User Experience Design (UX) Documentation for 'Exotic Fruits Quiz'
Five Planes of UX

Strategy

Brief:

“Create a web application that allows users to store and easily access cooking recipes “

Specifications:

- Create the backend code and frontend form to allow users to add new recipes to the site
- Create the backend code to group and summarise the recipes on the site, based on their attributes such as cuisine, country of origin, allergens, ingredients, etc. and a frontend page to show this summary, and make the categories clickable to drill down into a filtered view based on that category.
- Create the backend code to retrieve a list of recipes, filtered based on various criteria (e.g. allergens, cuisine, etc…) and order them based on some reasonable aspect (e.g. number of views or upvotes). Create a frontend page to display these, and to show some summary statistics around the list (e.g. number of matching recipes, number of new recipes.
- Create a detailed view for each recipes, that would just show all attributes for that recipe, and the full preparation instructions
- Allow for editing and deleting of the recipe records, either on separate pages, or built into the list/detail pages
- Optionally, you may choose to add basic user registration and authentication to the site. This can as simple as adding a username field to the recipe creation form, without a password (for this project only, this is not expected to be secure)

My target market could include anyone of any age group, but more specifically this application is made for people who enjoy cooking, learning new recipes and wanting to share their own recipes online. This type of user will want to be able to have their very own account where they can add, edit or delete any of their recipes, view recipes of different ingredients, categories and difficulties, and also explore different cuisines.

Scope

Features to be implemented:

- Login/Register form - this will allow the user to either log into an existing account or create a new account, inserting them into the database.
- Add/edit recipe form - this will allow logged in user to either create a new recipe (which will be entered into the database) or edit one of their own existing recipes (which will be updated).
- Delete recipe function - this will allow the logged in user to delete one of their own recipes, completely removing it from the database.
- Filter by category, cuisine, main ingredient or difficulty - this will allow the user to sort recipes by these filters, directing them to a page displaying these sorted recipes.
- Filter by cuisine - this will allow the user to be shown all recipes from a specific cuisine which they have chosen from the homepage.
- Logout - this will allow the user to log out of the current session, also providing them the option to return to the homepage or log back in.

Structure

Base:

- Header – will include brand logo and navigation links
- Footer – author tag

Home:

- Difficulty, category and ingredient filters underneath a welcome message. If user not logged in, there will be links for user to log in or register.
- Collection of cuisines displayed in grid layout, and when clicked, will reveal a description of the cuisine and a link to view recipes from that cuisine.

Login/Register:

- Form for login or register new account

All Recipes:

- Page which will display all recipes found in database, also letting the user know the total amount of recipes there are.
- Each recipe card will reveal information about the recipe and will give the user the option to view the recipe.

Get_recipe:

- Page which will display a single selected recipe, with details include category, difficulty, main ingredient, preparation time, cooking time, how many people the recipe serves and allergens within the recipe.
- Ingredients and method will be shown, along with an image.

My Recipes:

- Page which will display all recipes belonging to the user in the current session.
- Similar to 'All Recipes', each card will reveal details about the recipe, but will additionally provide the options to View, Edit or Delete the recipe.

Add Recipe:

- Form which will allow the user to create their own recipe. User will have to provide information such as cuisine, category, recipe name, recipe description, image URL, preparation time, cooking time, difficulty, main ingredient, how many servings, allergens, ingredients and method.
- Add recipe button which will insert recipe into the database.

Edit Recipe:

- Form which will allow the user to edit their own recipe.
- Update button which will update the recipe in the database.

Recipes by...

- Page which will display all recipes of a chosen filter (category, difficulty, cuisine or ingredient).
- Each card will reveal information about the recipe and a 'View Recipe' button.

Skeleton

Interface design:

The web application will need to have multiple pages with a simple, understandable yet eye-catching layout of features which will fulfil the specification points. There will need to be a base page which will set out the header and footer, keeping the layout of each page consistent and tidy. All other pages will extend from this base page with Flask. The web application will be taking a mobile-first approach and will therefore need to be responsive on all device screen sizes. This can be accomplished with the Materialize CSS Framework.

Navigation design: The header of each page will include the brand logo, 'The Cookery Cove', and navigation links allowing the user to access 'Home', 'All Recipes', 'My Recipes' and 'Add Recipe' pages (unless the user is not logged in whereby 'Add Recipe' and 'My Recipes' will not be available). The footer will include a link to my GitHub. On each page there will be a means of navigating to any page without difficulty.

Wireframes:
The wireframes for Desktop, Tablet and Mobile view can be found in /documentation.
These have been designed with wireframe.cc.

Surface

This web application is focused on an online cookbook. Whilst considering the background image to be recipe-related or cookbook-related, I have decided I will avoid this approach in order to keep the colour scheme from clashing with the recipe images. I will use the Materialize CSS Framework grid system to clearly present the recipes and will also use an off-white colour for the font to improve visibility. The user-interface will need to be easy to navigate – the navigation links within the header will be permanently visible with a fixed navbar, and for mobile users, there will be an easy-to-use side navigation bar. I will use no more than two fonts for the web application in order to avoid readability becoming complex.