Casa Domus

Software Test Report (STR)

Date: May 12th, 2018

1 Scope

The purpose of this document is to describe the results of the tests performed as described on the Casa Domus web service. The results of the tests are given as either 'PASS' or 'FAIL'. The results of the test determine whether the web service has met the respective requirement given by the test.

1.1 Identification

This Software Test Report (STR) document applies to CasaDomus 1.0, a web-based home finding software that utilizes user inputs to generate a map with a color gradient for each county in the lower-forty eight states of the United States of America, corresponding to the closeness that the conditions in each county has with the user's preferences.

1.2 System overview

The purpose of this piece of software, which should run on any system that is capable of any of the two following web browsers, Chrome version 60.0+, and Firefox version 58.0+, is to enable users to easily find housing locations based off their preferences. The web-based software should perform these tasks by using user preferences obtained from a questionnaire to search through information obtained through various APIs, .csv files, and a database, which display to the user a map that is gradiented based off the comparability of the housing options in various areas to the user preferences.

The main function of the Casa Domus webservice is to find housing locations on the county level based off their preferences. When the webservice is loaded the system presents an initial questionnaire to obtain the user's basic living preferences. The user shall answer the questionnaire by the use of sliders. After submitting, the web service generates a colored map using the user's preferences obtained in the previous questionnaire. The system then searches through county information obtained through various .csv files and compares every county in the US with the user's preferences. After every county has been ranked, these results are sent to the Google Maps API, which displays a map that is gradiented based off similarity to the user's living preferences.

2 Referenced documents

Document	**Content Referenced**
Casa Domus Software Requirements Specification (SRS)	● Requirements in this documents are all referenced from the Casa Domus SRS ● System overview contents are all referenced from the Casa Domus SRS
Casa Domus Software Design Description (SDD)	● CSCI's in this document are all referenced from the Casa Domus SDD

Casa Domus Software Test Description (STD)	● Tests in this document are all referenced from the Casa Domus STD

3 Overview of test results

Test results are given next to their respective test. Tests that are passed are marked with a 'PASSED'. If a test was not passed it is marked with a 'FAILED', along with a short description of the problems encountered. If a test needed to be redesigned to better fit the requirement it test or the result that is provided, then a short description of the deviation taken will also be provided.

3.1 Overall assessment of the software tested

Given the requirements that were being tested for, only 3 of the requirements were not passed. This includes FUNC-1.1, FUNC-2.1, and FUNC-3.7. Due to the failure of these requirements, 3 tests were failed. This includes FUNC-1-TEST, FUNC-2.1-TEST, and FUNC-3.X.

None of these tests and requirements were critical to the function of the website. FUNC-1.1 suggested that the website to supply the median age for houses. This information could not be readily obtained online and therefore the requirement was not met. The requirement was listed as "should" requirement and is not critical to the function and operation of the website.

In addition FUNC-2.1 was not met because it required that FUNC-1.*, FUNC-3.*, and FUNC-5.* be met with the lower 48 states. Due to the failure of FUNC-1.1 described previously, FUNC-2.1 is automatically failed because the web service did not provide median ages of houses in the lower 48 states.

Finally, FUNC-3.7 suggested that the website allow the user to eliminate states they do not wish see on the map. This was not met because the feature was not implemented in the service. FUNC-3.7 was listed as a "should" requirement, and is not critical to the function and operation of the website.

3.2 Impact of test environment

The only potential difference between the test environment and the final operation of the website is that for testing purposes the web service is hosted locally on the developer's machine. In the final operation of the webservice, the website shall be hosted on a remote server, accessed by some url from the user's browser. This may result in some change in latency considering the local hosted server used in testing may respond faster than a remotely hosted server. But this change in latency will not affect any function of operation of the website.

3.3 Recommended improvements

Based on the failures of FUNC-1.1, FUNC-2.1, and FUNC-3.7, improvements to the data and features of the website could be made. Median ages of houses per county is not readily available, but it may be possible to acquire this information somehow. For counties in the lower

48 states which are missing data and are thrown out, their data may be acquired from a dataset that is not missing the information, or acquired separately from the current dataset and substituted in. As for FUNC-3.7, the feature for removing the states a user does not want in their results can be added at a later date.

4 Detailed test results

Included in this section are the identifiers of each test, and the exact results of each test along with any notes associated with each test.

4.1 (Project-unique identifier of a test)

Each of the tests have identifiers similar to the requirements that they are testing. Tests are named beginning with the requirement identifier that it tests with the string "-TEST" appended to the end. For example, a test that tests the requirement with identifier "PLATFORM-49" would have the test identifier "PLATFORM-49-TEST."

Each test serves to test a requirement that was specified in the Software Requirements Specification (SRS), their relationships are described below:

Identifier	Description of relationship to Requirements
PLAT-1-TEST	Tests whether or not the product meets requirement PLAT-1.
PLAT-1.1-TEST	Tests whether or not the product meets requirement PLAT-1.1.
PER-1-TEST	Tests whether or not the product meets requirement PER-1.
PER-1.1-TEST	Tests whether or not the product meets requirement PER-1.1.
FUNC-1-TEST	Tests whether or not the product meets requirement FUNC-1, FUNC-1.1, FUNC-1.2, FUNC-1.3, FUNC-1.4, FUNC-1.5, FUNC-1.6, FUNC-1.7, FUNC-1.8, FUNC-1.9, FUNC-1.10, FUNC-1.11, FUNC-1.12, FUNC-1.13, FUNC-1.14, FUNC-1.15.
FUNC-2-TEST	Tests whether or not the product meets requirement FUNC-2.
FUNC-2.1-TEST	Tests whether or not the product meets requirement FUNC-2.1.
FUNC-2.2-TEST	Tests whether or not the product meets

	requirement FUNC-2.2.
FUNC-3-TEST	Tests whether or not the product meets requirement FUNC-3.
FUNC-3.X-TEST	Tests whether or not the product meets requirement FUNC-3.1, FUNC-3.2, FUNC-3.3, FUNC-3.4, FUNC-3.5, FUNC-3.6, FUNC-3.7.
FUNC-4-TEST	Tests whether or not the product meets requirement FUNC-4.
FUNC-5-TEST	Tests whether or not the product meets requirement FUNC-5, FUNC-5.1, FUNC-5.2.
GUI-1-TEST	Tests whether or not the product meets requirement GUI-1.
GUI-2-2.1-TEST	Tests whether or not the product meets requirement GUI-2, GUI-2.1.
GUI-2.2-2.3-TEST	Tests whether or not the product meets requirement GUI-2.2, GUI-2.3.
GUI-3-TEST	Tests whether or not the product meets requirement GUI-3.

4.2 Summary of test results

Included in this section is a chart that compiles all of the test results with their project unique identifier. Also contained in the chart are any problems encountered during testing and any notes made during testing:

Identifier	**Result of Test**	**Problems Encountered**	**Notes**
PLAT-1-TEST	Pass Test		
PLAT-1.1-TEST	Pass Test		
PER-1-TEST	Pass Test		Done in 4.76 s
PER-1.1-TEST	Pass Test		Page must load data.js before website renders

FUNC-1-TEST	Not Pass	All data provided except median age of houses	
FUNC-2-TEST	Pass Test		Works within the specified boundaries
FUNC-2.1-TEST	Not Pass	Missing 12 counties on map	Nothing we can do about missing county data.
FUNC-2.2-TEST	Pass Test		Works only within the lower 48 states, Alaska and Hawaii not included.
FUNC-3-TEST	Pass Test		
FUNC-3.X-TEST	Not Pass	All data inquired from user except for states the user is not interested in.	
FUNC-4-TEST			
FUNC-5-TEST	Pass Test		
GUI-1-TEST	Pass Test		
GUI-2-2.1-TEST	Pass Test		
GUI-2.2-2.3-TEST	Pass Test		
GUI-3-TEST	Pass Test		

5 Notes

Included in this section are all terms and acronyms used in this document:

Term	**Meaning**
AngularJS	Angular Javascript web framework

CSCI	Computer Software Configuration Item
SRS	Software Requirement Specification
SDD	Software Design Description
STD	Software Test Description
PLAT	Platform
FUNC	Functional
PER	Performance
GUI	Graphical User Interface

A. Appendixes

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix shall be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes shall be lettered alphabetically (A, B, etc.).

DESCRIPTION/PURPOSE

he Software Test Report (STR) is a record of the qualification testing performed on a Computer Software Configuration Item (CSCI), a software system or subsystem, or other software-related item.

The STR enables the acquirer to assess the testing and its results.

APPLICATION/INTERRELATIONSHIP

Portions of this plan may be bound separately if this approach enhances their usability. Examples include plans for software configuration management and software quality assurance.

The Contract Data Requirements List (CDRL) should specify whether deliverable data are to be delivered on paper or electronic media; are to be in a given electronic form (such as ASCII, CALS, or compatible with a specified word processor or other support software); may be delivered in developer format rather than in the format specified herein; and may reside in a computer-aided software engineering (CASE) or other automated tool rather than in the form of a traditional document.

PREPARATION INSTRUCTIONS

General instructions.

a. Automated techniques. Use of automated techniques is encouraged. The term "document" in this means a collection of data regardless of its medium.

b. Alternate presentation styles. Diagrams, tables, matrices, and other presentation styles are acceptable substitutes for text when data required can be made more readable using these styles.

c. Title page or identifier. The document shall include a title page containing, as applicable: document number; volume number; version/revision indicator; security markings or other restrictions on the handling of the document; date; document title; name, abbreviation, and any other identifier for the system, subsystem, or item to which the document applies; contract number; CDRL item number; organization for which the document has been prepared; name and address of the preparing organization; and distribution statement. For data in a database or other alternative form, this information shall be included on external and internal labels or by equivalent identification methods.

d. Table of contents. The document shall contain a table of contents providing the number, title, and page number of each titled paragraph, figure, table, and appendix. For data in a database or other alternative form, this information shall consist of an internal or external table of contents containing pointers to, or instructions for accessing, each paragraph, figure, table, and appendix or their equivalents.

e. Page numbering/labeling. Each page shall contain a unique page number and display the document number, including version, volume, and date, as applicable. For data in a database or other alternative form, files, screens, or other entities shall be assigned names or numbers in such a way that desired data can be indexed and accessed.

f. Response to tailoring instructions. If a paragraph is tailored out of this document, the resulting document shall contain the corresponding paragraph number and title, followed by "This paragraph has been tailored out." For data in a database or other alternative form, this representation need occur only in the table of contents or equivalent.

g. Multiple paragraphs and subparagraphs. Any section, paragraph, or subparagraph in this DID may be written as multiple paragraphs or subparagraphs to enhance readability.

h. Standard data descriptions. If a data description required by this document has been published in a standard data element dictionary specified in the contract, reference to an entry in that dictionary is preferred over including the description itself.

i. Substitution of existing documents. Commercial or other existing documents, including other project plans, may be substituted for all or part of the document if they contain the required data.