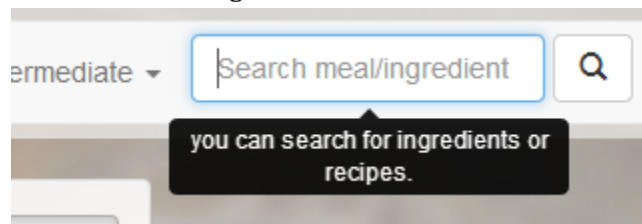
Figure 5: Search bar highlighted with a tooltip.
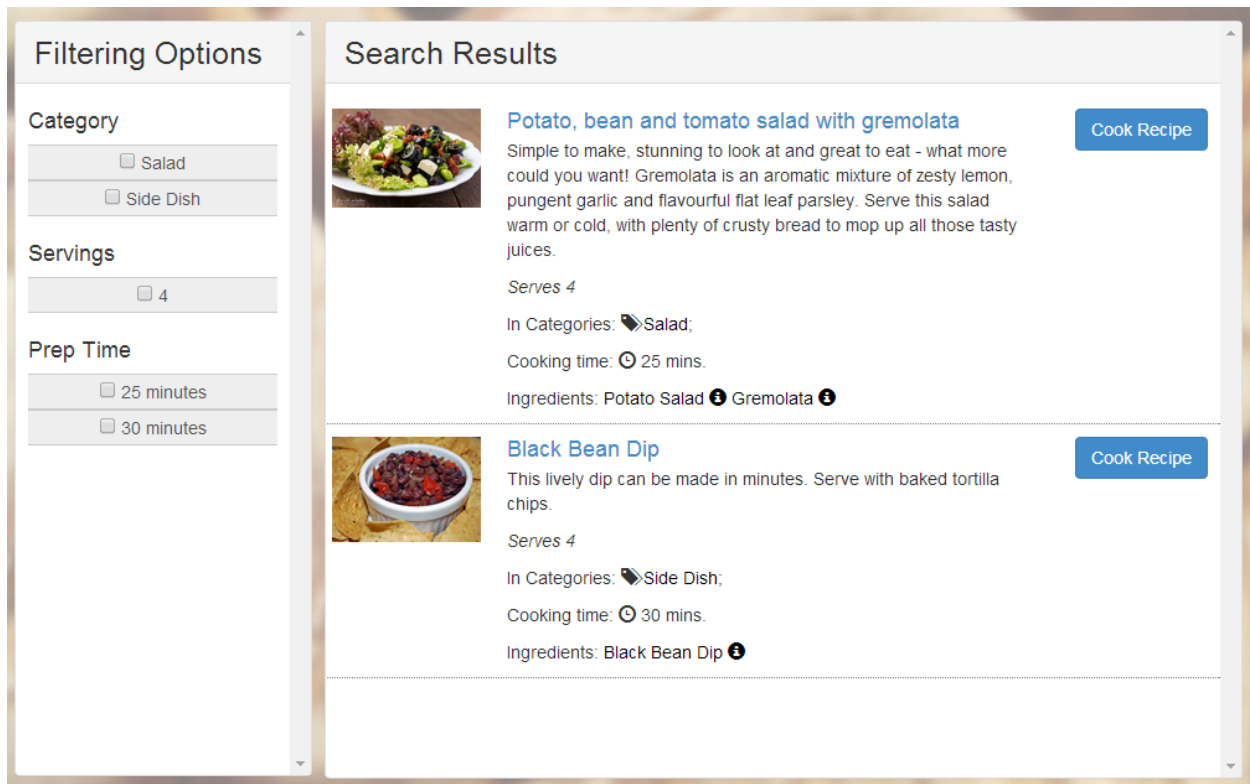
Figure 6: Search results from searching for "beans".
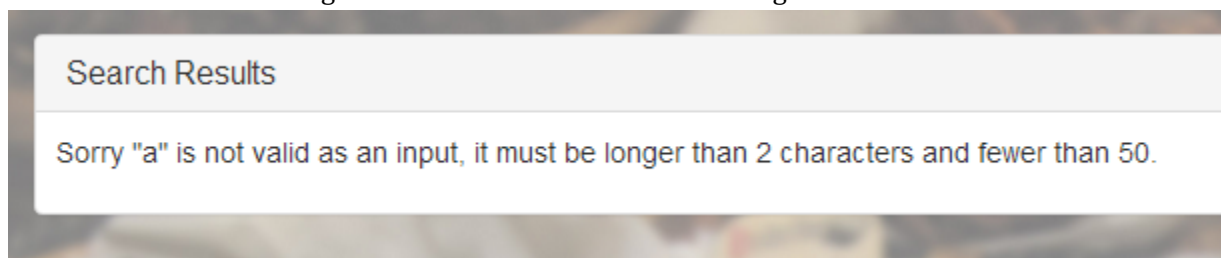
Figure 7: Search error handling.
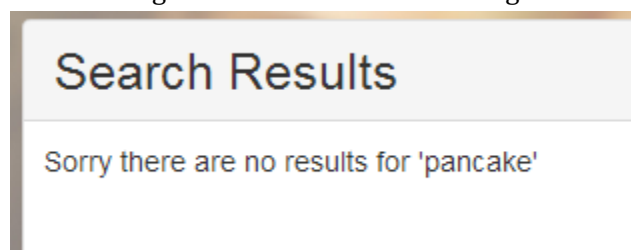
Figure 8: No search results.
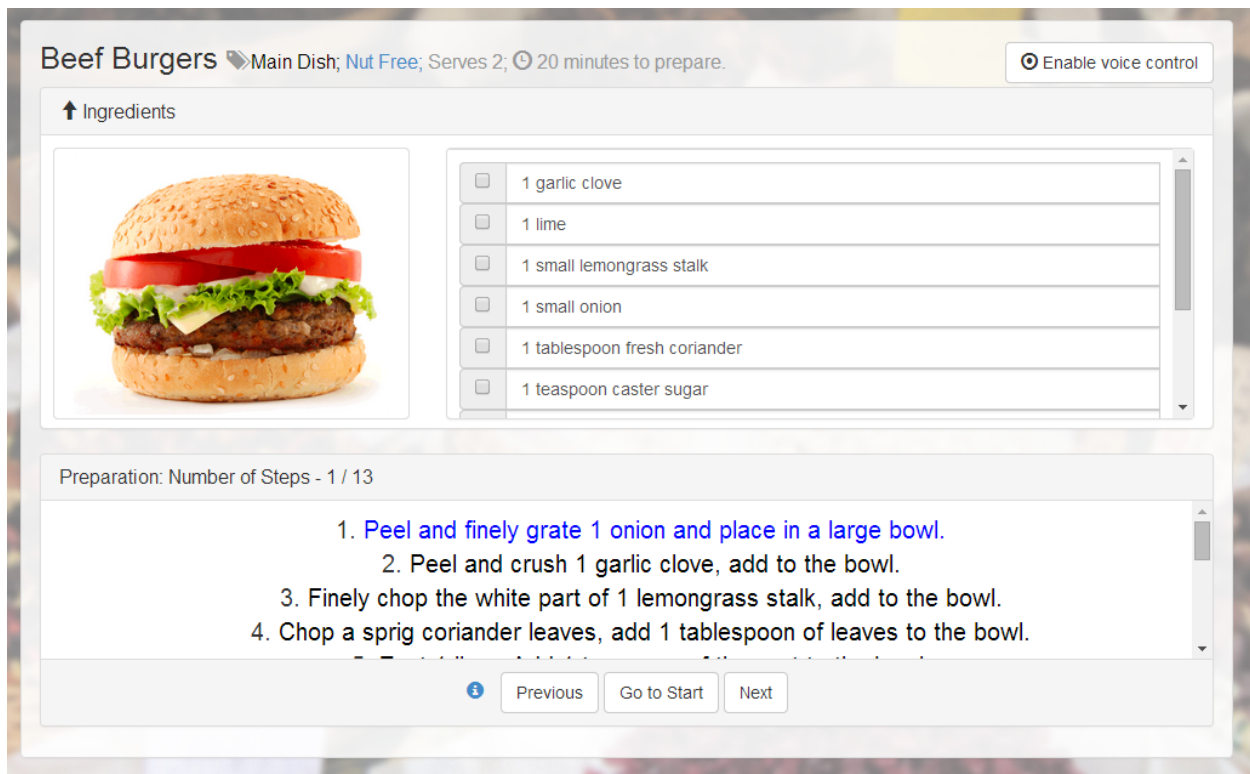
Figure 9: Beef burger recipe page with the "Novice" style selected.
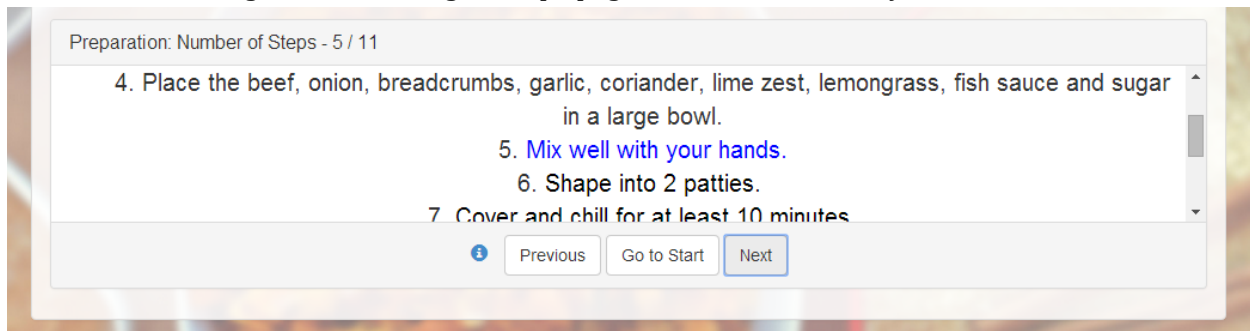
Figure 10: Current step in the preparation panel is always in the center.
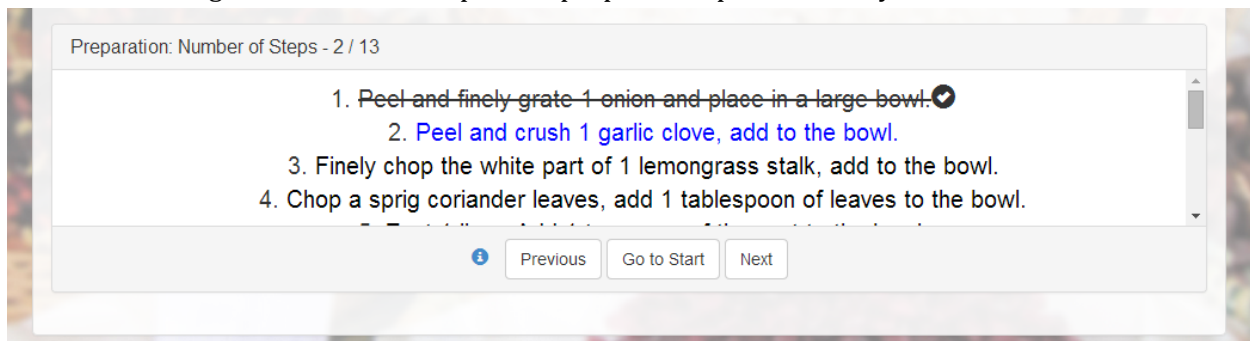
Figure 11: Crossing out steps in the step preparation panel.
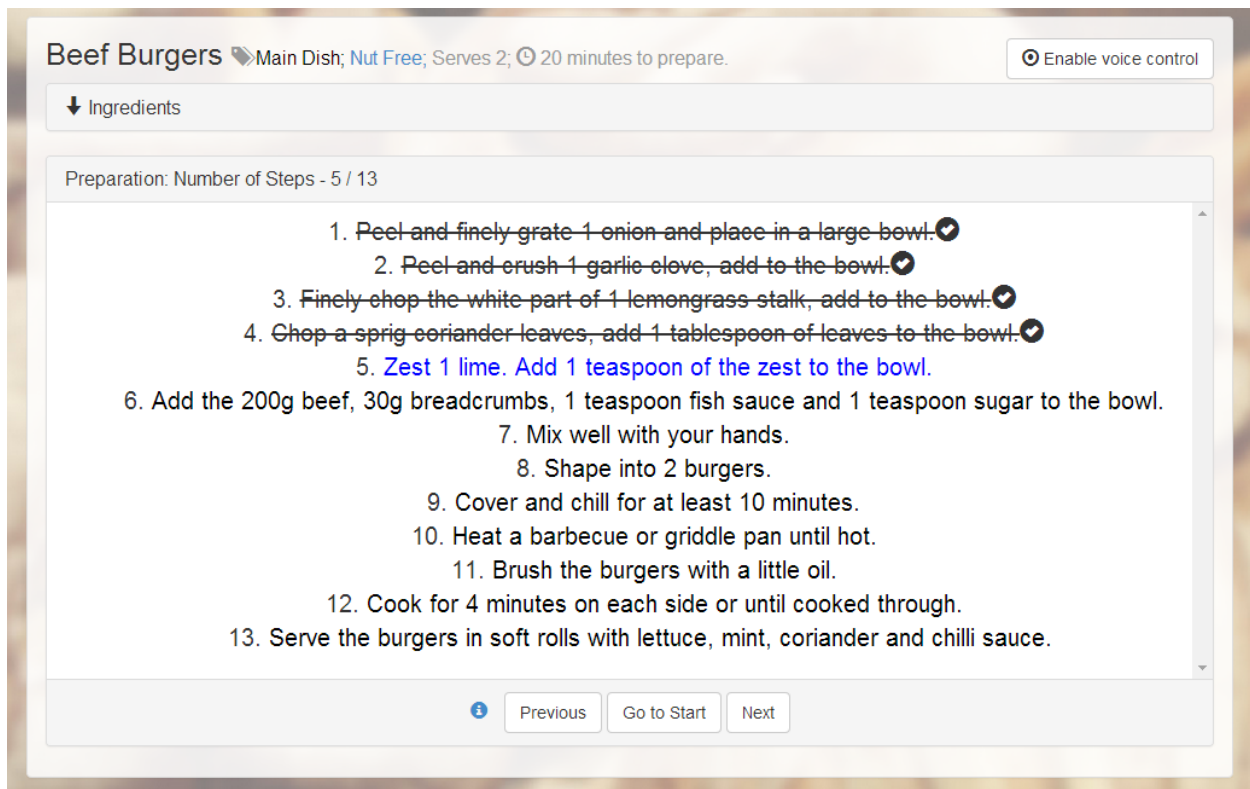
Figure 12: Hiding the ingredients panel to enlarge preparation panel.
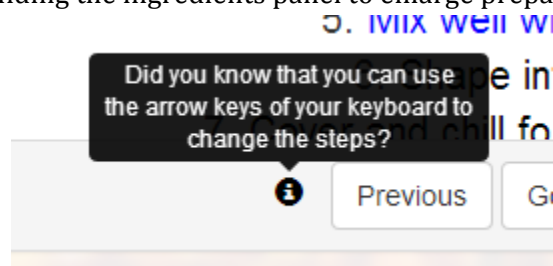
Figure 13: Information icon left of step controls provides hints to users.
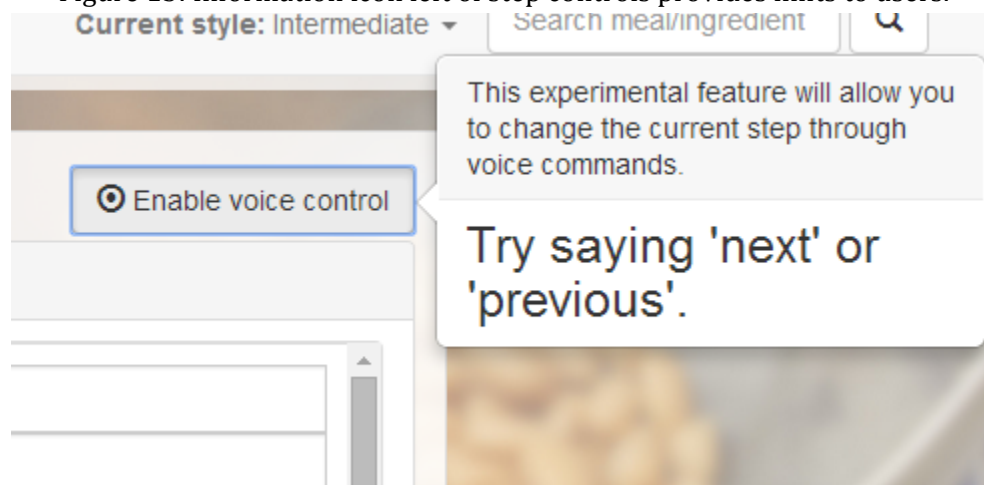
Figure 14: Enabling voice control - bubble with information.
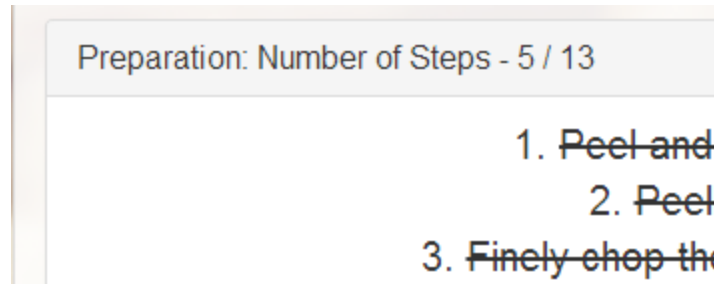
Figure 15: Display of recipe progress - steps accomplished and total number.
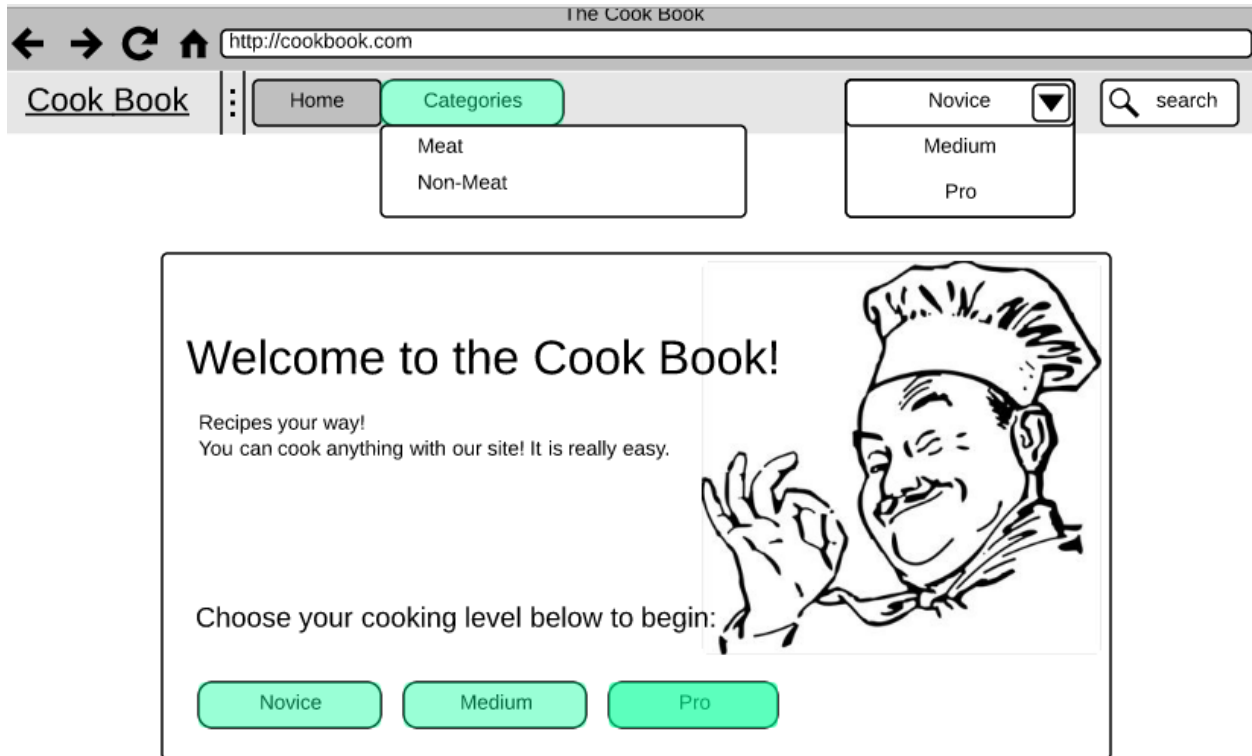
Figure 16: First wireframe of the welcome page of the cooking application.
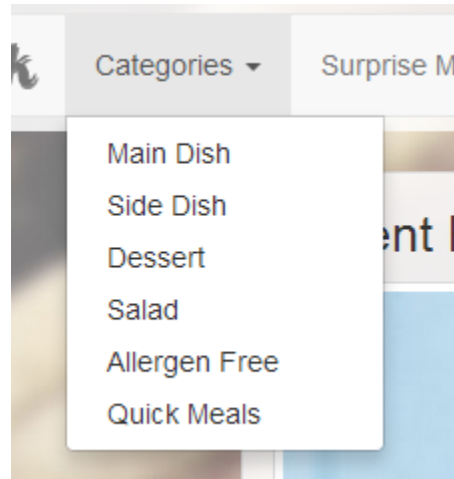
Figure 17: Categories drop-down menu.
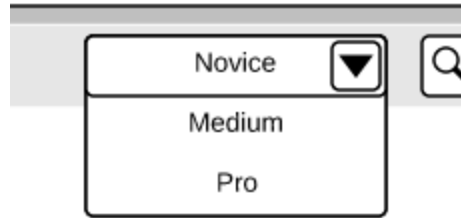
Fig. 18: Presentation drop-down menu of wireframe.
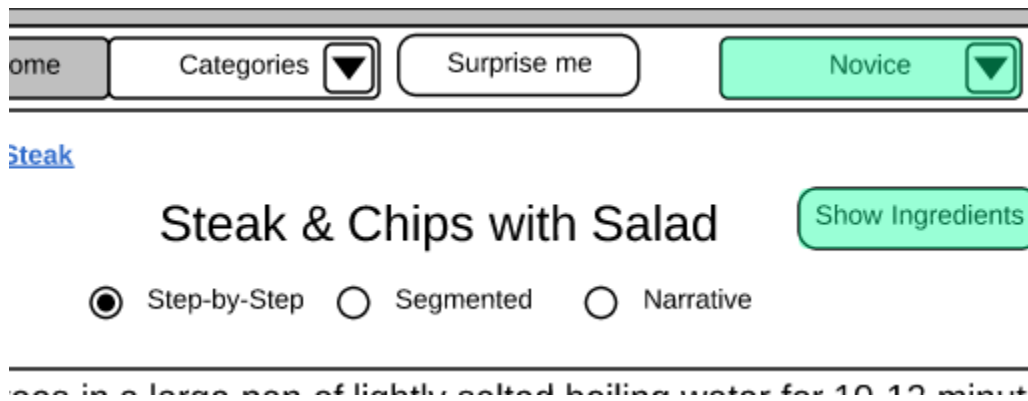
Fig. 19: Presentation style buttons on recipe page as well as the drop-down list with different names.
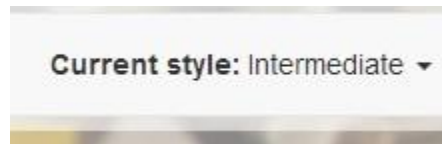
Figure 20: Presentation style dropdown after redesign.
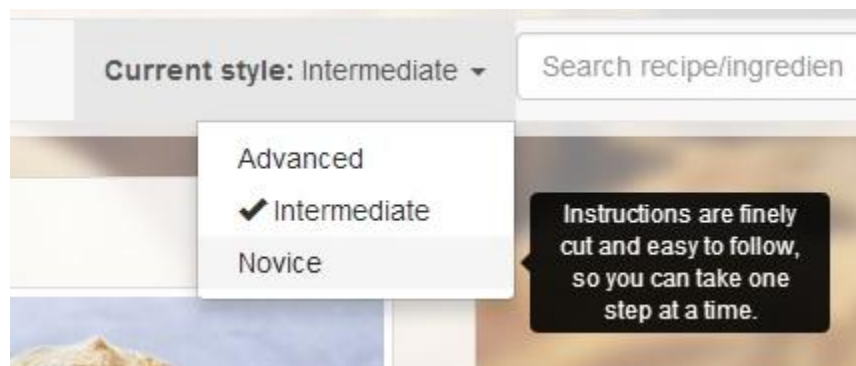
Figure 21: Tooltips for the presentation style dropdown.
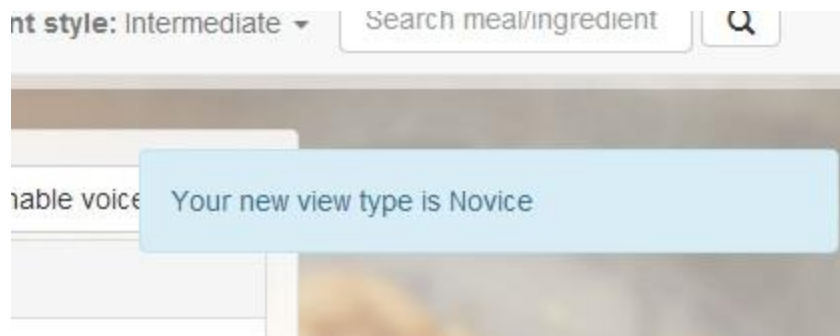
Figure 22: Presentation style change notification.
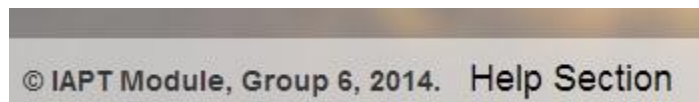
Figure 23: Footer of the cooking application with a link to the help section.
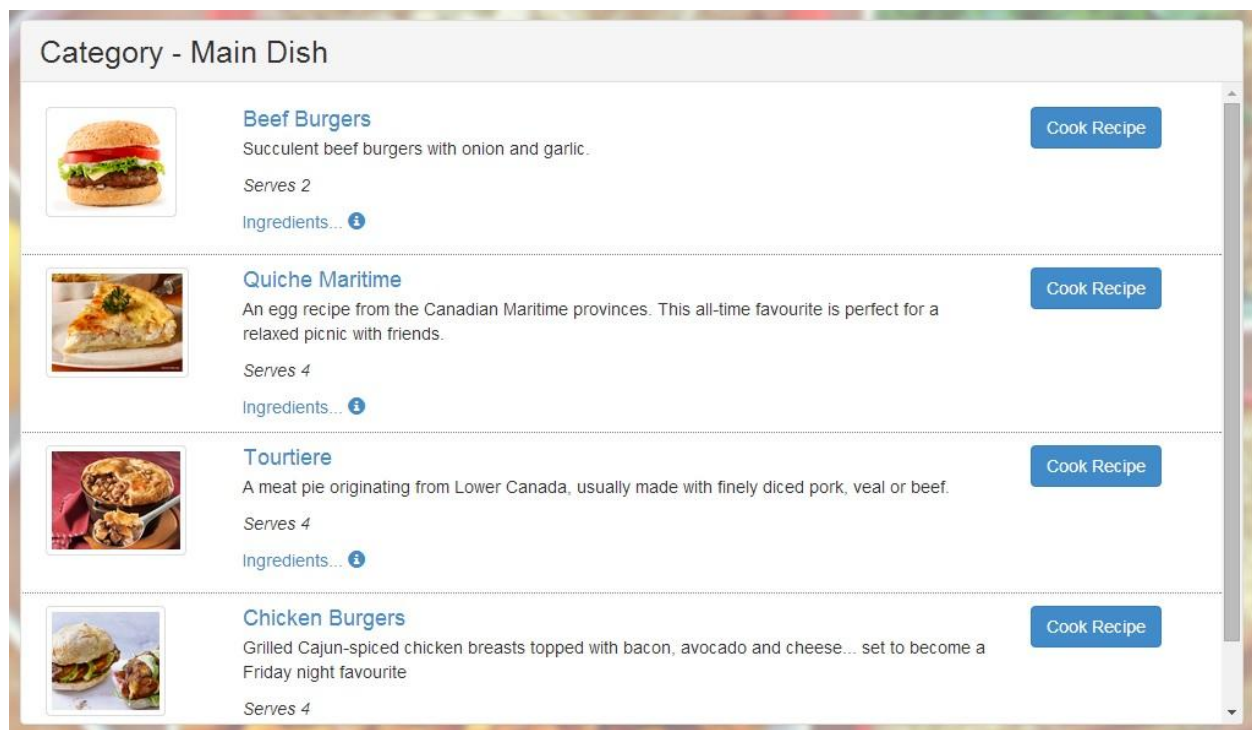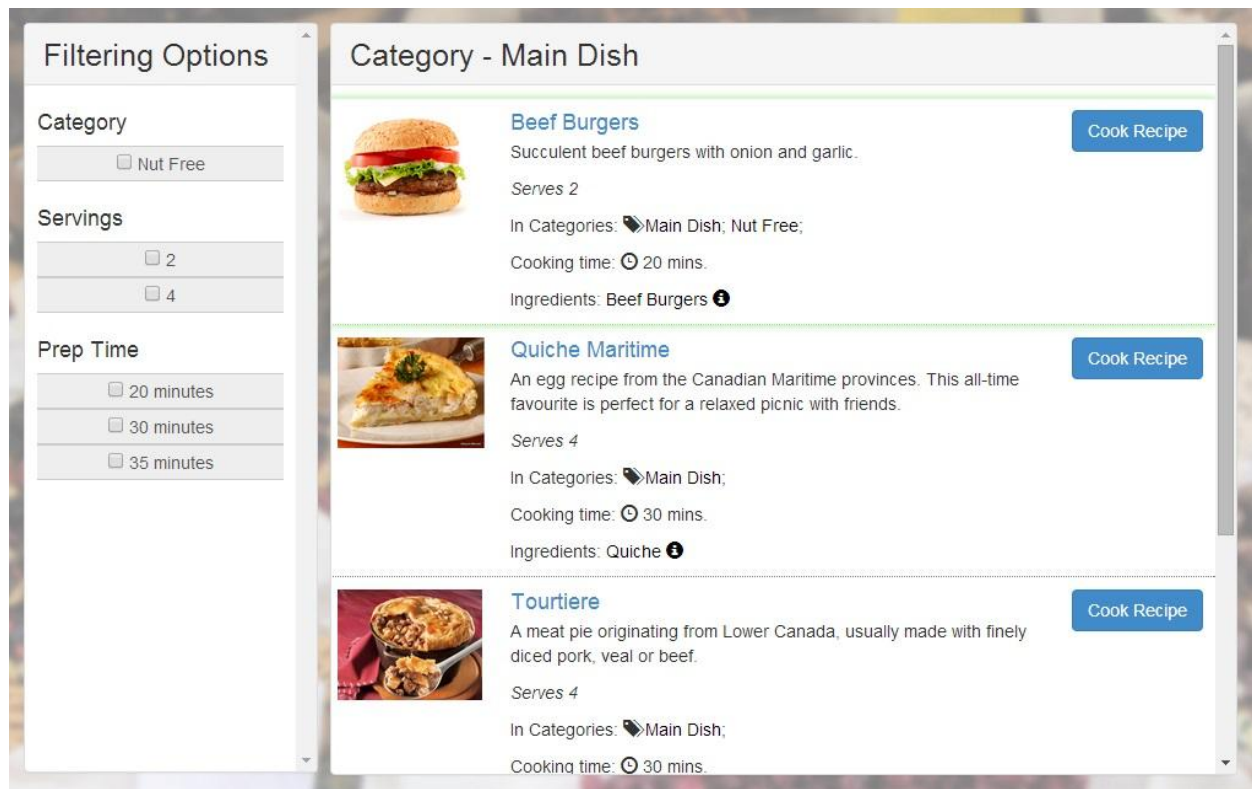

Figure 24: Category page prior to redesign.

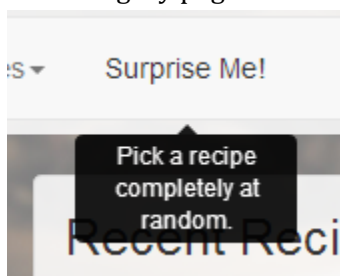Figure 25: Category page after redesign.

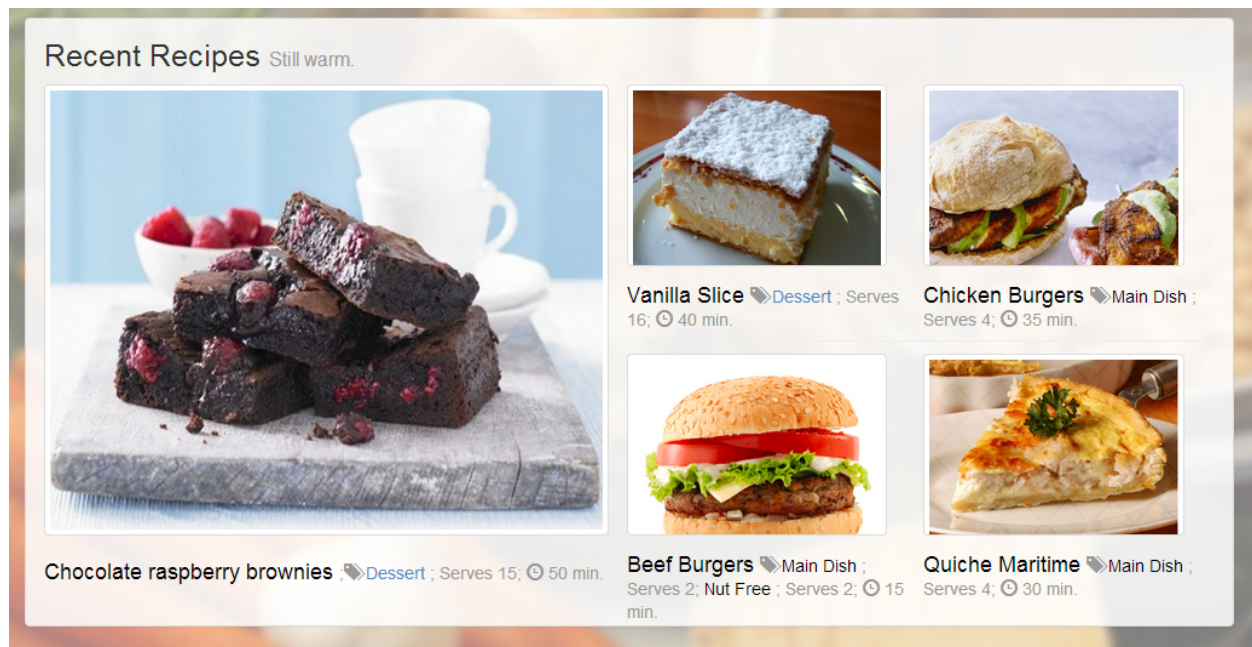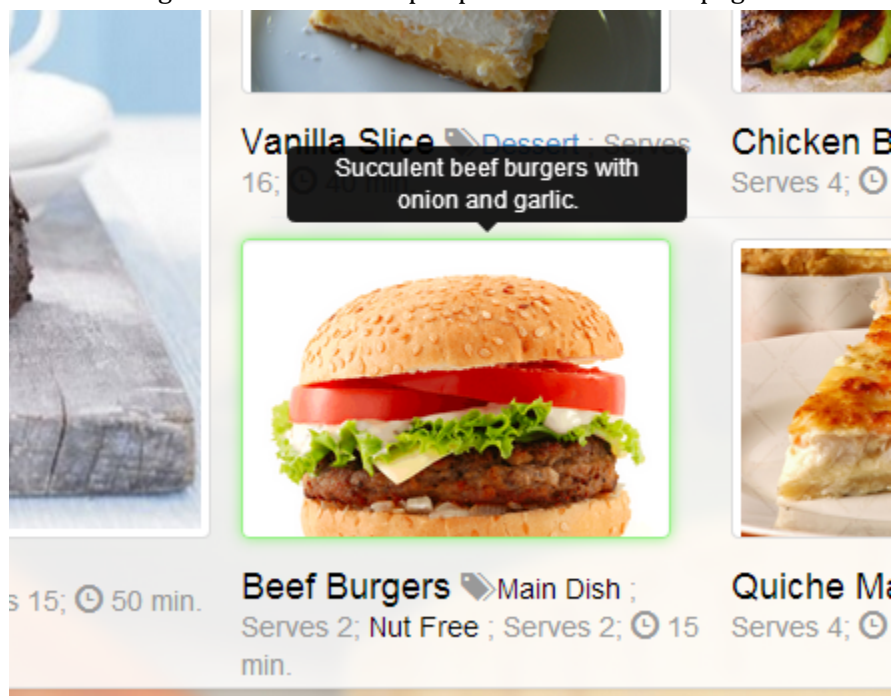Figure 26: Surprise Me button.

Figure 27: Recent recipes panel on the Home page.

Figure 28: Hovering over a recipe in the recent recipes panel.