

Test Specification

Long Beach Vitality



David Li: 013996799

Anton Shirokov: 013591485

Andre Barajas: 011667576

Sovathana Heng: 016712603

Ernest Arreola: 016817994

Instructor: Dr. Birgit Penzenstadler

Date: April 7, 2019

Version 1.0

Unit Level

Test Case	#1
Test Level	Unit: Map Visualization
Quality criterion/attribute	External/Internal quality - Functionality
Description of test	User visualize the map with zip code boundaries
Requirements reference	Use Case: View Business Map
Steps of test cases	1. Navigate to the home page. 2. Click on "Dashboard"
Expected outcome	Check if a map is rendered with zip code boundary clearly defined

Test Case	#2
Test Level	Unit: Business List Request
Quality criterion/attribute	External/Internal quality - Interoperability
Description of test	The business list should be able to send a request to the backend, so that a list is returned to the front end for display.
Requirements reference	Use Case: Business List
Steps of test cases	1. Navigate to the home page. 2. Click on "Dashboard" 3. Click on a zip code boundary.
Expected outcome	A list of businesses should be generated and displayed.

Test Case	#3
Test Level	Unit: Interpretation of Business Lists
Quality criterion/attribute	External/Internal quality - Understandability

Description of test	User understands the list of businesses generated within a zip code
Requirements reference	Use Case: Business List
Steps of test cases	1. Navigate to the home page. 2. Click on "Dashboard" 3. Click on a zip code boundary.
Expected outcome	User should understand that the businesses on the list are the ones not doing well compare to others businesses that are not on the list.

Test Case	#4
Test Level	Unit: Zip Code Accuracy of Businesses within the Business List
Quality criterion/attribute	External/Internal quality - Accuracy
Description of test	Businesses in the business list should all be in the correct zip code.
Requirements reference	Use Case: Business List
Steps of test cases	1. Navigate to the home page. 2. Click on "Dashboard" 3. Click on a zip code boundary. 4. Click on a business with in the list generated
Expected outcome	The business's zip code should be the zip code the user clicked on.

Test Case	#5
Test Level	Unit: Generation of Business Lists
Quality criterion/attribute	External/Internal quality - Time Behavior
Description of test	The list generated after clicking on a specific zip code should be displayed with a fast response time.

Requirements reference	Use Case: Business List
Steps of test cases	<ol style="list-style-type: none"> 1. Navigate to the home page. 2. Click on "Dashboard" 3. Click on a zip code boundary.
Expected outcome	The list of businesses should be displayed within 10 seconds when user click on a zip code boundary.

Test Case	#6
Test Level	Unit: Business Social Media Links Reliability
Quality criterion/attribute	External/Internal quality - Reliability
Description of test	Social Media links for a business should direct user to the correct business
Requirements reference	Use Case: Business List
Steps of test cases	<ol style="list-style-type: none"> 1. Navigate to the home page. 2. Click on "Dashboard" 3. Click on a zip code boundary. 4. Click on a business with in the list generated 5. Click on a social media link for that business
Expected outcome	The social media page the application guides the user to should correspond with the business user selected. Same name and address often confirms the correctness.

Module Level

Test Case	# 1
Test Level	Module: Async Requests
Quality criterion/attribute	Internal and External Quality - Efficiency
Description of test	Everytime the main page loads, there are several components that must load with correct api calls, thus the user must be aware there are changes happening.
Requirements reference	Use Case: View Business Map
Steps of test cases	1. Tester loads the main page. 2. If Components aren't ready they are suspended or shown loading.
Expected outcome	The Main Page of the SPA should take correct measures to notify the user of what the program is doing, via loading or suspended components.

Test Case	# 2
Test Level	Module: Data Automation DB
Quality criterion/attribute	Internal and External Quality - Maintainability
Description of test	Modules within code should be able to automate data gathering from sources. So we must check if the scripts are able to run and update the database.
Requirements reference	N/A
Steps of test cases	1. Run data gathering script. 2. Check if data is most recent.
Expected outcome	The data stored in the database should be populated in accordance to the specified

	frequency.
--	------------

Test Case	# 3
Test Level	Module: Mapping Component
Quality criterion/attribute	Internal and External Quality - Functionality
Description of test	The API should render out a map with controls typically associated with common applications, zoom in, search, drag ... etc.
Requirements reference	N/A
Steps of test cases	<ol style="list-style-type: none"> 1. Navigate to home page. 2. Load Map 3. Move around map 4. Highlight markets. 5. Click around areas for events.
Expected outcome	Information should be able to be properly retrieved within the context of that component, Map should load and maintain familiar controls and functions.

Test Case	# 4
Test Level	Module: Health Aggregate
Quality criterion/attribute	Internal and External Quality - Operability
Description of test	Check whether businesses retrieve a vitality score, and If they don't have sufficient information don't display anything.
Requirements reference	Use Case: Business List
Steps of test cases	<ol style="list-style-type: none"> 1. Proceed to dashboard (Home Page). 2. Click around on map. 3. Zoom In 4. Find business and click on it. 5. Transferred to business page 6. Check several business for a Vitality Score.

Expected outcome	Businesses with all the available attributes should yield a score, and those that don't should show N/A.
------------------	--

Test Case	# 5
Test Level	Module: Dashboard
Quality criterion/attribute	Internal and External Quality - Operability
Description of test	This test should be for the general operation of the dashboard, every clickable item should be properly handled, anyone should be able to navigate around the dashboard without spending too much time fumbling around.
Requirements reference	Software Product Quality – Ease of Use
Steps of test cases	1. Proceed to dashboard. 2. Have tester engage with everything on dashboard.
Expected outcome	Everything should be in working order, without any significant errors that would render the app unusable, features available should all have been used.

Integration Level

Test Case	#1
Test Level	Integration: DB Connection
Quality criterion/attribute	Connecting to database via Django REST
Description of test	Running the system for 1st time

Requirements reference	Use case: start up
Steps of test cases	<ol style="list-style-type: none"> 1. Call DB 2. Wait for DB to start 3. If available, start Django app
Expected outcome	<p>The Database will not always start before Django which required a feature to wait for the DB to start up before starting the app.</p> <p>The test will expect a DB Free for connections as output.</p>

Test Case	#2
Test Level	Integration: Create user with Email
Quality criterion/attribute	Software Product Quality: Validation
Description of test	The test will if a user was create by the API if a email and passwd were given as input with correct format.
Requirements reference	Quality NFR: User Authentication ISO Standard
Steps of test cases	<ol style="list-style-type: none"> 1.) Display user create dashboard 2.) Received email and passwd params 3.) If both are valid 4.) Create user 5.) Validate user created
Expected outcome	The user must only be created if a correct format email is provided along with a secure passwd which meets passwd requirements.

Test Case	#3
Test Level	Integration: Test New Email normalized
Quality criterion/attribute	Software Product Quality: Accessibility
Description of test	User try to look for businesses who are at risk of delinquency at a certain area
Requirements reference	Use case: Create a user with bad email

	format
Steps of test cases	<ol style="list-style-type: none"> 1. Display create user dashboard 2. Receive bad email formatted and passwd. 3. Normalize email 4. Create user 5. Validate user creation
Expected outcome	The feature is expected to output a normalized email that is an email with capital letters will be converted to an email with lower case letters.

Test Case	#4
Test Level	Integration: Create super User
Quality criterion/attribute	Software Product Quality: Security
Description of test	Super User Creation for Admin Access
Requirements reference	Use case: Create a super user
Steps of test cases	<ol style="list-style-type: none"> 1. Display admin page for Django Rest 2. Receive email and passwd 3. Create super user 4. Validate user creation.
Expected outcome	The Super User must be created with the proper permissions to access certain layers of the stack and provide services to users.

Test Case	#5
Test Level	Integration: Tag creation
Quality criterion/attribute	Software Product Quality: Authentication
Description of test	Test the authentication of user and API called to endpoints
Requirements reference	Login, api calls and user creation.
Steps of test cases	<ol style="list-style-type: none"> 1. Receive tag 2. Authenticate tag with REST Framework 3. Return tag

	4. Test Tag string 5. Free resources being served.
Expected outcome	The expected outcome of this feature is to provide authentication for api calls when creating users or to endpoints which must have security implementation before establishing distributed network.

Test Case	#6
Test Level	Integration: Test Location
Quality criterion/attribute	Software Product Quality: Accuracy
Description of test	The generated list of locations for each business must all be up to date with the PostGres DB which hold the most up to date data.
Requirements reference	Use case: Location List Businesses
Steps of test cases	<ol style="list-style-type: none"> 1. Create test user 2. Enter new location for business 3. Call locations 4. Display all locations in database 5. Validate endpoint.
Expected outcome	The expected outcome of this feature is the return of a list which has been been generated and produced by users or the superuser of the application. This list must be up to date and the api call must return the user and name of location.

Acceptance Level

Test Case	#1
Test Level	Acceptance
Quality criterion/attribute	Software Product Quality - Credentials Validation

Description of test	User attempts to log into the website after registering.
Requirements reference	Functional requirements - Login interface
Steps of test cases	<ol style="list-style-type: none"> 1. Register user credentials 2. Attempt to login using registered credentials
Expected outcome	The expected outcome is a successful user login.

Test Case	#2
Test Level	Acceptance
Quality criterion/attribute	Software Product Quality - Map Visualization
Description of test	Ensure that the map of the Long Beach City is displayed on the front page of the website.
Requirements reference	Functional requirements - Google Map API implementation
Steps of test cases	<ol style="list-style-type: none"> 1. User logs in 2. User is taken to the front page of the website 3. User should see the map of the Long Beach City as the center piece
Expected outcome	The expected outcome is that user should see the map of Long Beach City as the centerpiece on the front page of the website.

Test Case	#3
Test Level	Acceptance
Quality criterion/attribute	Software Product Quality - Reliability
Description of test	Ensure that user is able to generate a list of delinquent businesses pertaining to a particular zip code once that zip code is clicked.

Requirements reference	Functional requirements - Business list generation
Steps of test cases	<ol style="list-style-type: none"> 1. Click on a zip code boundary within the Long Beach City map 2. Confirm that list of delinquent businesses is displayed
Expected outcome	The expected outcome is that a list of delinquent businesses is displayed once user clicks on any particular zip code boundary within the Long Beach City.

Test Case	#4
Test Level	Acceptance
Quality criterion/attribute	Software Product Quality - Reliability
Description of test	Ensure that a social media score, aka Vitality score, is displayed for every business.
Requirements reference	Functional requirements - social media APIs
Steps of test cases	<ol style="list-style-type: none"> 1. Generate business list by clicking on a zip code boundary within the Long Beach City. 2. Select a particular business 3. Make it has a Vitality score
Expected outcome	The expected outcome is that every selected business has a Vitality score.

Test Case	#5
Test Level	Acceptance
Quality criterion/attribute	Software Product Quality - Functionality
Description of test	Ensure that the front page of the website has a working dashboard.
Requirements reference	Functional requirements - dashboard components

Steps of test case	<ol style="list-style-type: none"> 1. Login to the website 2. Check whether the front page has a working dashboard
Expected outcome	The expected outcome is that the website's front page has a working dashboard.

System Level

Test Case	#1
Test Level	System: User Interface
Quality criterion/attribute	Internal/External quality - understandability
Description of test	First time user learn how to use the system
Requirements reference	Use case: Login
Steps of test cases	<ol style="list-style-type: none"> 1. First time user open up the system 2. Try to sign up or login without having to look for additional help 3. Use the map to view business information 4. Logout
Expected outcome	The user interface should be friendly enough and intuitive for first time user to navigate through the system without the need of step by step instruction to walk the user through the system

Test Case	#2
Test Level	System: access to the application from different devices
Quality criterion/attribute	Software Product Quality: portability
Description of test	The application layout on different devices with different screen size
Requirements reference	Quality NFR: Web Application Should Be Responsive

Steps of test cases	<ol style="list-style-type: none"> 1. Open up the application desktop browser and mobile browser 2. Login and go to dashboard 3. Click on map and view business information
Expected outcome	The application should adjust the layout of each component dynamically according to the screen size of the device the users are using

Test Case	#3
Test Level	System: view businesses at risk of delinquency
Quality criterion/attribute	Software Product Quality: helpfulness
Description of test	User try to look for businesses who are at risk of delinquency at a certain area
Requirements reference	Use case: view business map
Steps of test cases	<ol style="list-style-type: none"> 1. User click on a zipcode on the map 2. List of delinquent businesses is shown to the user 3. User click on a business in the list 4. User see the detail information of the business including contact information 5. Save the business to the list of businesses to reach out ot
Expected outcome	When user want to look for businesses to reach out to, user can choose which are he want to focus on and then pick an at risk business in that area to view detail information.

Test Case	#4
Test Level	System: access to the system 24/7
Quality criterion/attribute	Software Product Quality: availability
Description of test	User have 24/7 access to the system

Requirements reference	Use case:
Steps of test cases	<ol style="list-style-type: none"> 1. Login to the system in different time of the day. 2. Login to the system in different day of the week. 3. Login to the system during holiday
Expected outcome	User have access to the system at any time whether it is weekday or weekend or holiday.

Test Case	#5
Test Level	System: Only logged in user have access to the system
Quality criterion/attribute	Software Product Quality: confidentiality
Description of test	no access to the system without valid login credential
Requirements reference	Login to the system
Steps of test cases	<ol style="list-style-type: none"> 1. User go to the login section on the screen 2. Login with invalid credential multiple times 3. Username is frozen
Expected outcome	The system should block the username with multiple fail attempts to login and sending alert email to the user of that account regarding the login attempts

Test Case	#6
Test Level	System: accuracy of list of businesses at risk of delinquency
Quality criterion/attribute	Software Product Quality: accuracy
Description of test	The generated list of businesses at risk of delinquency must be accurate
Requirements reference	Use case: Business List
Steps of test cases	<ol style="list-style-type: none"> 1. Click on a zipcode on the map 2. List of businesses at risk in that

	<p>zipcode is generated</p> <ol style="list-style-type: none"> 3. Click on a business to view the detail information. 4. Looking at the graph of how the business is doing overtime. 5. Look at all other business information such as social media information to see how the business is doing. 6. Verify that the business is not doing well.
Expected outcome	The businesses in the generated list of at risk businesses are really at risk of delinquency.