

[QA6D-27] QA6 DF Solo Project spoonacular Created: 03/Aug/18 Updated: 08/Aug/18 Due: 10/Aug/18

Status:	Open		
Project:	QA6 Dan		
Component/s:	None		
Affects Version/s:	None		
Fix Version/s:	None		
Type:	Epic	Priority:	Highest
Reporter:	Daniel Flanagan	Assignee:	Daniel Flanagan
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	1 week		
Time Spent:	Not Specified		
Original Estimate:	1 week		
Environment:	1. macOS 10.13.6 2. Google Chrome Version 68.0.3440.84 (Official Build) (64-bit) or newer 3. Microsoft VS Code Version 1.24.1 (1.24.1) 4. selenium-server-standalone-3.13.0 5. ChromeDriver 2.41 6. jdk-10.0.1_osx-x64 7. Postman Version 6.2.3 (6.2.3) 8. Nightwatch JS		
Epic Name:	QA6 DF spoonacular		

Description

QA6 Solo Project spoonacular

This Epic will focus on the 'Recipe Box' functionality within's spoonacular's app. API testing will also be conducted on spoonacular's API.

[QA6D-26] QA6 spoonacular 'Recipe' functionality Created: 03/Aug/18 Updated: 10/Aug/18

Status:	Open		
Project:	QA6 Dan		
Component/s:	None		
Affects Version/s:	None		
Fix Version/s:	None		

Type:	Test Plan	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		

Issue links:	Blocks			
	blocks	QA6D-28	QA6 spoonacular API testing	Open
	is blocked by	QA6D-31	Adding Recipe to 'Favorites' Box	Open
	is blocked by	QA6D-32	Move Recipe to different box. Delete ...	Open
	is blocked by	QA6D-33	User can search for Recipes in Boxes.	Open

	<p>is blocked by QA6D-30 User can add their own recipes and fr... Idea</p> <p>is blocked by QA6D-29 User's ability to create, name and de... Retired</p> <p>is blocked by QA6D-43 Execute QA6D-31 Passed</p> <p>is blocked by QA6D-44 Execute QA6D-32 Passed</p> <p>is blocked by QA6D-45 Execute QA6D-33 Passed</p> <p>is blocked by QA6D-46 Execute steps in QA6D-30 Passed</p>
Epic Link:	QA6 DF spoonacular
Introduction:	spoonacular's 'Recipe' functionality allows users to save food recipes in a 'Recipe Box'. These recipes can be the user's own or from recipes found with the 'Search spoonacular' function. Multiple 'Recipe Boxes' can be created, renamed, populated and deleted.
Test Items:	<ol style="list-style-type: none"> https://spoonacular.com/ Decision Table Control Flow Diagram Automation macOS 10.13.6 Google Chrome Version 68.0.3440.84 (Official Build) (64-bit) or newer Microsoft VS Code Version 1.24.1 (1.24.1) selenium-server-standalone-3.13.0 ChromeDriver 2.41 jdk-10.0.1_osx-x64 Postman Version 6.2.3 (6.2.3) Nightwatch JS
Features to be Tested:	<ol style="list-style-type: none"> User's ability to create, name and delete a 'Recipe Box' QA6D-29 RETIRED The User can add recipes from multiple sources including their own and from Internet. QA6D-30 IDEA You can add items to a 'Favorite' 'Recipe Box'. QA6D-31 OPEN Able to move a recipe between boxes or delete a recipe QA6D-32 OPEN User can search the 'Recipe Box' for saved recipes. QA6D-33 OPEN
Features Not to be Tested:	<p>Items NOT in the scope of this Test Plan</p> <ol style="list-style-type: none"> Search functionality will be used as a dependency of some test cases. Beyond its ability find a specific item Search will not be tested in depth. Meal Planner functionality will not be tested. Ingredient Listing within recipes will not be tested in depth. Nutritional information will not be taken into consideration. Actions performed when creating a recipe. API testing is conducted in QA6D-28 OPEN
Approach:	<ol style="list-style-type: none"> Recipes will be added, modified and removed as well as Recipe Boxes. This will first be done using manual methods but will also be automated. Control flow diagrams will be made as well as decision tables
Item Pass/Fail Criteria:	<ol style="list-style-type: none"> Creation of new 'Recipe Box' QA6D-29 RETIRED Changing a 'Recipe Box's name QA6D-29 RETIRED Add custom (your own recipe) recipe to 'Recipe Box' QA6D-30 IDEA Search for recipe using 'Search for spoonacular' and add to Recipe Box QA6D-30 IDEA Search for recipe using 'Search for spoonacular' and add to 'Favorites' box QA6D-31 OPEN Delete a recipe added from Internet in a 'Recipe Box' QA6D-32 OPEN Delete a custom added recipe from 'Recipe Box' QA6D-32 OPEN Able to Search for a saved Recipe inside of a 'Recipe box' QA6D-33 OPEN
Test Tasks:	<ol style="list-style-type: none"> Establish Functionality with manual exploratory testing. Create decision diagrams and tables Automate to provide baseline for future testing.
Environment Needs:	<ol style="list-style-type: none"> Chrome web browser Internet Access spoonacular.com
Risks & Contingencies:	<p>Product risks...</p> <ol style="list-style-type: none"> Underlining systems (ie Search) that allow a "Recipe Box" to be populated with recipe's will not tested. This dependency will greatly hamper Recipe' Box's functionality if Users are unable to search for a recipe and add it to a Recipe Box. Users might also be storing trade secrets or intellectual property in the forms of pictures or other assets of another party in the Recipe Box System. Also allowing users to upload their own pictures can open up legal ramifications even though the subject matter is food.

8/10/2018	JIRA
Other Supporting Info:	Deadline for project: August 10th 2018
Approvals:	Lead Instructor: AJ Larson Mentor: Andrew Call Spiritual Advisor: Max Bracken Life Coach: Vitaily Matiyev Astrologer: Bryen North

Comments
Comment by Daniel Flanagan [10/Aug/18]
Test Summary for API Testing
<ul style="list-style-type: none">• API testing first require familiraztion with API's access• API requires X-Mashape-Key to function• Review of API docs is imperative• Select 4 functions of API that show off its dynamic nature• Use Test data that is reflective of most use cases as well as some unexpected use cases to prove robustness.• API testing proved successful and no major defects found in 4 test cases.
Comment by Daniel Flanagan [10/Aug/18]
Test Summary for 'Recipe' Functionality
<ul style="list-style-type: none">• Users are able to Search for recpies using sponnaculars search engine.• They are then able to add the recpies found to thier Favories Recpie box or to a custom named recpie box• Users are able to rename, or delete recipe boxes• Users can move a recipe between boxes and delete a recipe• Users can search the recipe boxes for recipes they added from the internet or their own recipes.

[QA6D-33] User can search for Recipes in Boxes. Created: 08/Aug/18 Updated: 10/Aug/18

Status:	Open				
Project:	QA6 Dan				
Component/s:	None				
Affects Version/s:	None				
Fix Version/s:	None				
Type:	Test Case		Priority:	Medium	
Reporter:	Daniel Flanagan		Assignee:	Unassigned	
Resolution:	Unresolved		Votes:	0	
Labels:	None				
Σ Remaining Estimate:	Not Specified		Remaining Estimate:	Not Specified	
Σ Time Spent:	Not Specified		Time Spent:	Not Specified	
Σ Original Estimate:	Not Specified		Original Estimate:	Not Specified	
Issue links:	Blocks blocks QA6D-26 QA6 spoonacular 'Recipe' functionality Open				
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-45	Execute QA6D-33	Test Execution	Passed	
Epic Link:	QA6 DF spoonacular				
Test Preconditions:	<div>1. Have spoonacular logged in</div> <div>2. Use Google Chrome to perform testing</div> <div>3. Have test data if applicable.</div>				
Test Script:	<div>1. Start after logging into spoonacular</div> <div>2. Click on recipes from the main navigation bar</div> <div>3. At the recipes screen, you will see a field that states "Search my Boxes: Cheese, Italian, dinner, vegetarian</div> <div>4. Type Chimichangs into the search field</div> <div>5. If the previous item was added you should see Chimichanags listed.</div>				

8/10/2018	JIRA
	6. Create your own test recipe. Name it test 7. Try and search for your own recipe using the phrase TEST 8. You should also see your own recipe appear.
Test Postconditions:	User is able to search for recipes within different recipe boxes. They can also search for their own recipes.

Description
Using the search function within 'Recipes' users can search for a recipe that is saved in a box from the main 'Recipe' screen.

[QA6D-32] Move Recipe to different box. Delete Recipe. Created: 08/Aug/18 Updated: 10/Aug/18	
Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Issue links:	Blocks blocks QA6D-26 QA6 spoonacular 'Recipe' functionality Open				
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-44	Execute QA6D-32	Test Execution	Passed	
Epic Link:	QA6 DF spoonacular				
Test Preconditions:	1. Have spoonacular logged in 2. Use Google Chrome to perform testing 3. Have test data if applicable.				
Test Script:	1. Navigate to 'Recipes' section. 2. Create a new Recipe box called Drinks 3. Click on the coffee recipe. 4. You will see a rebox button at the bottom of the card . 5. Click on the rebox button and change the box to Drinks. 6. Coffee should be switched to Drinks 7. Now to delete the Coffee recipe 8. Click on the Coffee recipe, then edit recipe 9. Scroll down to the bottom of the recipe 10. Click on the Delete this Recipe button. 11. You will see pop up confirming action 12. You will be directed back to User profile page				
Test Postconditions:	You are able to change the box of a recipe and delete a recipe.				

Description
User is able to move a recipe to a different Recipe box. You can also delete a recipe from a box.

[QA6D-31] Adding Recipe to 'Favorites' Box Created: 08/Aug/18 Updated: 10/Aug/18

Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Issue links:	Blocks				
	blocks	QA6D-26	QA6 spoonacular 'Recipe' functionality		Open
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-43	Execute QA6D-31	Test Execution	Passed	
Epic Link:	QA6 DF spoonacular				
Test Preconditions:	<div>1. Have spoonacular logged in</div> <div>2. Use Google Chrome to perform testing</div> <div>3. Have test data if applicable.</div>				
Test Script:	<div>1. Search for a recipe containing the word 'Tacos'</div> <div>2. Select a recipe containing Tacos</div> <div>3. Click on the Save to Recipe Box</div> <div>4. Select the Favorites box, it should be there already</div> <div>5. Recipe should now be saved to favorites.</div> <div>6. Go to Recipes, select Coffee, the recipe we added previously.</div> <div>7. Select Coffee, then Rebox, click Favorites</div> <div>8. Coffee should now be under Favorite recipies</div>				
Test Postconditions:	User should be able to add recipes to the Favorites box from either recipes found on the internet or their own recipes.				

Description

The Favorites Box is a default 'Recipe Box that contains Recipe's users have decided to favorite. User is able to Search for a recipe using 'Search for spoonacular' and add to 'Favorites' box. User can also favorite their own recipe.

[QA6D-30] User can add their own recipes and from Internet. Created: 08/Aug/18 Updated: 10/Aug/18

Status:	Idea
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0

8/10/2018

JIRA

Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Issue links:	Blocks blocks QA6D-26 QA6 spoonacular 'Recipe' functionality Open				
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-46	Execute steps in QA6D-30	Test Execution	Passed	
Epic Link:	QA6 DF spoonacular				
Test Preconditions:	<div>1. Have spoonacular.com loaded into Google Chrome</div> <div>2. Login info user: df4029 pass: foodisgood</div> <div>3. Have test Data available.</div>				
Test Script:	<div>1. Log into spoonacular</div> <div>2. Go to search bar at the top of the page.</div> <div>3. Search for Chimichangas.</div> <div>4. Select the first result for Chimichangas.</div> <div>5. There will be an add to 'Recipe Box' button at the top of the Recipe</div> <div>6. Click to add to 'Mexican' 'Recipe Box'</div> <div>7. Now add your own recipe to a box</div> <div>8. At the navigation bar, click on recipe once, then add a recipe.</div> <div>9. Create a recppe named 'Coffee'</div> <div>10. Fill out the recpie form. This feature is asumed to be functioning normally.</div> <div>11. Change box to 'Mexican' save recipe.</div> <div>12. Coffee should be in two boxes, 'My Own' box, and 'Mexican'.</div>				
Test Postconditions:	<div>1. Users should be able to find a recipe online and add it to a recipe box</div> <div>2. Users should be able to create a recipe and add it to a 'Recipe Box'</div>				

Description

Test Case concerns User's ability to
<div>1. Add 'Recipes' that the user has searched for on spoonacular to a 'Recipe Box'</div> <div>2. Users can add their own recipes to a 'Recipe Box'</div>

[QA6D-29] User's ability to create, name and delete a 'Recipe Box' <small>Created: 07/Aug/18 Updated: 10/Aug/18</small>	
Status:	Retired
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		

Issue links:	Blocks blocks QA6D-26 QA6 spoonacular 'Recipe' functionality Open			
Epic Link:	QA6 DF spoonacular			

8/10/2018JIRA

Test Preconditions:	1. Have spoonacular.com loaded into Google Chrome 2. Login info user: df4029 pass: foodisgood
Test Script:	1. Login to spoonacular with provided username and password 2. Using Navigation bar, go to 'Recipes' 3. Create a new 'Recipe Box', by clicking on the plus icon to the left of the existing recipe boxes 4. Name that 'Recipe Box'. 'Rbox1' 5. Change the name of the 'Recipe Box' to 'French Cooking' by clicking on the recipe box , then clicking rename box 6. Delete the 'Recipe Box' 'French Cooking by clicking on the box and clicking the delete icon.
Test Postconditions:	1. User should be able to <ul style="list-style-type: none">Create and rename a 'Recipe Box'Delete a 'Recipe Box'

Description
Test Case concerns User's ability to <ul style="list-style-type: none">1. Create a new 'Recipe Box',2. Name that 'Recipe Box'.3. Change the name of the 'Recipe Box'.4. Delete a Recipe Box.

[QA6D-37] API Get: Random Recipe Created: 08/Aug/18 Updated: 10/Aug/18	
Status:	Retired
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Attachments:

response (2).json

Issue links:

Blocks

blocks

QA6D-28

QA6 spoonacular API testing

Open

is blocked by

QA6D-42

Execute QA6D-37

Passed

Sub-tasks:

Key	Summary	Type	Status	Assignee
QA6D-42	Execute QA6D-37	Test Execution	Passed	

Epic Link:

QA6 DF spoonacular

Test Preconditions:

1. Have Postman Open.

2. Have Test Data Data available.

3. API Key "X-Mashape-Key", "oaNmJs8gdImshKNbXOPxLKHG68vOp1MM9h7jsn6WvTZ0Zn4eqC"

4. API Reference documentation [here](#)

Test Script:

1. Open postman and confirm URL is correct

2. Run Get: Random Recipe command to get random recipe

3. Type value 'dinner' into Tag string

4. See attached JSON for expected results.

Test Postconditions:

See attached JSON for expected results.

8/10/2018JIRA

Description

Using Postman, confirm functioning of Get: Random Recipe command is able to present a random recipe to the user.

[QA6D-36] API Get: Extract Recipe from Website Created: 08/Aug/18 Updated: 10/Aug/18

Status:	Retired		
Project:	QA6 Dan		
Component/s:	None		
Affects Version/s:	None		
Fix Version/s:	None		

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Attachments:

response.json

Issue links:

Blocks

blocksQA6D-28QA6 spoonacular API testingOpen

is blocked byQA6D-44Execute QA6D-36Passed

Sub-tasks:

Key	Summary	Type	Status	Assignee
QA6D-41	Execute QA6D-36	Test Execution	Passed	

Epic Link:

QA6 DF spoonacular

Test Preconditions:

1. Have Postman Open.

2. Have Test Data Data available.

3. API Key "X-Mashape-Key", "oaNmJs8gdImshKNbXOPxLKHG68vOp1MM9h7jsn6WvtZ0Zn4eqC"

4. API Reference documentation [here](#)

Test Script:

1. Open postman and confirm URL is correct

2. Run Get: Extract Recipe from Website to extract recipe from external website

3. Extract recipe from this URL

Test Postconditions:

See attached JSON for proper response.

Description

Using Postman, confirm functioning of Get: Extract Recipe from Website command is able to extrapolate food items out of a body of text.

[QA6D-35] API Post: Classify a Grocery Product Created: 08/Aug/18 Updated: 10/Aug/18

Status:	Retired
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

8/10/2018

JIRA

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Issue links:	Blocks blocks QA6D-28 QA6 spoonacular API testing Open				
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-40	Execute QA6D-35	Test Execution	Passed	
Epic Link:	QA6 DF spoonacular				
Test Preconditions:	<div>1. Have Postman Open.</div> <div>2. Have Test Data Data available.</div> <div>3. API Key "X-Mashape-Key", "oaNmJs8gdImshKNbXOPxLKHG68vOp1MM9h7jsn6WvTZ0Zn4eqC"</div> <div>4. API Reference documentation here</div>				
Test Script:	<div>1. Open postman and confirm URL is correct</div> <div>2. Run POST Request to classify food grocery items</div> <div>3. "title": "Kellog Rice Krispie Treat", "upc": "", "plu_code": ""</div> <div>4. "title": "Dole Fruit Cup", "upc": "", "plu_code": ""</div> <div>5. "title": "Pineapple", "upc": "", "plu_code": "" }</div> <div>6. Confirm API has parsed and sorted grocery items</div>				
Test Postconditions:	<div>1. Confirm API returned data</div> <div>2. "breadcrumbs": ["rice crispies treat", "dessert"], "category": "rice_crispy_treats", "usdaCode": null, "image": "https://spoonacular.com/cdn/ingredients_100x100/port.jpg", "cleanTitle": "Kellog Rice Krispies Treat" }</div> <div>3. "breadcrumbs": ["fruit cups", "snack"], "category": "fruit_cups", "usdaCode": null, "image": null, "cleanTitle": "Dole Fruit Cup"</div> <div>4. "breadcrumbs": ["pineapple", "tropical fruit", "fruits", "flavor"], "category": "pineapple", "usdaCode": 9266, "image": "https://spoonacular.com/cdn/ingredients_100x100/pineapple.jpg", "cleanTitle": "Pineapple" }</div>				

Description

[QA6D-34] API Post : Detect Food In Text Created: 08/Aug/18 Updated: 10/Aug/18	
Status:	Retired
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None


Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Issue links:	Blocks blocks QA6D-28 QA6 spoonacular API testing Open				
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-39	Execute QA6D-34	Test Execution	Passed	
Epic Link:	QA6 DF spoonacular				
Test Preconditions:	<div>1. Have Postman Open.</div> <div>2. Have Test Data Data available.</div> <div>3. API Key "X-Mashape-Key", "oaNmJs8gdImshKNbXOPxLKHG68vOp1MM9h7jsn6WvTZ0Zn4eqC"</div> <div>4. API Reference documentation here</div>				
Test Script:	<div>1. Open postman and confirm URL is correct</div> <div>2. Run POST Request to extract food items from provided Text</div> <div>3. " First announced in 2000, Foodfight! takes place in a supermarket, and is set among the colourful, kid-friendly world of brand mascots.</div> <div>We're not just talking fictional mascots either: in the movie, invented characters such as hero Dex Dogtective (who fronts a cereal line and is voiced by Charlie Sheen) rub badly animated shoulders with the real-world likes of Charlie the Tuna, a cartoon tuna fish created to sell, well, tuna fish, and motherly Mrs Butterworth, a US pancake-mix mascot. The UK equivalent would be a film starring fictional animated characters alongside Tony The Tiger or the Coco Pops monkey."</div> <div>1. Confirm data provided is correct.</div>				
Test Postconditions:	<div>Requested API data should be provided.</div> <div>"annotations": [<div>{ "annotation": "cereal", "tag": "dish", "image": "https://spoonacular.com/cdn/ingredients_100x100/rice-crispy-cereal.png" }</div><div>,</div><div>{ "annotation": "tuna", "tag": "dish", "image": "https://spoonacular.com/cdn/ingredients_100x100/tuna-canned.jpg" }</div><div>,</div><div>{ "annotation": "hero", "tag": "dish", "image": "https://spoonacular.com/menultemImages/sandwich.jpg" }</div><div>,</div><div>{ "annotation": "fish", "tag": "dish", "image": "https://spoonacular.com/cdn/ingredients_100x100/fish-fillet.jpg" }</div><div>,</div><div>{ "annotation": "pops", "tag": "dish", "image": "https://spoonacular.com/menultemImages/plastic-to-go-drink-cup.jpg" }</div><div>,</div><div>{ "annotation": "rub", "tag": "dish", "image": "https://spoonacular.com/cdn/ingredients_100x100/seasoning.png" }</div></div>				

Description

Using Postman, confirm functioning of API Post: Detect Food in Text command is able to extrapolate food items out of a body of text.

[QA6D-28] QA6 spoonacular API testing Created: 06/Aug/18 Updated: 10/Aug/18			
Status:	Open		
Project:	QA6 Dan		
Component/s:	None		
Affects Version/s:	None		
Fix Version/s:	None		
Type:	Test Plan	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		
Issue links:	<div>Blocks</div> <div><div>is blocked by</div><div>QA6D-26</div><div>QA6 spoonacular 'Recipe' functionality</div><div>Open</div></div> <div><div>is blocked by</div><div>QA6D-34</div><div>API Post : Detect Food In Text</div><div>Retired</div></div> <div><div>is blocked by</div><div>QA6D-35</div><div>API Post: Classify a Grocery Product</div><div>Retired</div></div> <div><div>is blocked by</div><div>QA6D-36</div><div>API Get: Extract Recipe from Website</div><div>Retired</div></div> <div><div>is blocked by</div><div>QA6D-37</div><div>API Get: Random Recipe</div><div>Retired</div></div> <div><div>is blocked by</div><div>QA6D-39</div><div>Execute QA6D-34</div><div>Passed</div></div> <div><div>is blocked by</div><div>QA6D-40</div><div>Execute QA6D-35</div><div>Passed</div></div>		
Epic Link:	QA6 DF spoonacular		
Introduction:	Testing spoonacular's API allows for a Robust set of Features. Some of those features include the ability to POST detect food in Text, POST Classify a grocery product, GET a random recipe, and GET extract a recipe from a website.		
Test Items:	<div>1. Link to API</div> <div>2. API documentation</div> <div>3. Postman</div>		
Features to be Tested:	<div>1. Post : Detect Food In Text QA6D-34 RETIRED</div> <div>2. Post: Classify a Grocery Product QA6D-35 RETIRED</div> <div>3. Get: Extract Recipe from Website QA6D-36 RETIRED</div> <div>4. Get: Random Recipe QA6D-37 RETIRED</div>		
Features Not to be Tested:	The API allows for an extensive set of functions. To to constraints a list of items NOT to be tested can be found here .		
Approach:	<div>1. Testing will include the use of POSTMAN as well as Mashape's own API testing application built into their website.</div> <div>LINK</div>		
Item Pass/Fail Criteria:	<div>1. Post : Detect Food In Text</div> <div><div>1. QA6D-34 RETIRED will test API's ability to scan a block of text, detect any food items in the text, and present those to the user, parsed.</div></div> <div>2. Post: Classify a Grocery Product</div> <div><div>1. QA6D-35 RETIRED tests API's ability to categorize a grocery item from a text string. This then presents the user tier item requested food category .</div></div> <div>3. Get: Extract Recipe from Website</div> <div><div>1. QA6D-36 RETIRED Tests the API's ability to Extract a recipe from a URL, the API will also parse the ingredients and present it in a consistent format.</div></div>		



	4. Get: Random Recipe 1.  QA6D-37 RETIRED shows the API can provided some parameters in the form of a string, present a random recipe to the user.
Test Tasks:	1. Create Test Plan 2. Research API's documentation for appropriate commands 3. Select 4 commands that prove API is dynamic, reliable, and programable. 4. Create Test Cases for API 5. Execute Test Cases.
Environment Needs:	1. Internet connection 2. Postman application 3. API documentation 4. API key
Risks & Contingencies:	1. API testing can prove problematic with API access being dependent on a variety of factors. 2. This testing will only prove functional requirements, not non functional requirement such as reliability and performance.
Other Supporting Info:	Deadline for project: August 10th 2018
Approvals:	Lead Instructor: AJ Larson Mentor: Andrew Call Spiritual Advisor: Max Bracken Life Coach: Vitaily Matiyev Astrologer: Bryen North

[QA6D-38] Application Crash on Android Device. Created: 09/Aug/18 Updated: 09/Aug/18	
Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Bug	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		

[QA6D-23] 4.2 Project Black Box Techniques Created: 24/Jul/18 Updated: 25/Jul/18	
Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Plan	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0

8/10/2018	JIRA
Labels:	None
Remaining Estimate:	Not Specified
Time Spent:	Not Specified
Original Estimate:	Not Specified
Attachments:	 4.2 PROJECT.xlsx  4.2Project.pdf
Issue links:	Blocks is blocked by QA6D-24 4.2 Project Black Box Techniques Open
Introduction:	Your project today will be to use the black box techniques discussed over the past few days to implement/retrofit testing for the weatherman application.
Test Items:	https://devmountain-qa.github.io/weatherman/build/index.html
Features to be Tested:	Regression ready test cases have been provided, with linked examples of Decision Tables and/or State Transition Diagrams and Tables. You should use Boundary Value Analysis to further refine the testing you will need.
Approach:	Decision Tables and/or State Transition Diagrams and Tables.
Item Pass/Fail Criteria:	Optional: Weatherman could be replaced as the application to test by any other web application with instructor permission. Goal: Regression ready test cases have been provided, with linked examples of Decision Tables and/or State Transition Diagrams and Tables. You should use Boundary Value Analysis to further refine the testing you will need. Stretch: Test Automation has been implemented Black Diamond: Data-driven automated testing has been implemented.
Test Tasks:	Create Test Case Create Decision Tables and State Transition Diagram Automate.
Environment Needs:	Lucid chart Nightwatch JS

4.2 Project Black Box Techniques (QA6D-24)

[QA6D-25] 4.2 Project Black Box Techniques

Created: 24/Jul/18 Updated: 24/Jul/18

Status:	Open		
Project:	QA6 Dan		
Component/s:	None		
Affects Version/s:	None		
Fix Version/s:	None		
Type:	Test Execution	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		
Test Preconditions:	App loaded up		

[QA6D-24] 4.2 Project Black Box Techniques Created: 24/Jul/18 Updated: 24/Jul/18	
Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Issue links:	Blocks				
	blocks	QA6D-23	4.2 Project Black Box Techniques	Open	
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-25	4.2 Project Black Box Techniques	Test Execution	Open	
Test Preconditions:	App loaded up				

Description

Your project today will be to use the black box techniques discussed over the past few days to implement/retrofit testing for the weatherman application.

[QA6D-15] 3.4 Project: Using Exploratory Testing, cards.devclock DF Created: 19/Jul/18 Updated: 22/Jul/18	
Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Plan	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		

Issue links:	Blocks			
	is blocked by	QA6D-18	Account creation, Password change, ac...	Open
	is blocked by	QA6D-16	Create, edit and delete, Cards and De...	Ready

	is blocked by QA6D-17	Execute procedures in QA6D-16	Failed
	is blocked by QA6D-19	Execute procedures in QA6D-17	Failed
	Cloners		
	clones QA6D-13	3.3 Project: Using Exploratory Testin...	Open
Epic Link:	Cards Testing		
Introduction:	Test functionality of app in live environment. Find bugs; if present.		
Test Items:	Artifacts to be tested: https://cards.devclock.com/login username : daniel@dflanagan.com password: 1punchman		
Features to be Tested:	1. Creation of cards. 2. Creation of decks. 3. Deletion of cards. 4. Deletion of decks. 5. Mastery level works. 6. Mastery level diminishes. 7. Presenting of cards based on mastery level, IE lower mastery first.		
Features Not to be Tested:	1. Google SSO. 2. Design decisions. 3. Payment.		
Approach:	1. Initial round of exploratory testing. 2. Static analysis. 3. Black box testing. 4. Gathering of selectors. 5. Automation testing of components.		
Item Pass/Fail Criteria:	1. Flash cards able to be created, edited, deleted. 2. Decks and cards able to be renamed, deleted. 3. Decks and cards are saved after user logs off. 4. Decks and cards available when user logs in. 5. Mastery function works. 6. Cards lose mastery after time. 7. Cards able to be presented based on mastery.		
Test Tasks:	1. Exploratory testing 2. Build test cases 3. Find selectors 4. Begin automation 5. Report bugs		
Environment Needs:	1. Application loaded into Chrome 2. Nightwatch JS and Selenium 3. Account for cards-beta		
Risks & Contingencies:	Exhaustive testing of card inputs impossible. Issues with Google SSO and account persistence.		
Other Supporting Info:	Will determine test schedule after exploratory testing.		
Approvals:	Approval from AJ and Andrew.		

Comments

Comment by Daniel Flanagan [22/Jul/18]

Test Summary

- Cards was tested first using manual exploratory testing to confirm base functionality. Bugs discovered during initial testing for "Deletion of Cards and decksQA6D-20, [QA6D-21](#) [OPEN](#) , QA6D-22"
- Testing was conducted in OSX 10.13.3 with Chrome Version 67.0.3396.99 (Official Build) (64-bit)
- Test pre conditions were App loaded into browser at <https://cards.devclock.com/login>
- Test Script included in [QA6D-16](#) [READY](#) and [QA6D-18](#) [OPEN](#) . Account created for this test.
- Base functionality confirmed. App suffers from major Bugs however.
 - Cannot Delete an Account
 - Cannot Delete a Deck
 - Cannot move a card between Decks
 - Contact Us page is missing
 - Password field needs convention. (length req., too many characters.possible)

- Unable to Test payment system do to constraints of project
- Did not test Sign in with Google.
- Added following functionality testing after initial testing had begun.
 - Tested Search functionality within Decks. Searching Works.
 - Tested Mastery Functionality. Mastery Works
- QA's opinion of product is that it passes functional testing req. but requires major rework before it passes full acceptance criteria for delivery to client.
 - App suffers from design flaws, color scheme and button placement, however design is reactive across multiple devices/ formats.
 - App does not offer enough functionality for its asking price of \$9/month. More functionality is recommend or lower price to perhaps free and support with ads.
 - App has good ideas but could execute them in a more cohesive and intelligent fashion. Should start at Decks and use settings as sidebar item. Perhaps split design
- QA has also built an Automated test Script in Nightwatch JS to automate testing:
 - Does app login.
 - Are all pages from <https://cards.devclock.com/> present at their URLs and is there 1 unique element of the page present.
- More Automation can be built if required.
- Base functionality is established but some rework to fix major bugs is needed before product delivery. In order of significance.
 1. Delete Accounts
 2. Contact Us page
 3. Move Cards between Decks
 4. Delete Decks
 5. Password complexity requirements.

[QA6D-22] Contact Us Page is 404 Created: 21/Jul/18 Updated: 21/Jul/18

Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Bug	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		

Steps to Reproduce:	<ol style="list-style-type: none"> 1. Login to site with any user account. 2. Click on the Contact us page. 3. Instead of contact us page, get 404 erro message. 4. URL https://www.devclock.com/contact
Expected Result:	Able to view Contact US page.
Actual Result:	Unable to view contact us page.
Epic Link:	Cards Testing

Description

After login on main page, unable to view Contact Us page. Getting 404 file or directory not found.

[QA6D-21] Can't Move Card between Decks Created: 21/Jul/18 Updated: 21/Jul/18

Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Bug	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		

Steps to Reproduce:	<ol style="list-style-type: none">1. Login to application.2. Go to Decks. Click on the name of the deck to see cards.3. Once in a deck, click on the edit button on the right hand side of the text field.4. Click on the move button.5. You will see decks where you can move the card.6. Select any deck, click ok.7. You will get a few second pause of no action then a generic error message saying no response.
Expected Result:	Able to move a card between decks.
Actual Result:	Unable to move a card between decks using the move button when editing an existing card.
Epic Link:	Cards Testing

Description

Perform steps in [QA6D-17](#) **FAILED** . During manipulation of cards and decks, unable to move a card between test123 and test 3 decks. Get generic no response error.

[QA6D-20] Cannot delete account. Created: 21/Jul/18 Updated: 21/Jul/18

Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Bug	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		
Environment:	OSX 10.13.3, Chrome Version 67.0.3396.99		

Steps to Reproduce:	<ol style="list-style-type: none">1. Login with user account daniel@dflanagan.com pass: 3punchman2. At home screen go to settings menu.3. Scoll to bottom where you see red box that says Delete account.4. Re Enter password to re-authenticate.5. Get warning message about all account data being removed.6. Click yes to delete.
---------------------	---

8/10/2018

JIRA

	7. Get about 5 second pause, then the no response dialog.
Expected Result:	Able to delete user account. Would most likely be logged out at end of process.
Actual Result:	Unable to delete user account.
Epic Link:	Cards Testing

Description

During procedure [QA6D-19](#) FAILED , I was unable to remove my account from the "Settings" menu after re-authentication prompt. The resulting error shows a no response dialog from a small browser window.

[QA6D-18] Account creation, Password change, account deletion. Created: 21/Jul/18 Updated: 21/Jul/18			
Status:	Open		
Project:	QA6 Dan		
Component/s:	None		
Affects Version/s:	None		
Fix Version/s:	None		
Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Issue links:	Blocks blocks QA6D-15 3.4 Project: Using Exploratory Testin... Open				
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-19	Execute procedures in QA6D-17	Test Execution	Failed	
Epic Link:	Cards Testing				
Test Preconditions:	App should be loaded into the browser. All test data should be created before hand, ie decks and cards info, accounts. Subject matter is arbitrary.				
Test Script:	<div>1. Create test account using provided test email address. Daniel@dflanagan.com Password 1punchman.</div> <div>2. Log off account.</div> <div>3. Log back in with incorrect password.</div> <div>4. Should get login error.</div> <div>5. Attempt to reset password. Password 2punchman</div> <div>6. Login to account with new password.</div> <div>7. If Login attempt successful. Attempt to delete account from settings menu.</div>				
Test Postconditions:	User should be able to create account, reset password and delete account.				

Description

Create a test account. Change the accounts password. Delete account.

[QA6D-16] Create, edit and delete, Cards and Decks. Created: 19/Jul/18 Updated: 19/Jul/18	

8/10/2018

JIRA

Status:	Ready
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Daniel Flanagan
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Issue links:	Blocks				
	blocks	QA6D-15	3.4 Project: Using Exploratory Testin...		Open
	is blocked by	QA6D-17	Execute procedures in QA6D-16		Failed
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-17	Execute procedures in QA6D-16	Test Execution	Failed	Daniel Flanagan
Epic Link:	Cards Testing				
Test Preconditions:	1. App should be loaded into the browser. All test data should be created before hand, ie decks and cards info. Subject matter is arbitrary.				
Test Script:	1. Load application. 2. sign in with provided user account and password. 3. Go to deck section of app, then click the plus arrow at the top to create a new deck. 4. Create new cards for the deck 5. Save the deck and try to "Study" with it. 6. Create a second deck of cards. Use different data. 7. Go back and modify first decks contents. Are they saved? 8. Delete second deck of cards. Do you get a prompt before deletion? 9. Check to see if deck is gone. 10. Log off. Load incognito window. Log back in and see if cards, decks still present.				
Test Postconditions:	App will maintain decks for user after log off. Decks and cards can be modified and deleted.				

Description

Use static analysis, dynamic testing, and automation to test creation, modification, and deletion of cards and decks.

[QA6D-13] 3.3 Project: Using Exploratory Testing, cards-beta DF

Created: 18/Jul/18 Updated: 19/Jul/18

Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Plan	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		

Time Spent:	Not Specified
Original Estimate:	Not Specified
Issue links:	<p>Blocks</p> <p>is blocked by QA6D-14 Cards-beta, test CC processing Open</p> <p>Cloners</p> <p>is cloned by QA6D-15 3.4 Project: Using Exploratory Testin... Open</p>
Epic Link:	Cards Testing
Introduction:	Test functionality of app. Find bugs if present.
Test Items:	<p>Artifacts to be tested:</p> <p>https://cards-beta.devclock.com/</p> <p>username : daniel@dflanagan.com</p> <p>password: 1punchman</p>
Features to be Tested:	<ol style="list-style-type: none"> 1. Creation of cards. 2. Creation of decks. 3. Deletion of cards. 4. Deletion of decks. 5. Mastery level works. 6. Mastery level diminishes. 7. Presenting of cards based on mastery level, IE lower mastery first.
Features Not to be Tested:	<ol style="list-style-type: none"> 1. Google SSO 2. Design decisions.
Approach:	<ol style="list-style-type: none"> 1. Initial round of exploratory testing. 2. Static analysis. 3. Black box testing. 4. Gathering of selectors. 5. Automation testing of components.
Item Pass/Fail Criteria:	<ol style="list-style-type: none"> 1. Flash cards able to be created, edited, deleted. 2. Decks and cards able to be renamed, deleted. 3. Decks and cards are saved after user logs off. 4. Decks and cards available when user logs in. 5. Mastery function works. 6. Cards lose mastery after time. 7. Cards able to be presented based on mastery.
Test Tasks:	<ol style="list-style-type: none"> 1. Exploratory testing 2. Build test cases 3. Find selectors 4. Begin automation 5. Report bugs
Environment Needs:	<ol style="list-style-type: none"> 1. Application loaded into Chrome 2. Nightwatch JS and Selenium 3. Account for cards-beta
Risks & Contingencies:	<p>Exhaustive testing of card inputs impossible.</p> <p>Issues with Google SSO and account persistence.</p>
Other Supporting Info:	Will determine test schedule after exploratory testing.
Approvals:	Approval from AJ and Andrew.

[QA6D-14] Cards-beta, test CC processing Created: 18/Jul/18 Updated: 18/Jul/18

Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

8/10/2018

JIRA

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		


Issue links:	Blocks blocks QA6D-13 3.3 Project: Using Exploratory Testin... Open
Epic Link:	Cards Testing
Test Preconditions:	<div>1. App signed in and subscription page navigated to.</div> <div>2. CC information from stripe.com/docs/testing.</div>
Test Script:	<div>1. Login to https://cards-beta.devclock.com/subscription. Login with provided username and password.</div> <div>2. at the menu page, click on subscriptions.</div> <div>3. You should see a field for CC number. As you fill the field out the cc type will be selected.</div> <div>4. Fill out with cc number 4242424242424, exp 01/19 cvv 823</div> <div>5. or 5555555555554444 . exp 05/21 . cvv 261</div> <div>6. Click on Subscribe for \$9/month button.</div> <div>7. Pop up should indicate that you have successfully upgraded to premium tier.</div>
Test Postconditions:	<div>1. CC information was processed.</div> <div>2. User now has \$9/month premium subscriptions which allows for unlimited cards.</div>

Description

Test CC processing with Stripe.com/docs/testing on subscription page <https://cards-beta.devclock.com/subscription>.

[QA6D-12] QA6 v1.1 Employee Mgmt Sys DF <small>Created: 16/Jul/18 Updated: 16/Jul/18</small>	
Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Plan	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Daniel Flanagan
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Attachments:	 employee-managerv1.1.pdf				
Issue links:	Cloners clones QA6D-7 QA6 Employee Mgmt Sys DF Open				
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-10	Test Employee manager blank field det...	Test Execution	Passed	Daniel Flanagan
Epic Link:	Module 1 Assessment				
Introduction:	<div>1. This project will test a front end for an employee management system.</div> <div>2. It should run on any modern browser (post I.E. 8 and similar), and be fully functional, and reasonably styled.</div>				

Test Items:	Wireframe.pdf (attached) Requirements Doc Link to App
Features to be Tested:	<ol style="list-style-type: none"> Edit existing employees for individual or multiple values, and the changes will persist, including employee names in the list to the left. Cancelling prior to saving will return edited fields to their prior values. Navigating to another employee without saving will NOT save changes. Employee IDs should be whole positive numbers. Save and Cancel buttons should be disabled when no changes have been made. All fields must be populated. <ol style="list-style-type: none"> ID Name Phone Title Pressing Save, save's changes made. Cancel button will revert the employee's file back to the most recently saved version. All fields except for ID are editable. Changes persist after navigation to alternate record. <p>*****features added as part of v1.1*****</p> <ol style="list-style-type: none"> Added error messages for incorrect inputs The invalid field is underlined in red Fields cannot be saved while invalid
Features Not to be Tested:	<p>To be aware of, but not currently implemented</p> <p>FUTURE FEATURES</p> <ol style="list-style-type: none"> Phone numbers will be validated 10 digit, area code included, US phone numbers. The Employee List will reside in a database and be accessed through API integration An Add Employee function will be implemented A search function to filter the employee list will be available, based on Job Title, Name, and/or ID An Email Address field will be added for each employee file All fields will be required for any entry/modification
Approach:	<ol style="list-style-type: none"> Manual testing will be the primary methodology Will use Document requirements and wireframe to form test case basis. After v1.1, introduce automated testing to test case.
Item Pass/Fail Criteria:	<p>Acceptance Criteria</p> <ol style="list-style-type: none"> Edit existing employees for individual or multiple values, and the changes will persist, including employee names in the list to the left. Cancelling prior to saving will return edited fields to their prior values. Navigating to another employee without saving will NOT save changes. Employee IDs should be whole positive numbers. Save and Cancel buttons should be disabled when no changes have been made. All fields must be populated. <p>ID Name Phone Title</p> <ol style="list-style-type: none"> Changes persists after navigation. Cancel button reverts any changes to last saved record. Error message if field is invalid. Invalid fields will underline in red You cannot save the record while the field is invalid.
Test Tasks:	<p>Test following</p> <ol style="list-style-type: none"> Can Employee fields be modified, and changes saved after clicking submit? Does changing the Name field affect the Employee name on the left hand side of the page? Does clicking cancel, undo any changes made to record? Will navigating to new record without saving changes , cancel any changes made? Are Employee ID's positive numbers? Do the save and Cancel buttons work if no change are made? Must all fields be populated? Use wireframe and Readme doc to inform testing procedures. Create test cases to test save changes functionality and Blank fields detection.

8/10/2018	JIRA
	10. Are invalid fields highlighted in red and does the error message appear? 11. Can you save a record while a field is invalid?
Environment Needs:	Any modern browser (post I.E. 8 and similar) Chrome will be our recommended browser. App in Test Items.
Risks & Contingencies:	1. Incorrect employee data. 2. Requirements incomplete. 3. Incorrect error message. 4. Able to save with incorrect data.
Approvals:	The Money: Big Cheese Chief Disinformation Officer: Head Honcho Executioner: El Heffe Procurement: Hisenberg

[QA6D-7] QA6 Employee Mgmt Sys DF Created: 06/Jul/18 Updated: 16/Jul/18	
Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Plan	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Daniel Flanagan
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		

Attachments:	Wireframe.pdf																										
Issue links:	Blocks <table><tr><td>is blocked by</td><td>QA6D-5</td><td>Changes are saved in Employee records...</td><td>Open</td></tr><tr><td>is blocked by</td><td>QA6D-8</td><td>Employee Manager, Test Save Function,...</td><td>Open</td></tr><tr><td>is blocked by</td><td>QA6D-9</td><td>Test Blank Field Detection in Emplo...</td><td>Open</td></tr><tr><td>is blocked by</td><td>QA6D-11</td><td>Run Check of Submit button, Navigate ...</td><td>Failed</td></tr><tr><td>is blocked by</td><td>QA6D-10</td><td>Test Employee manager blank field det...</td><td>Passed</td></tr></table> Cloners <table><tr><td>is cloned by</td><td>QA6D-12</td><td>QA6 v1.1 Employee Mgmt Sys DF</td><td>Open</td></tr></table>			is blocked by	QA6D-5	Changes are saved in Employee records...	Open	is blocked by	QA6D-8	Employee Manager, Test Save Function,...	Open	is blocked by	QA6D-9	Test Blank Field Detection in Emplo...	Open	is blocked by	QA6D-11	Run Check of Submit button, Navigate ...	Failed	is blocked by	QA6D-10	Test Employee manager blank field det...	Passed	is cloned by	QA6D-12	QA6 v1.1 Employee Mgmt Sys DF	Open
is blocked by	QA6D-5	Changes are saved in Employee records...	Open																								
is blocked by	QA6D-8	Employee Manager, Test Save Function,...	Open																								
is blocked by	QA6D-9	Test Blank Field Detection in Emplo...	Open																								
is blocked by	QA6D-11	Run Check of Submit button, Navigate ...	Failed																								
is blocked by	QA6D-10	Test Employee manager blank field det...	Passed																								
is cloned by	QA6D-12	QA6 v1.1 Employee Mgmt Sys DF	Open																								
Epic Link:	Module 1 Assessment																										
Introduction:	1. This project will test a front end for an employee management system. 2. It should run on any modern browser (post I.E. 8 and similar), and be fully functional, and reasonably styled.																										
Test Items:	Wireframe.pdf (attached) Requirements Doc Link to App																										
Features to be Tested:	1. Edit existing employees for individual or multiple values, and the changes will persist, including employee names in the list to the left. 2. Cancelling prior to saving will return edited fields to their prior values. 3. Navigating to another employee without saving will NOT save changes. 4. Employee IDs should be whole positive numbers. 5. Save and Cancel buttons should be disabled when no changes have been made. 6. All fields must be populated. 1. ◦ ID																										

	<ul style="list-style-type: none"> ◦ Name ◦ Phone ◦ Title <p>7. Pressing Save, save's changes made.</p> <p>8. Cancel button will revert the employee's file back to the most recently saved version.</p> <p>9. All fields except for ID are editable.</p> <p>10. Changes persist after navigation to alternate record.</p>
Features Not to be Tested:	<p>To be aware of, but not currently implemented</p> <p>FUTURE FEATURES</p> <ol style="list-style-type: none"> 1. Phone numbers will be validated 10 digit, area code included, US phone numbers. 2. The Employee List will reside in a database and be accessed through API integration 3. An Add Employee function will be implemented 4. A search function to filter the employee list will be available, based on Job Title, Name, and/or ID 5. An Email Address field will be added for each employee file 6. All fields will be required for any entry/modification
Approach:	<ol style="list-style-type: none"> 1. Manual testing will be the primary methodology 2. Will use Document requirements and wireframe to form test case basis.
Item Pass/Fail Criteria:	<p>Acceptance Criteria</p> <ol style="list-style-type: none"> 1. Edit existing employees for individual or multiple values, and the changes will persist, including employee names in the list to the left. 2. Cancelling prior to saving will return edited fields to their prior values. 3. Navigating to another employee without saving will NOT save changes. 4. Employee IDs should be whole positive numbers. 5. Save and Cancel buttons should be disabled when no changes have been made. 6. All fields must be populated. <p>ID</p> <p>Name</p> <p>Phone</p> <p>Title</p> <ol style="list-style-type: none"> 7. Changes persists after navigation. 8. Cancel button reverts any changes to last saved record.
Test Tasks:	<p>Test following</p> <ol style="list-style-type: none"> 1. Can Employee fields be modified, and changes saved after clicking submit? 2. Does changing the Name field affect the Employee name on the left hand side of the page? 3. Does clicking cancel, undo any changes made to record? 4. Will navigating to new record without saving changes , cancel any changes made? 5. Are Employee ID's positive numbers? 6. Do the save and Cancel buttons work if no change are made? 7. Must all fields be populated? 8. Use wireframe and Readme doc to inform testing procedures. 9. Create test cases to test save changes functionality and Blank fields detection.
Environment Needs:	<p>Any modern browser (post I.E. 8 and similar)</p> <p>Chrome will be our recommended browser.</p> <p>App in Test Items.</p>
Risks & Contingencies:	<p>Corruption of employee data during testing.</p> <p>Incorrect employee data.</p> <p>Requirements incomplete.</p>
Approvals:	<p>The Money: Big Cheese</p> <p>Chief Disinformation Officer: Head Honcho</p> <p>Executioner: El Heffe</p> <p>Procurement: Hisenberg</p>

[QA6D-9] Test Blank Field Detection in Employee Manager App Created: 06/Jul/18 Updated: 08/Jul/18

Status:	Open
Project:	QA6 Dan

Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Daniel Flanagan
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		

Issue links:	Blocks <div> <div>blocks</div> <div>QA6D-7</div> <div>QA6 Employee Mgmt Sys DF</div> <div>Open</div> </div> <div> <div>is blocked by</div> <div>QA6D-6</div> <div>Blank Field Detection does not functi...</div> <div>Open</div> </div>
Epic Link:	Module 1 Assessment
Test Preconditions:	<ol style="list-style-type: none"> 1. Tester should start with all fields populated. 2. App should also be loaded into browser.
Test Script:	<ol style="list-style-type: none"> 1. Tester will commence with app loaded up, all fields populated. 2. Tester will proceed to remove all fields from the employee record. Name, Phone Number, and Title. 3. Tester will then attempt to save changes. They should not be able to as the blank field detection will not allow blank fields in the employee record. 4. Tester will also attempt to leave fields blank, then navigate to another employee record. The expected behavior is that the fields should revert back to their prior fields as no changes should be saved since the submit button was never pressed.
Test Postconditions:	<ol style="list-style-type: none"> 1. The app's blank field detection shouldn't allow a field to be left blank, let alone the entire employee record. The changes should be reverted back to normal since an input is required for each field. 2. Navigating to another page after leaving another field blank should revert to previous saved data as a blank field is not allowed and the submit button was not pressed.

Description

This test will test the functions for the blank field detection in the Employee Manager App.

[QA6D-8] Employee Manager, Test Save Function, No save if Navigate, Submit, Cancel Buttons Created: 06/Jul/18 Updated: 08/Jul/18	
Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Case	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Daniel Flanagan
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified

Issue links:	Blocks <div> <div>blocks</div> <div>QA6D-7</div> <div>QA6 Employee Mgmt Sys DF</div> <div>Open</div> </div>
---------------------	---

	is blocked by QA6D-5 Changes are saved in Employee records... Open				
Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-11	Run Check of Submit button, Navigate ...	Test Execution	Failed	Daniel Flanagan
Epic Link:	Module 1 Assessment				
Test Preconditions:	<ul style="list-style-type: none"> User will start with no changes made to application. Application will be loaded on tester's browser. 				
Test Script:	<ol style="list-style-type: none"> Click on first employee record. Test if submit or cancel buttons make any changes. They should not work at this time. Add ! to each field in employee record, Name, ID, Telephone, title. Click to see if ID is editable. It should not be. Click on submit. Navigate to next employee record. Navigate back to previous record to confirm if ! has been added to each field and that the changes were saved. Remove the ! from each of the employee's fields. Do not click submit. Navigate to another employee record. Navigate back to first record. Have the changes been saved? The ! should remain as we did not confirm changes by clicking submit. Add ? to each field in the employee record you are currently navigated too. Click on the cancel button. Were the changes discarded? Repeat steps above for other employee records on page. <p>If applicable:</p> <ol style="list-style-type: none"> The exact steps to get from precondition to postcondition <ul style="list-style-type: none"> And their expected results This includes what buttons to click <ul style="list-style-type: none"> What should then occur What inputs to fill Etc. 				
Test Postconditions:	<ol style="list-style-type: none"> Only clicking on submit will allow changes to be saved to the employee record. Submit and cancel buttons have no function if no changes were made to file. Navigating to another page while making changes, discards any changes that are made. 				

Description

- Test the functionality of the save changes button in Employee manager.
- Test that changes are not saved if you navigate out of the page you have made your changes on.
- Test that the submit or cancel buttons do not function if no changes have been made.

[QA6D-6] Blank Field Detection does not function. Created: 06/Jul/18 Updated: 08/Jul/18			
Status:	Open		
Project:	QA6 Dan		
Component/s:	None		
Affects Version/s:	None		
Fix Version/s:	None		
Type:	Bug	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		
Issue links:	Blocks		

8/10/2018

JIRA

	blocks	QA6D-9	Test Blank Field Detection in Employee...	Open
Steps to Reproduce:	<div>1. Navigate to any employee record.</div> <div>2. Remove one or more fields.</div> <div>3. Click the submit button.</div>			
Expected Result:	<div>1. Blank fields should revert back to their original saved data. Blank fields are not allowed.</div>			
Actual Result:	<div>1. Blank fields are allowed to persist after saving record.</div>			

Description

Able to save records after leaving blank fields. Also navigating away from page allows changes saved. That bug has already been reported in [QA6D-5](#) [OPEN](#) .

[QA6D-5] Changes are saved in Employee records by navigating to another record. Created: 06/Jul/18 Updated: 08/Jul/18

Status:	Open
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Bug	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Daniel Flanagan
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		

Issue links:	Blocks			
	blocks	QA6D-7	QA6 Employee Mgmt Sys DF	Open
	blocks	QA6D-8	Employee Manager, Test Save Function,...	Open
Steps to Reproduce:	<div>1. Open application</div> <div>2. Navigate to any employee record.</div> <div>3. Add or remove nay character from each field</div> <div>4. Navigate to an alternate employee record</div> <div>5. Navigate back to original record.</div> <div>6. Changes should be made, even though submit was not pressed.</div>			
Expected Result:	<div>1. Navigating to an alternate page should not cause changes to be saved.</div> <div>2. Only click submit should allow any changes to be submitted.</div>			
Actual Result:	<div>1. Navigating to an alternate record, saves any changes made to the previous records, without pressing the submit button.</div>			
Epic Link:	Module 1 Assessment			

Description

Changes made to employee records are saved by navigating to another employee record. This is the same action as hitting the submit button, however that button is never pressed in this process.

[QA6D-1] Wanted Test Created: 05/Jul/18 Updated: 06/Jul/18

Status:	In Progress
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Plan	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Daniel Flanagan
Resolution:	Unresolved	Votes:	0
Labels:	None		
Σ Remaining Estimate:	Not Specified	Remaining Estimate:	Not Specified
Σ Time Spent:	Not Specified	Time Spent:	Not Specified
Σ Original Estimate:	Not Specified	Original Estimate:	Not Specified



Sub-tasks:	Key	Summary	Type	Status	Assignee
	QA6D-2	Test Enter Wanted with good data.	Test Execution	Failed	Daniel Flanagan
	QA6D-4	Test Enter Wanted App with bad data	Test Execution	Passed	Daniel Flanagan
Introduction:	<div>1. This project is to create a web form that will generate the text to submit to the NCIC to add a warrant.</div> <div>1. It if incorrect entry occurs, the proper error message will be displayed to assist the user in correcting their mistakes.</div> <div>1. Focus only on completing the form and validations, stylistic concerns are not being addressed at this time.</div>				
Test Items:	<div>1. Enter Wanted App</div> <div>2. enter-wanted/README</div> <div>3. Test Data.</div>				
Features to be Tested:	<div>1. Field Inputs, Do they enforce criteria?</div> <div>2. Results, Are they formatted in the correct fashion?</div> <div>The requirements are as follows for each field, and the results: Header: Required, 9-19 characters in length, any allowed MKE: Required, 2-4 alpha/special characters in lenth Originating Agency Identifier: Required, 9 alphanumeric characters in length Name: Rewired, 3-30 characters in length, any allowed Sex: Required, 1 character in length, alphabet only Race: Requires, 1 character in length, alphabet only Height: Required, 3 characters in length, numeric only Weight: Required, 3 characters in length, numeric only Hair: Required, 4-10 characters in length, alpha only Offense: Required, 5-15 characters in length, any allowed Date of Warrant/Violation, 8 characters in length, numeric in MMDDYYYY format Drivers License: Optional, 1-20 characters in length, any characters allowed, if included requires DL State & DL Expiration Year DL State: Optional, 2 characters in length, State Abbreviations only, if included requires Drivers Licens & DL Year DL Year: Optional, 4 characters in length, numeric in YYYY format, if included requires Drivers License & DL State</div>				
Features Not to be Tested:	Any Stylistic Concerns. Function only.				
Approach:	<div>1. Manual testing will be the main method</div> <div>2. Requirements will be based on enter-wanted/README</div> <div>3. The source of test data, some being inside of the parameters, some being not to test incorrect characters</div> <div>4. Test format and transmission of webform</div>				
Item Pass/Fail Criteria:	<div>All fields will be filled out with their correct number/type of characters.</div> <div>Results will be formatted correctly for transmission to NCIC.</div> <div>Upon successful entry, a text blob will be created, which consists of each of the values entered, separated by a '-' - any optional fields left blank will still be denoted in the blob by an additional '-'</div> <div>It is assumed that if incorrect entry occurs, the proper error message will be displayed to assist the user in correcting their mistakes.</div> <div>Webform is transmitted to NCIC in correct format.</div>				
Test Tasks:	<div>1. Create test data that falls within parameters set forth by requirements.</div>				

	2. Create test data that does not fall within guidelines for fields. 3. Use correct data to test proper functionality of system 4. Use incorrect data to test error detection in fields 5. Enter blank or incomplete data to test if required fields functionality is present. 6. Document any deviation from accepted parameters and steps used to recreate them.
Environment Needs:	Windows, PC, OSX Browsers: IE 10+, Microsoft Edge, Safari, Chrome, and Firefox should all be fully functional and supported. Safari, Chrome, and Firefox should all be tested
Risks & Contingencies:	1. Incorrect test data. <ul style="list-style-type: none"> Confer with client that test data fulfills requirements sufficiently. 2. Transmission of webform is unsuccessful to NCIC, or format is incorrect <ul style="list-style-type: none"> Schedule test with NCIC technicians.
Other Supporting Info:	Approval from Professor/ Peer before commencing Test Case.
Approvals:	Instructor Peers

Wanted Test (QA6D-1)
 [QA6D-2] Test Enter Wanted with good data. Created: 05/Jul/18 Updated: 05/Jul/18

Status:	Failed
Project:	QA6 Dan
Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Type:	Test Execution	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Daniel Flanagan
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		

Attachments:	 Wanted Testing Template.ods  qa wanted test data.rtf
Test Preconditions:	1. App should be loaded and all fields should be blank. 2. Device should be connected to internet to allow transmission of Warrant request. 3. Good Test Data should be available for data entry.
Test Script:	1. Open application and look to see if all fields are empty. 2. Using test data fill out each field with the good test data. This data represents acceptable criteria for each field. 3. There are 14 fields in total. Ensure all 14 fields have been filled out with the data that correlates with the question. Correct data is on the left hand side of attached TEST DATA.rtf 4. Click submit button to post Warrant request to NCIC.
Test Postconditions:	1. Form was filled out with appropriate data. Each field confirmed that data falls within entry criteria. No prompts for additional information of too many/too few characters or NA characters. 2. Submit button sends Warrant out to NCIC with correct data.
Test Execution Notes:	Test Failed with following result for good test data. Invalid The "Sex" should contain letters, no numbers, and nospecial characters. The "Race" should contain letters, no numbers, and nospecial characters. The "Hair" should contain letters, numbers, and nospecial characters. The "Date of Warrant/Violation" should contain letters, no numbers, and special characters. The "DL Expiration Year" should be between 3 and 3 characters long. The "License State" should be between 4 and 4 characters long.

Description

Using good test data, test all fields in Wanted app for acceptance of data that falls within criteria.

Comments

Comment by Daniel Flanagan [05/Jul/18]

Test Failed with following Error using good test data

Invalid The "Sex" should contain letters, no numbers, and nospecial characters.
The "Race" should contain letters, no numbers, and nospecial characters.
The "Hair" should contain letters, numbers, and nospecial characters.
The "Date of Warrant/Violation" should contain letters, no numbers, and special characters.
The "DL Expiration Year" should be between 3 and 3 characters long.
The "License State" should be between 4 and 4 characters long.
The "License Year" should contain letters, no numbers, and nospecial characters.
The "License Year" should be between 2 and 2 characters long.

[QA6D-3] Wanted App. Fields inconsistent in requirements. Created: 05/Jul/18 Updated: 05/Jul/18			
Status:	Open		
Project:	QA6 Dan		
Component/s:	None		
Affects Version/s:	None		
Fix Version/s:	None		
Type:	Bug	Priority:	Medium
Reporter:	Daniel Flanagan	Assignee:	Unassigned
Resolution:	Unresolved	Votes:	0
Labels:	None		
Remaining Estimate:	Not Specified		
Time Spent:	Not Specified		
Original Estimate:	Not Specified		
Steps to Reproduce:	15th line in app is License Plate number. Requirements makes no mention of License plate number. Field is below 'DL expiration year' and above 'License State'		
Expected Result:	No prompt for License Plate Number. Fields should only read Drivers License Number, State of issue, and expiration date for license.		
Actual Result:	Prompted for License plate number in app which was not listed in requirements.		

Description

App contains field 'License Plate Number'
Requirement do not specify License plate number. But specify Drivers license number, State of issue, and expiration year for license.