SIT120 – ASSESSMENT 1: PROJECT PLAN, ANALYSIS AND DESIGN

PROJECT PROPOSAL

VICTORIA PRICILLA THENIA
219007874

Easy Recipe is a web app made for cooking. It has a collection of recipes of different cuisines for people to use. Users can see the list of recipes available in each cuisine category, search for recipes through the search field, see the current most viewed recipes, and can create an account to save recipes to look back at them. Easy Recipe is a responsive app so it can be accessed using different devices, including mobile.

Asset list

Type	Name	Uses and Elements	Vue	Detail
Page	index.html	Acts as a home page, consisting of navigation bar, search field, filter by elements function to find recipes from cuisine category, button images to go to the most viewed recipes, and internal JavaScript.	Inner Vue.JS: Search field – Vue form text input binding.	HTML file, static. Content: Navigation bar, interactive. Filter by elements function, responsive. Button images, interactive. Search field, responsive.
	contact.html	Business contact info, such as phone number, email address, and office address.		HTML file, static. Content: Navigation bar, interactive. Contact info (text), static.
	signin.html	Two-field form to enter email, password, and includes submit button. Includes a text with link to go to sign up page.		HTML file, static. Content: Navigation bar, interactive. Form, responsive. Text link, interactive.
	signup.html	Four-field and one selection field form to enter first name, last name, gender, email, password, and includes submit button. Includes a text with link to go to sign in page.		HTML file, static. Content: Navigation bar, interactive. Form, responsive. Text link, interactive.
	saved-recipe.html	Page for saved recipes. Includes a message asking users to sign in in order to save recipes.		HTML file, static. Content:

				Navigation bar, interactive. Message, static.
	dishname.html	Page for dish recipe consisting of <u>ingredients</u> and <u>steps</u>. Includes <u>dish image</u>, <u>back button</u> to go back to home page, <u>save button</u> to go to saved recipes page, <u>count button</u> to count how many times users have used the recipe, and <u>comment field</u> with <u>post button</u>.	Inner Vue.JS: Ingredients and Steps - list rendering. Count button – event handling. Comment field – form multiline text input binding. Post button – enter/leave transition.	HTML file, static. Content: Navigation bar, interactive. Ingredients, responsive. Steps, responsive. Back button, interactive. Save button, interactive. Count button, responsive. Comment field, responsive. Post button, responsive.
Other	style.css	Main stylesheet for all html files.		CSS file for all pages.
	dishname.jpg	Main image of the dish. Works as a button in the home page to go to the recipe page of the dish and as a static image in recipe page.		Image, static.

<u>Product purpose</u>

Target audience:

1. Adult Female and Male, 25 - 45 years old. Spends a lot of time at home, taking care of the house and family.
I'm targeting this group of people because I believe that they will find the web app useful for them to do home cooking for their family.

2. Student Female and Male, 18 - 24 years old. College/university students that live far away from their parents.
I'm also targeting this group because I believe that they will find the web app convenient since they can cut down cost by making home-made foods instead of buying from outside.

Creativity demonstration:

- Simple and clean design: I use neutral colours that blend well together, which makes the interface very clean-looking. The colours are bright and nice to look at giving a warm feeling to users. The colours aren't commonly used either, which helps add distinctiveness from other existing websites. I also make sure that the positioning of the elements is consistent from page to page and I don't put many things leaving empty areas/spaces, so that the interface doesn't look messy and heavy in the eyes. I set the elements/functions in an order that make sense and flow well, so starting with the most viewed function, search field, and then the filter by elements in the home page.

- Easy searching: I add filter by elements function for users to find recipes based on the type of cuisine. This allows users to just simply click on the cuisine category and the list of recipes in that category will show up in the same page without them being redirected to another page to see the results (faster and more convenient searching). This function makes it easier for users that want to cook multiple dishes from one particular cuisine. This function is located under the search field in the home page giving two options of searching to users.

- Unique feature: I come up with a function that allows users to keep track on how many times they have used a recipe. I have an idea to have this as a function since I feel that this can be helpful for them to know which recipe that they have used before. I consider this function as unique since I have never seen this function exists in similar apps.

- Easy to use: I use simple words for the headings and I make sure that the web's functions are easy to use and understand by just using common sense. I also make sure that navigation is easy in the web by setting a different colour for active button or page so that users know in which page they're at or what button that they clicked on.

<u>User stories</u>

No.	Story	Solution	Complexity	Priority
1	User wants to find recipes to cook multiple dishes from one type of cuisine.	Filter by elements function based on cuisine type/ category.	Moderate complexity.\n\nJavaScript to filter items by elements. Just a prototype, so no links added to results, results not interactive.\n\nElement formatting in style.css.	Highest
2	User wants to search for dish using a search field.	A search field user input function.	Moderate complexity.\n\nVue for text input field with text input binding (v-model) that shows the text entered on top of the field. Inner JavaScript in index.html and element formatting in style.css.	High
3	User wants to save recipes to look back at them.	Sign in/ sign up forms to save items.\n\nA save button in recipe page that links to HTML page for saved recipes.	Low complexity.\n\nJust a prototype, no actual sign in or sign up to occur after submit button of the form is clicked, no actual save function to occur, save button directs user to saved-recipe.html.\n\nSign in/sign up forms will be simple html forms. Form formatting in style.css.	High
4	User wants the web app to have a simple and clean design with consistent colours and interface across different devices.	CSS for interface and element formatting (responsive design).	Low complexity.\n\nstyle.css to be used for all pages containing style id and class for all element formatting.	High
5	User wants to keep track of how many times they have used a recipe.	A count button that increases in number if clicked.	Low complexity.\n\nVue component with event handling (v-on:click) for count button in *dishname*.html.	High
6	User wants to know the most viewed or popular recipes.	Images of popular dishes appearing on the top area of the home/ index page containing links to redirect users to the dish recipe	Low complexity.\n\nImage formatting in style.css. HTML 'a href' to add link and	Low

		page when the image is clicked.	'img src' to import *dishname*.jpg in index.html.	
7	User wants to make comments on the recipe.	A user input field with a button to post comment.	Moderate complexity. Just a prototype, no actual comments can be posted. Inner JavaScript. Element formatting in style.css. Vue multiline input field with text input binding that shows text entered below the field. Post button with simple Vue fade enter/leave transition for the message that tells user to sign in when button is clicked.	Low

Since I'm just creating a prototype of the project, not the full functional web app, there will be some advance extended coding that I will skip over.

UI design
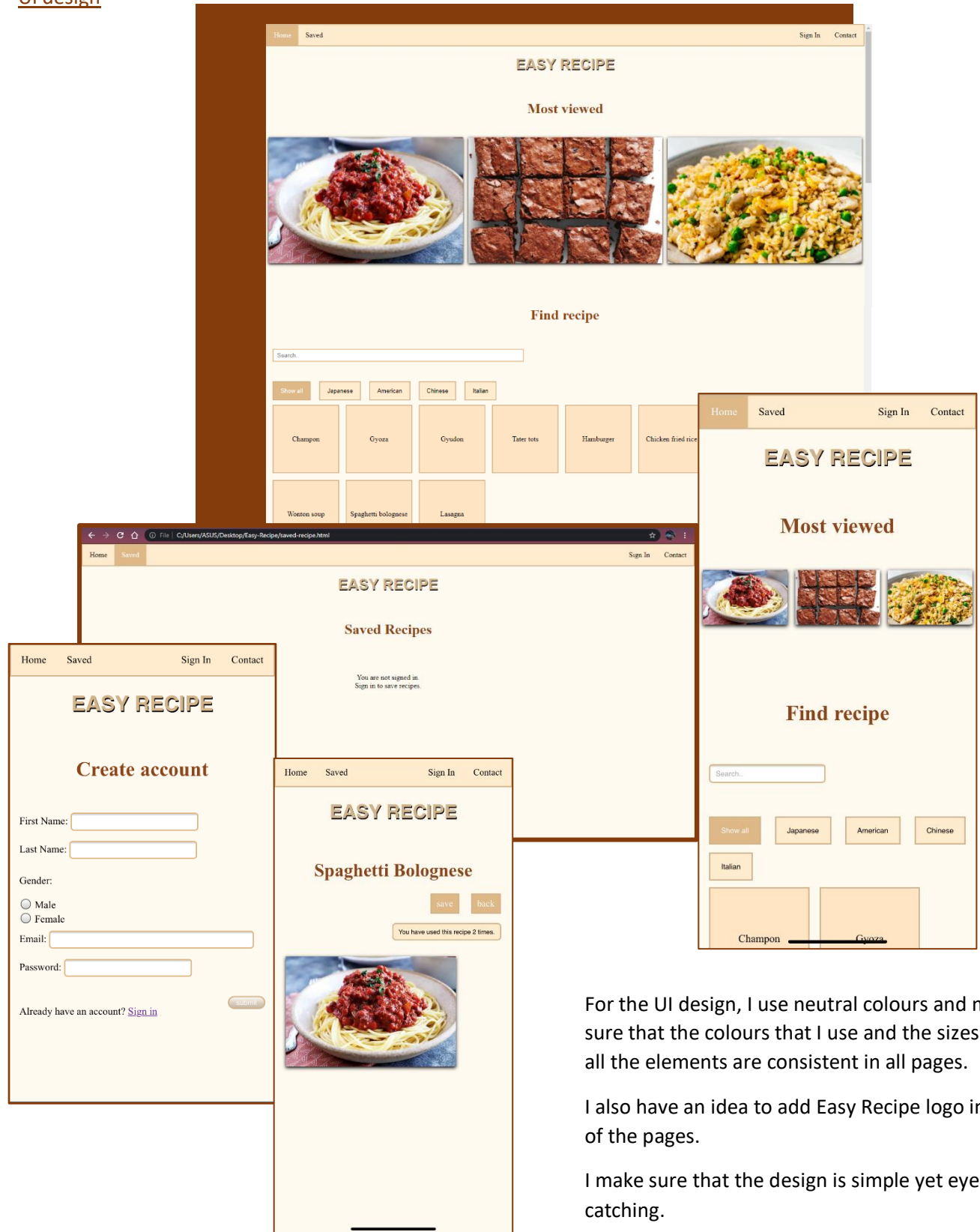
For the UI design, I use neutral colours and make sure that the colours that I use and the sizes of all the elements are consistent in all pages.

I also have an idea to add Easy Recipe logo in all of the pages.

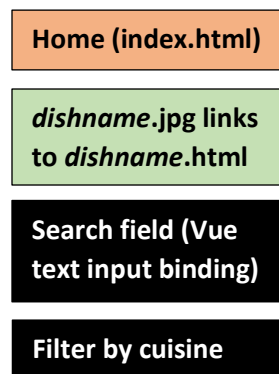I make sure that the design is simple yet eye-catching.

Project functionality
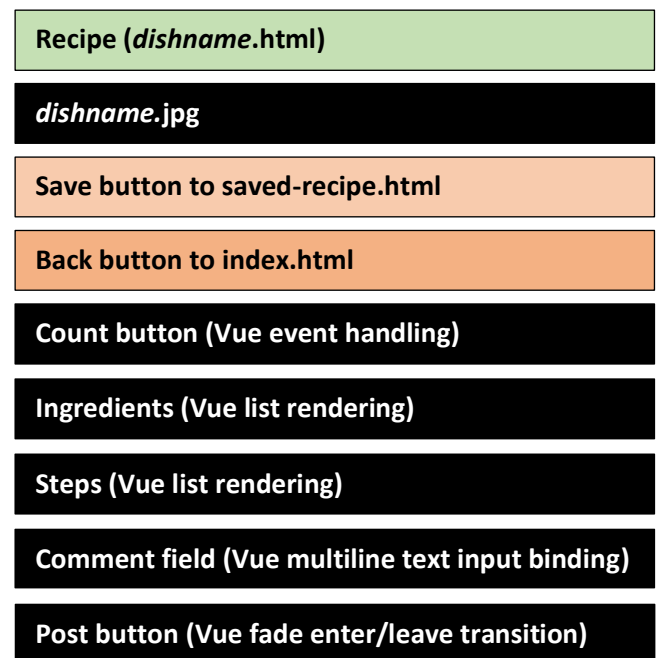
Navigation bar
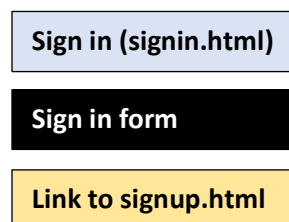
Navigation bar is present in all pages.

| Home | Saved | | Sign In | Contact |

Home page elements

Home (index.html)

dishname.jpg links to *dishname*.html

Search field (Vue text input binding)

Filter by cuisine

Sign up page elements

Sign up (signup.html)

Sign up form

Link to signin.html

Recipe page elements

Recipe (*dishname*.html)

dishname.jpg

Save button to saved-recipe.html

Back button to index.html

Count button (Vue event handling)

Ingredients (Vue list rendering)

Steps (Vue list rendering)

Comment field (Vue multiline text input binding)

Post button (Vue fade enter/leave transition)

Sign in page elements

Sign in (signin.html)

Sign in form

Link to signup.html

Contact page elements

Contact information (contact.html)

Contact information
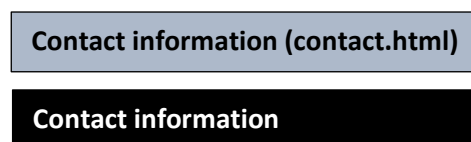
Saved recipes page elements

Saved recipes (saved-recipe.html)

A short message for users

These are the finalized components for Easy Recipe. More functionalities may be added.

<u>Project timeline</u>

No.	Task	Date
1	Vue.JS research.	Week 4 - 5
2	Product implementation.	Week 5 - 8
3	Adding additional functions (if necessary).	Week 8
4	Checking and finalizing work.	Week 9