

Test Strategy Document

Build a prospective list of customers based on Ideal Customer Profile

Team 25

Astitva Gupta

Sachin Kumar Danisetty

Trusha Sakharkar

Vivek Pamnani

Scope

Review: The document will be reviewed by both client and the team.

Approval: Client and Mentor of the team will approve the document.

The project is made in modules and assigned to team members. So, each module is unit tested (tested individually). Each of the integration with a new module will be followed by an integration test (tested regressively).

Test Approach

- Process of testing
 - Backend code consists of scrapy and node.
 - Scrapy module is unit tested manually by checking the number of items collected from page and no of items shown in page.
 - NodeJS is unit tested using postman and curl command.
 - Frontend code is written in React. Hence, it will be unit tested manually on the browser.
- Testing levels
 - Unit Testing: All the modules are tested individually
 - Integration Testing: Every integration will be followed by an integration testing on all the modules integrated.
 - System Testing - The whole system was tested after each sprint or before each release of the version.

- Acceptance Testing - The system developed was constantly checked by the some end users to ensure that the developed system was in the right track and as per needs asked by the client.
- Roles and responsibilities of each team member
 - **Astitva:** Testing the backend code using postman and curl command. Also taking care of integration testing when every module is integrated.
 - **Trusha and Vivek:** Testing frontend code (or React code) manually by checking visually in browser and console.
 - **Sachin:** Testing scrapy module by checking the items collected from the page and the number of results.
- Types of Testing
 - Performance Testing.
 - Unit Testing.
 - Integration Testing.
 - Regression Testing.
 - Stress Testing.
 - Endurance Testing.
- Testing approach & automation tool if applicable
 - White Box Testing.
 - Black Box Testing.
 - Bottom Up Integration Testing approach.

Test Environment

- Environment 1-
 - OS - Linux
 - Browser - Firefox
 - Network - Lan Wi-Fi setup
 - Web server - Local host
- Environment 2-
 - OS - Windows x64
 - Browser - Firefox
 - Network - Lan Wi-Fi setup
 - Web server - Local host
- Environment 3-
 - OS - Linux
 - Browser - Chrome 80.0
 - Network - Lan Wi-Fi setup

- Web server - Local host
- Environment 4-
 - OS - Windows x64
 - Browser - Chrome 80.0
 - Network - Lan Wi-Fi setup
 - Web server - Local host

Testing Tools

- Postman: To test backend components.
- React Dev Tool: To test frontend components.
- Passport: To test SSO login via Google. Serves as a middleware.
- Mongo Compass: To test the database.

Use Cases

No.	Use Case Name	Description	Release
1.	User Authentication	User can log into the system via Google Auth.	R1
2.	Search	User can input a search phrase along with attributes/filters and initiate the search.	R1
3.	View Search History	User can view the list of their previous searches (hyperlinked to results) and opt for a new search.	R1
4.	Profile	User can view their personal information and opt to view their search history.	R1
5.	Logout	User can log out of the system ending the session.	R1
6.	Show Results	User can view the results of a chosen search and choose sorting parameters as desired.	R1
5.	View List of Employees (potential contacts)	Users can view the people to be contacted for communication with the prospective customer.	R2
6.	Company Knowledge Base	Users can view the list of interesting news articles, announcements, etc. about customer companies.	R2
7.	Startup Profile	User can view the complete details of a search result.	R2

Test Cases

Test No. ID	Related Use case	Pre-conditions	Test Description (steps)	Expected Outcome
1	User Authentication	User has Google account for SSO	1. New User clicks the SSO button on home page. 2. User enters their details on the SSO host's portal. 3. On succesful log in, the host sends a token and the user is added to the database.	User is redirected to their emply search history page.
2	User Authentication	User has Google account for SSO	1. Existing User clicks the SSO button on home page. 2. User enters their details on the SSO host's portal. 3. On succesful log in, the host sends a token which is checked in the database.	User is redirected to their search history page.
3	Search	User is logged in.	1. User enters the desired attrubutes. 2. User clicks the 'Search' button.	User is redirected to the results page.
4	Search	User is logged in.	1. User enters the invalid attrubutes. 2. User clicks the 'Search' button.	User is redirected to the search page, prompted for input.
5	Logout	User is logged in.	1. User clicks the 'Account' button on navbar. 2. User clicks the 'Logout' button.	User is logged out and redirected to the start page.
6	View Search History	User is logged in.	1. User clicks the 'Account' button on navbar. 2. User clicks the 'Search History' button.	User is redirected to the history page.
7	View Search History	User is logged in.	1. User clicks the 'Account' button on navbar. 2. User clicks the 'Search History' button. 3. User clicks on a search item.	User is redirected to the show results page of the selected search item.
8	View Search History	User is logged in.	1. User clicks the 'Account' button on navbar. 2. User clicks the 'Search History' button. 3. User clicks on a 'New Search' button.	User is redirected to the search page.
9	Profile	User is logged in.	1. User clicks the '<user name>' button.	User is redirected to their profile page.
10	Profile	User is logged in.	1. User clicks the '<user name>' button. 2. User clicks 'Search History' button.	User is redirected to the history page.
11	Show Results	1. User is logged in. 2. User has started a search.		The results are displayed to the user.
12	Show Results	1. User is logged in. 2. User has started a search.	1. User selects one or more 'sort' parameters.	The results are sorted accordingly and displayed to the user.
13	Search	1. User is not logged in. 2. User is on the search page.	1. User enters the desired attrubutes. 2. User clicks the 'Search' button.	Search failed, user not authenticated. User is redirected to welcome page.
14	Show Results	1. User is logged in. 2. User has started a search.	1. User click an entry in the results.	The selected element displays additional infromation.
15	Startup Profile	1. User is logged in. 2. User has started a search. 3. User is on show results page.	1. User double clicks an entry in the results.	User is redirected to startup profile page with information of the corresponding startup.
16	Startup Profile	1. User is logged in. 2. User is on startup profile page.	1. User clicks a social button under startup description	User is redirected to the corresponding profile page.
17	Startup Profile	1. User is logged in. 2. User is on startup profile page.	1. User clicks a contact.	User is redirected to the contact's Twitter page.
18	Startup Profile	1. User is logged in. 2. User is on startup profile page.	1. User clicks a Tweet.	User is redirected to the corresponding Twitter page.
19	View Search History	1. User is not logged in.	1. User enters URL of search history page.	User is redirected to the welcome page.
20	Profile	1. User is not logged in.	1. User enters URL of profile page.	User is redirected to the welcome page.
21	View Search History	1. User is logged in. 2. User is on the search history page.	1. User clicks the "Clear Search History" button.	User's search history in database is erased.
22	Search	1. User is logged in. 2. User has started a search.	1. User closes the browser/tab.	User recieves an email notification once the search is completed.