

	Stack Side	Test Description	Input	Expected	Actual	Result	Parameter Explanation
1	Front	Search restaurants on Restaurant Page	Triple XXX, 9 Irish	Triple XXX restaurant and related restaurants from the area	Expected	Success	Make sure that it actually returns real restaurants
2	Front	Search restaurants on Restaurant Page	"gibberish"	No Results	Expected	Success	Wanted to test parameters with gibberish to see if it would return actual results
3	Front	Delete text from search bar on Restaurants Page		Most popular restaurants in area	Expected	Success	We never want the search to stop displaying results, so if the user clears the search just show the default yelp results for the area
4	Front/Back	Like restaurants from restaurants page	Click like button	Shows restaurants in favorite page	Nothing	Success	Testing the favorites page, making sure it works
5	Front	Dislike restaurants from restaurants page	Click dislike button	Shows showing restaurants in favorite page	Nothing	Success	Make sure user can't see restaurants that they've previously disliked
6	Front	Pull info from the USDA DB and show in dropdown	Froot Loops	Shows list of 50+ entries in the USDA for 'Froot Loops'	Expected	Success	Make sure that pulling from the USDA DB actually works using a real food item in it
7	Front/Back	Clear search bar and try adding something on entry page		Won't add anything and nothing is selected in the search bar	Expected	Success	Make sure a user can't add a null element to their meals
8	Front/Back	Save information from the entry page to the database	Click save data button	Data gets saved to database and can be reviewed via PostMan	Nothing	Failure	Make sure that user data is actually being saved
9	Front/Back	Ability to add an available recipe to a meal	Ev's Hamburgers	Add Ev's Hamburgers to a meal with the serving size 1	Expected	Success	Make sure that user stored recipes can be added to meals with no problem
10	Front	Make sure that all meals can be added to	Add ingredients/recipes to individual meals	Each meal can be added to without problem	Expected	Success	We want to make sure that users can be specific with how they plan out their day, so we allow them to add to different meals. This test demonstrates that.
11	Front	Make sure users can remove ingredients/recipes from a meal if needed	Remove a meal/recipe from daily entry page	Meal is removed, no longer displays	Expected	Success	We want to make sure that users can delete ingredients from meals if they wish in case they input bad data
12	Front/Back	Show recommendations on main page for restaurants the user might like	None	Show some kind of recommended restaurant based on location and user preference			
13	Front/Back	Show users their liked restaurants/recipes on page	Like a message on restaurants page	Show that restaurant in the favorites page	Nothing	Failure	Show that users can keep track of where they like/want to eat
14	Back	Check for valid meal object	meal object with valid fields	Return Success from API	Nothing	Success	Show that the validates parameters are configured correctly
15	Back	Check for invalid meal object	meal object with invalid fields	Return Error from API	Nothing	Failure	Show that the validates parameters are configured correctly
16	Back	Ensure Total Nutrition info is being added for meals	Recipe and ingredients entries	Valid total values for all nutrition information	Nothing	Success	This ensures that the triggers set on Ingredient and Recipe models function as intended
17	Back	Allow users to dislike restaurants	Restaurant and dislike	Restaurant given has dislike status given	Nothing	Success	This ensures that the user can dislike restaurants
18	Back/Front	di values in db are consistent with di graph	none	All values are correctly shown	shows 1 value	Failure	
19	Front/Back	Add recipe on the Add Recipe page and save to backend		Form successfully submits, backend shows recipe and ingredients are saved, page returns to front page	Expected	Success	
20	Back	Ensure recipes with specified allergen are not returned	gluten (id 76)	Recipes "15 minute Bread" and "Bread" should not be in the returned data	Expected	Success	Ensures that allergen recipes are filtered
21	Back	Ensure recipes with specified allergen are not returned	lactose (id 78)	Recipes with "Cheese" in the name are not returned	Expected	Success	Ensures that allergen recipes are filtered
22	Back	Ensure recipes with multiple allergens are not returned	gluten (id 76) and lactose (id 78)	Recipes with "Bread" or "Cheese" in the name are not returned	Expected	Success	Ensures that allergen recipes are filtered
23	Back	Ensure recipes with no allergens are not returned	nut (id 77)	All recipes will return since nut is not linked to any	Expected	Success	Ensures that allergen recipes are filtered
24	Back	Pull all recipes from the DB	Route index	All recipes from the DB will return	Expected	Success	Ensure that the index call works as expected in the recipes controller
25	Back	Pull specified recipe from DB	Route show	Specified recipe is returned when requested	Expected	Success	Ensure that the show call works as expected in the recipes controller
26	Back	Add new recipe	Route create	New recipe is added; reflected when a SHOW request is sent	Expected	Success	Ensure that the create call works as expected in the recipes controller
27	Back	Delete new recipe	Route delete	New recipe is deleted; reflected when a SHOW request is sent	Expected	Success	Ensure that the destroy call works as expected in the recipes controller
28	Back	Run empty search	Route search	All recipes are returned when this endpoint is hit	Expected	Success	Ensure that the search endpoint works in the recipes controller
29	Back	Run search filter	Route searchWithFilter param="This"	Only recipes with "This" in the name will be returned	Expected	Success	Ensure that the searchWithFilters endpoint works with the recipe controller
30	Front	Invalid Favorite on main page		Favorite score random restaurant	Expected	Success	Since the restaurants on the front page are randomized, selecting a random one should work
31	Front	Favorite more than one thing on insta favorite page	Favorite some random restaurant twice	Restaurant should only appear once	Shows dup	Failure	We just need to make sure that the same restaurant isn't added to the DB twice
32	Front	Favorite more than one thing on recipes page	Favorite random recipe twice	Recipe should only appear once	Shows dup	Failure	Same as above, we don't want the recipes to show twice
33	Front	Recipes page shows all available recipes to user	None	All recipes should show	Expected	Success	
34	Front	If no recipes exist, show blank page	None	No recipes should show	Expected	Success	Returns a successful query of nothing from the DB
35	Front	User can search recipes by name	Yum Yum Sauce	Should return the 'Yum Yum Sauce' recipe	Expected	Success	Chose a recipe with a distinct, specific name
36	Front	User can search recipes by partial name	Mac	Should return multiple recipes, all starting with or some variation of 'Mac' (Mac and cheese, mac's steak)	Expected	Success	Chose a specific title but something that multiple recipes may contain
37	Front	Sort recipes by preparation difficulty	Try the 'Somewhat challengin' option	Should return all recipes with the 'somewhat challenging' attribute	Expected	Success	Chose something where we could get two specific, succinct results with matching difficulties
38	Front	Sort by cook time	5:00, 10:00, 15:00	Should only return results with times exactly the same (15 minute pasta, 15 minute bread for 15:00)	Expected	Success	Chose something where we could get multiple succinct results with matching difficulties
39	Front	Sort by number of ingredients	1-2, 3-5, 6-8, 9+	Very Big Meal only shows for 9+, various meals show for number of ingredients	Expected	Success	Chose specific result that we know the exact number of ingredients for (very big meal = 9+)
40	Front	Only show user-owned recipes if they click recipe box	Click recipe ownership box	Should only show recipes owned by that user	Expected	Success	
41	Front	Show that search filter and dropdown filters work together	Type 'Very Big' and select 8+ on ingredients filter	Should still show 'Very Big Meal' since it falls under both those attributes	Expected	Success	Pick something specific with two attributes that we know (very big meal has many ingredients and a unique name)
42	Front	Show that search filter and dropdown filters fail effectively	Type 'Very Big' and select 1-2 on ingredients filter	Should show blank results since 'Very Big Meal' has 9 ingredients, not 1-2	Expected	Success	Pick something specific with two attributes that we know (very big meal has many ingredients and a unique name)
43	Front	Show that search filter and checkbox (ownership) filter works together	Type 'Very Big Meal' and select ownership checkbox	Should show 'Very Big Meal'	Expected	Success	Pick something specific with two attributes that we know (very big meal has many ingredients and a unique name)
44	Front	Show that search filter and ownership filter fails effectively	Type 'This is' and select ownership checkbox	Should only show the 'This is' that belong to us	Expected	Success	Pick something that we know will belong to us but also will have similar recipes that won't
45	Front	Show that all filters work at the same time with no lagging	Turn on all filters to find a recipe using all	Should show very few results, but some results where all the filters actually line up	Expected	Success	Just use all the filters in some way
46	Back	Disable all filters after they are all in use	Reset all filters	Should just return all recipes like normal	Expected	Success	
47	Back	Pull all allergens from DB	Route index	Should return all allergens	Expected	Success	Ensure that the index call works as expected in the allergen controller
48	Back	Create new Allergen	Route create	New allergen added; reflected when show request is sent	Expected	Success	Ensure that create works as expected in the allergen controller
49	Front	Show reviews for a restaurant on the main page	Click reviews on restaurant	Should show the 3 most relevant 'Yelp' reviews for that location	Expected	Success	
50	Front	Show reviews for another restaurant if a user clicks away	Click reviews on another restaurant from another	Should replace the 3 most relevant 'Yelp' reviews that location, removing the others	Expected	Success	
51	Front	Filter reviews by price	Select \$ in price dropdown	Should only show results with price of '\$'	Expected	Success	Triple XXX, Cafe Literato, etc. should all be examples of 'cheap' restaurants to compare to
52	Front	Switch from filtering by price to no filter	Switch away from \$ in dropdown	Should show results with whatever 'Yelp' pricing they want	Expected	Success	
53	Front	Filtering by price works with searching	Cafe Bistro' and \$\$\$	Should return Cafe Bistro and no other results	Expected	Success	Searched a specific result that we knew had a high pricing
54	Front	Filtering by price won't work with searching if the filters don't match up	Cafe Bistro and \$	No results	Expected	Success	Searched a specific restaurant that we knew was pricey but mispriced it on purpose
55	Back	Pull specified allergen from DB	Route show	Should return allergen matching allergen ID	Expected	Success	Ensure that the show call works as expected in the allergen controller
56	Front	Disable price filter and delete search term from restaurants filters	Disable price filter, remove search term	Should just show no results	Expected	Success	
57	Back	Pull all meals from DB	Route index	Should return all meals	Expected	Success	
58	Back	Pull meals associated with user from DB	Route show	Should return meals owned by user	Expected	Success	
59	Back	Destroy specified meal in DB	Route destroy	Should destroy meal; reflected when show request is sent	Expected	Success	
60	Back	Create new Allergen with bad info	Route create	Should return error	Expected	Success	
61	Back	Create Recipe with no user	Route create	Should return error	Expected	Success	
62	Back	Create recipe with no name	Route create	Should return error	Expected	Success	
63	Back	Create recipe with no instructions	Route create	Should return error	Expected	Success	
64	Front	Submit profile photo	Image	Should call profile-controller on backend	Expected	Success	
65	Back	Generate presigned AWS S3 bucket URL	User ID	Should create URL and pass back to frontend	Expected	Success	
66	Front	Put image to S3 url	Image	Uploads image to S3	Expected	Success	
67	Front	Weekly Graph max value changes with new and higher calorie count	Added new meal and be increase calorie count past	1 Max value should change	Expected	Success	
68	Front	Token expires after 20 minutes	Token	User should be kicked out to login page	Expected	Success	
69	Front	Weekly Graph daily calories are tracked (sun - sat)	Add new meals on different days	Each days value should increase	Expected	Success	
70	Front	Progress Page Daily Value graph change with new meals	Add new meals from DailyEntryPage	Daily Values change according to food item values added	Expected	Success	
71	Front	Daily Calorie pie graph used calories change with new meals	Add new meals from DailyEntryPage	Daily Calories calorie goals change according to item values added	Expected	Success	
72	Front	Different recommended restaurants are given with each page reload	Load or reload home page	Should load three new restaurants	Expected	Success	
73	Front	Allergen filter is on	Check allergy filter	Should filter out recipes that contain allergens	Nothing	Failure	
74	Front	Allergen filter is off	Uncheck allergy filter	All recipes should show	Expected	Success	
75	Front	Daily Nutrients button leads to Progress page	Click Daily Nutrients button on homepage	Leads to Progress page	Expected	Success	
76	Front	Profile button leads to users profile page	Click Profile button on homepage	Leads to Profile page	Expected	Success	
77	Front	Budget progress bar on home page increases	Input money spent on Today's Items total money spent	Bar should increase according to money input	Expected	Success	
78	Front	Budget used number on home page increases	Input money spent on Today's Items total money spent	Number should increase according to money input	Expected	Success	
79	Front	When clicked, Daily Value graph on homepage shows values	Click a bar on Daily Value graph	Value of bar should be shown in text	Expected	Success	