Merquery

Test Plan

Team Green

Amy Nguyen
Jackeline Maldonado
Jacob Darensbourg
Rachita Gehi
Tina Bui

March 2016

Table of Contents

Part I: Software Test Plan 3
1. Introduction **3**
  1.1 Overview of Test Plan **3**
  1.2 Goals/Objectives **3**
  1.3 Constraints **3**
  1.4 Assumptions and Dependencies **3**
  1.5 References **3**
2. Test Items Pass/Fail Criteria **3**
3. Test Scenarios **5**
4. Test Deliverables **15**
Part II: User Test Plan **17**
1. Heuristic Evaluation **17**
  1.1 Overview **17**
  1.2 Heuristic Evaluation Template **17**
  1.3 Results from Heuristic Evaluations **17**
  1.4 References **26**
2. User Testing **27**
  2.1 Scope **27**
  2.2 Purpose **27**
  2.3 Materials **27**
  2.4 User **27**
  2.5 Location **27**
  2.6 Equipment **27**
  2.7 Task List (Scenarios) **28**
  2.8 Observations **28**
  2.9 Test Results **28**
  2.10 References **28**
Part III: Test Report **29**

Part I: Software Test Plan

1. Introduction

1.1 Overview of Test Plan

This document describes the testing for the data exploration tool, Merquery. It will also contain the passing/failing criteria along with various test items for unit/module, integration, scenario/use case, and system testing.

1.2 Goals/Objectives

The main purpose for conducting these test are to verify the usability and performance of Merquery. The functionality and features of the tool will also be tested.

1.3 Constraints

At the moment, the data exploration tool is not yet live and therefore must be used on a laptop with the program.

1.4 Assumptions & Dependencies

Assumptions

- Tester should have access to the database on Google BigQuery (GBQ)

Dependencies

- Authorization credentials must be valid
- Merquery should be able to connect to GBQ

1.5 References

List related documents (link if possible)

- Test Plan | Software Testing Fundamentals
    - http://softwaretestingfundamentals.com/test-plan/

2. Test Items Pass/Fail Criteria

Test Item	**Pass Criteria**	**Fail Criteria**	**Priority**
Display data through a table	Displays all the relevant consumer data that Viant possesses in an orderly tabular format.	No results are displayed.	High

Display Number of Results Count	When data is displayed in a table a total count of rows should be displayed.	No count displayed for results.	Medium
Pagination Functionality	User should be able to limit the number of results per page and also click through different pages of data.	The results are not limited per page as specified by the user.	Medium
Ability to traverse through data	After displaying the initial search, user should be able to click on different data points to explore data. Clicking on different data points should not append the values onto the initial query. A new query should be run.	The query does not change.	High
Breadcrumbs	After displaying the initial search, all the inputs from the left navigation should be displayed at the top.	No user inputs are displayed at the top after the Search button is selected.	High
Export functionality	User selects the export button and the data displayed in the application is exported to an excel spreadsheet.	No data is available on the excel spreadsheet or no spreadsheet is generated.	Medium
Displaying individual data	Individual information should be displayed in the form of a user profile data (in the form of a pop up)	Profile page does not pop up; Page pops up but no information is displayed	High
Loading indicator	When the search button is selected an indicator should be displayed while the query is being processed.	No loading indicator is displayed.	Medium
Help Page	When the help button is selected a pop up is displayed with information on how to use the application.	When the help button is clicked no pop is displayed.	Medium
Gender field	On the left navigation, the user can select male, female, unknown through a dropdown.	When the gender dropdown is clicked nothing happens.	Low
Table should not display sensitive information.	The results table should not display last name information.	The results display last name of all the individuals in the database.	High
Error Handling	If the user enters a string in an integer field and error is thrown.	No error is thrown when the user enters a string in an integer field.	High
Sorting	When the user clicks on the table header, the	When the user clicks	Low

functionality	data should be sorted in an ascending or descending order.	on the table header, nothing is sorted.	
No Results message	Displays “No results” returned message.	No message is displayed.	Medium
Reset Button	When the user inputs values in the left navigation and wants to clear his/her inputs. The user then selects the Reset button.	When the reset button is selected no inputs are cleared.	Low
Checkbox functionality	The user can select which tables can be viewed in the results set. The user selects tables, hits search and views the selected tables.	When the user selects tables, hits search and the selected tables are not displayed.	Low
Extensibility	The user is able to add a column in the database with minimal code changes.	The user adds a new column but the new column is not reflected in the application.	High

3. Test Scenarios

Test Scenario #1	Display data through a table
Priority	High
Requirement Traceability	1.5.2.1.b Display consumer data in a table
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Click the search button or press enter 5. A table will be displayed with results
Pass/Fail	
Comments	

Test Scenario #2	Display Number of Results Count
Priority	Medium
Requirement Traceability	1.5.2.2.b Show the total number of user profiles
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Click the search button 5. A table will be displayed with results 6. Confirm that the results counter is present 7. Ensure the counter is correct by clicking around different pages
Pass/Fail	
Comments	

Test Scenario #3	Pagination functionality
Priority	Medium
Requirement Traceability	1.5.2.2.c Show the table across multiple pages
Execution Steps	1. Load MerQuery 2. Choose how many “Results per page” should be displayed 3. Expand one of the search fields 4. Type in a search criteria (i.e. City: Washington) 5. Click the search button 6. A table will be displayed with results 7. Confirm the table displays with the number of results selected

	8. Use the "Prev, 1, 2" buttons to confirm that the data changes on the table
Pass/Fail	
Comments	

Test Scenario #4	Ability to traverse through the data
Priority	High
Requirement Traceability	1.5.2.1.d Traverse through the data via clicking
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Click the search button 5. A table will be displayed with results 6. Click on a data point in order to traverse 7. Confirm traversal with no errors 8. Confirm that the breadcrumbs are changing with the new search criteria
Pass/Fail	
Comments	

Test Scenario #5	Breadcrumbs functionality
Priority	High
Requirement Traceability	2.1.1.3 Should track user searches and queries through the use of breadcrumbs
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Choose another search field (i.e. Gender: Male) 5. Click the search button 6. A table will be displayed with results 7. Breadcrumbs will be displayed with both criterias 8. Click ‘x’ on one of the breadcrumbs 9. Confirm the results in the table change
Pass/Fail	
Comments	

Test Scenario #6	Exporting functionality
Priority	Medium
Requirement Traceability	Requested during a design session
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Click the search button 5. A table will be displayed with results 6. Click the “Export data” button 7. A spreadsheet with the name “MerQueryData.xls” 8. Open the spreadsheet 9. Confirm that all the data has been exported

Pass/Fail	
Comments	

Test Scenario #7	Individual Profile Pop Up
Priority	High
Requirement Traceability	1.5.2.1.e Display user profile pages
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Click the search button 5. A table will be displayed with results 6. Click the icon next to an EmailMD5 7. A loading indicator will appear on the pop up if the information is still loading 8. Confirm that a pop up appears with the individual's information
Pass/Fail	
Comments	

Test Scenario #8	Loading indicator

Priority	Medium
Requirement Traceability	1.5.2.2.f Visual indicators to demonstrate the system is working, not broken
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Click the search button 5. Ensure that a loading indicator is present 6. Confirm that the indicator disappears once the table is displayed
Pass/Fail	
Comments	

Test Scenario #9	Help Page Functionality
Priority	Medium
Requirement Traceability	1.5.2.2.d Run all within the same web page without opening new tabs 1.5.2.2.e A simple help page to assist first-time users understand the tool
Execution Steps	1. Load MerQuery 2. Click on the Help button 3. Ensure there are no spelling mistakes and that the language is clear for first-time users 4. Click the “Got It” button to close the Help page
Pass/Fail	
Comments	

Test Scenario #10	Gender field is a dropdown field
Priority	Low
Requirement Traceability	1.5.2.3.c Dropdown menus and radio buttons for left navigation search fields
Execution Steps	1. Load MerQuery 2. Expand the Demographics category 3. Select a gender type 4. Click Search 5. Confirm the results have the gender type that was selected
Pass/Fail	
Comments	

Test Scenario #11	Table should not display sensitive information
Priority	High
Requirement Traceability	2.1.1.6 Should not display any names of the individuals in the database, only encrypted email addresses (MD5)
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Click the search button 5. A table will be displayed with results 6. Confirm there is no sensitive information (i.e. First Name, Last Name)

Pass/Fail	
Comments	

Test Scenario #12	Error Handling
Priority	High
Requirement Traceability	2.2.7.1 Users should be notified when there is an invalid query
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in an invalid search criteria (i.e. Campaign Click Count: testing) 4. Click the search button 5. An friendly error message should be displayed
Pass/Fail	
Comments	

Test Scenario #13	Sorting functionality
Priority	Low
Requirement Traceability	Requested during a design session

Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Click the search button 5. A table will be displayed with results 6. Click on one of the headers in order to sort the data 7. Confirm that the data is sorted
Pass/Fail	
Comments	

Test Scenario #14	No Results message
Priority	Medium
Requirement Traceability	Requested during a design session
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in an invalid City criteria (i.e. City: Goshington) 4. Click the search button 5. “No Results” message should be displayed
Pass/Fail	
Comments	

Test Scenario #15	Reset Button
Priority	Low
Requirement Traceability	Suggested during Usability Testing I
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Click the search button 5. A table will be displayed with results 6. Click the “Reset” button 7. Confirm that all the values entered by the user in the left navigation have cleared
Pass/Fail	
Comments	

Test Scenario #16	Checkbox functionality
Priority	Low
Requirement Traceability	Suggested during Usability Testing I
Execution Steps	1. Load MerQuery 2. Expand one of the search fields 3. Type in a search criteria (i.e. City: Washington) 4. Using the checkboxes, select which tables you would like to see 5. Click the search button 6. A table will be displayed with results
Pass/Fail	
Comments	

Test Scenario #17	Extensibility - Left navigation should be dynamic
Priority	High
Requirement Traceability	1.5.2.1.c Left Navigation fields change according to the information contained within GBQ
Execution Steps	1. Open GBQ with admin permissions 2. Add a column of data to any of the six tables (demo_info, purchase_info, device_info, address_info, campaign_info, behavior_info) 3. Open the Constants.js file 4. Add the new column name next to the “columnNames” variable under the appropriate category. 5. Add a new placeholder for the new column next to the “placeholder” variable under the appropriate category. 6. Refresh MerQuery 7. Expand the category where the new column was added 8. Confirm the addition of a new field in the left navigation
Pass/Fail	
Comments	

4. Test Deliverables

- **Test Plan:** This Document
- **Test Cases:** Included in “test scenarios” above

- **Test Scripts:** Included in “test scenarios” above
- **Test Reports:** “Test results” included below
- **Defect/Enhancement Logs:** Included below

Defect/Enhancement log

updated as needed

Defect (D) /Enhancement (E)	Completed/Corrected (Yes/No)	Explanation
Search button clicked when “enter key” pressed (E)	**Yes**	Implemented functionality
Add Placeholder values in left-nav so user is aware of expected input (E)	**Yes**	Receive placeholder values and display them on left-nav
Conversion of timestamp into date for birthdate field (E)	**In progress**	Working on using function to convert timestamp
Export button download a file with related name (E)	**Yes**	Downloaded file now named “MerqueryData”
Loading Indicator for pop-up (E)	**Yes**	Added loading indicator
Help button/pop up hidden behind left nav (D)	**Yes**	Fixed positioning
No Error Messages (D)	**Yes**	Added friendly error messages
Traversing feature not right concept/ new query generated instead of drill down (D)	**Yes**	Gained clarity on concept of traversing, implemented functionality
Not all fields on left-nav searchable (D)	**Yes**	Fixed queries
No way to clear all fields in left nav (E)	**Yes**	Added reset button
popup not extensible (D)	**in progress**	Working on extensibility, new designs
No household card (D)	**in progress**	Working on design

Part II: User Testing Plan

Phase 1: Heuristic Evaluation

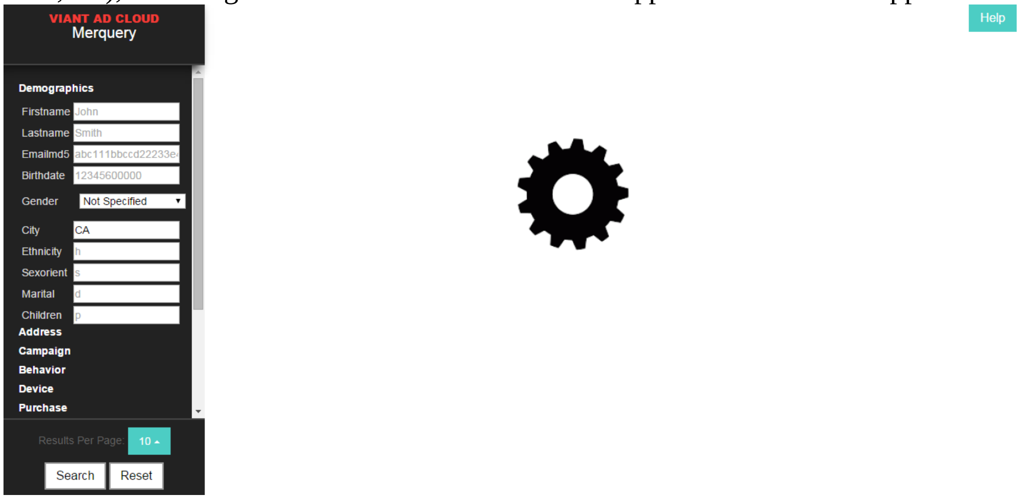
1.1 Overview

The heuristic evaluations will be based on Jakob Nielsen’s ten heuristic values.

1.2 Heuristic Evaluation Template

The following template from Carnegie Mellon University will be used to evaluate the different principles from Jakob Nielsen’s ten heuristic values. We will be running Merquery on our local computers to examine each point.

1.3 Results from Heuristic Evaluations

No. HE 1	**Good Aspect**
Name: The state of the system	
Evidence: Heuristic: Visibility of system status Interface aspect: Whenever there is a loading period for Merquery (e.g. loading the popup, results to the table, etc), a loading indicator will show and then disappear when the data appears on the screen.  VIANT AD CLOUD Merquery Demographics Firstname John Lastname Smith Emailmd5 abc111bbccd22233e Birthdate 12345600000 Gender Not Specified City CA Ethnicity h Sexorient s Marital d Children p Address Campaign Behavior Device Purchase Results Per Page: 10 Search Reset Help	
Explanation: When the user enters in a query to search, a loading indicator will appear to illustrate that the user must	

wait while the system processes the information that is needed to be displayed. By showing the indicator, users will be less likely to think the system is broken since there is a mildly long wait time for information to process. Therefore, they will probably end up waiting for things to load rather than clicking on various things trying to figure out why nothing is showing.

Severity or Benefit: Benefit
Rating: N/A
Justification (Frequency, Impact, Persistence, Weights):
Frequency: This situation is quite common since the system requires a lot of time to process. Regardless of what experience the user has, the absence of the loading indicator will make users question the state of the system and attempt to try anything to fix it or just close the program.
Impact: This issue is easy to overcome because the loading indicator will appear and disappear automatically between processes.
Persistence: This will arise each time GBQ is processing a query. Over time, they may grow tired of the wait time while the information loads.
How I weighted the factors: The benefits the user sees from this aspect is the knowledge that there are processes running in the background that are retrieving the information the user asked for.

Possible solution and/or Trade-offs:
The sole tradeoff for this aspect is that users will be looking at the loading indicator for about half a minute.

Relationships:

No. HE 2	**Good Aspect**

Name: Following real world conventions

Evidence:
Heuristic: Match between system and the real world
Interface aspect: On all the buttons, there are descriptive words which tell the user their function. The fields are also written in a way for users to understand what they are looking for. Messages about errors are also written in a way for them to comprehend.

Explanation:
Throughout the interface, the system uses the user's language. If the users want to search for people who live in the city of Irvine. They just have to go to demographics and type "Irvine" in the city field. Then to submit their query, they just have to click "Search." The table is also displayed in a way for easy understanding.

Severity or Benefit: Benefit
Rating: N/A
Justification (Frequency, Impact, Persistence, Weights):
Frequency: For the most part, this is quite common since most interfaces are written so that the users can understand the language it is in. In this case, Merquery only supports the English language (no computer language is shown on the UI side).
Impact: Once the users understand the flow of the system, they will have no problem in searching what they need to search.
Persistence: The only issue that may arise is that users have trouble finding what they are looking for. However, this is usually a one time issue.

How I weighted the factors: The benefits of having things written in the vernacular rather than computer terms allows the user to intuitively understand how to do things that they want to do (e.g "Search" button will send what the users want to search).

Possible solution and/or Trade-offs:
For this current situation, there is no trade-off.

Relationships:

No. HE 3 **Problem**

Name: Undo

Evidence:
Heuristic: User control and freedom
Interface aspect: Users has no way to stop their search query once they click search

Explanation: The user fills out the left navigation side with the fields they wish to search and the tables they wish to see. However, after clicking search, the user realizes that there is no way to stop the search because they have entered the wrong field.

Severity or Benefit: Severity
Rating: 1
Justification (Frequency, Impact, Persistence, Weights):
Frequency: This is quite common for both inexperienced and experienced users because they might want to stop their query since they have entered the wrong fields and want to go back.
Impact: This problem is not too difficult to overcome since the user can always type in the search fields again and click search which will send in a new query.
Persistence: This problem can happen more than once. They will not be bothered by this issue that much because there is still a way to stop their search by searching again.
How I weighted the factors: Based on the weighted factors, it would be nice to have a way to undo their search. If they sent in a wrong search, the loading indicator will still continue moving making the user feel pressured to quickly fix their search. However, if there was a way to stop that search, users can take their time in changing what they need.

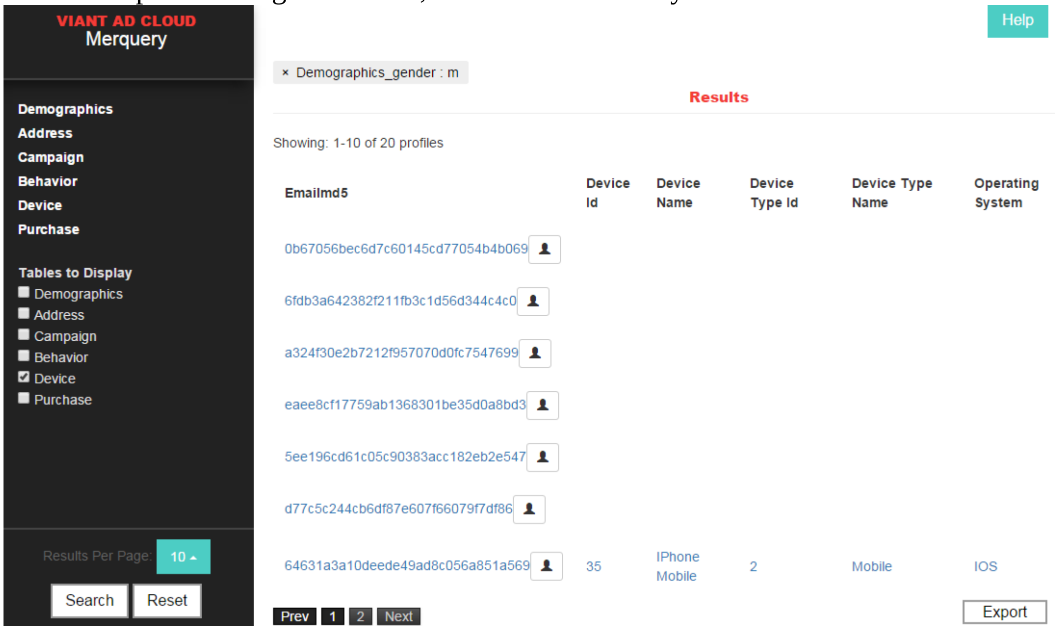
Possible solution and/or Trade-offs:
A solution for this issue is to have a cancel or abort button while the query is loading to allow the user to stop their query.

Relationships:

No. HE 4 **Good Aspect**

Name: Button Similarity

Evidence:
Heuristic: Consistency and standards
Interface aspect: Throughout the UI, there is a consistency on what a button is and looks like.



× Demographics_gender : m

Results

Showing: 1-10 of 20 profiles

Emailmd5	Device Id	Device Name	Device Type Id	Device Type Name	Operating System
0b67056bec6d7c60145cd77054b4b069					
6fdb3a642382f211fb3c1d56d344c4c0					
a324f30e2b7212f957070d0fc7547699					
eaee8cf17759ab1368301be35d0a8bd3					
5ee196cd61c05c90383acc182eb2e547					
d77c5c244cb6df87e607f66079f7df86					
64631a3a10deede49ad8c056a851a569	35	IPhone Mobile	2	Mobile	IOS

Explanation: Although each button has a different functionality, the size, shape, and color of the buttons are consistent.Therefore, the user will quickly learn to recognize what can be clicked.

Severity or Benefit: Benefit
Rating: N/A
Justification (Frequency, Impact, Persistence, Weights):
Frequency: Users will always encounter buttons in this system in order to carry out what they want. They will use the buttons when searching, exporting, resetting, etc.
Impact: After the initial use of this system, users will be able to quickly identify where the buttons are and what they do.
Persistence: This is only a one time issue as they will get used to what the UI looks like for this interface.
How I weighted the factors: The benefits of consistent buttons allow the users to need to learn only once. With consistency, they will be able to interact with the system with ease since they know what to expect.

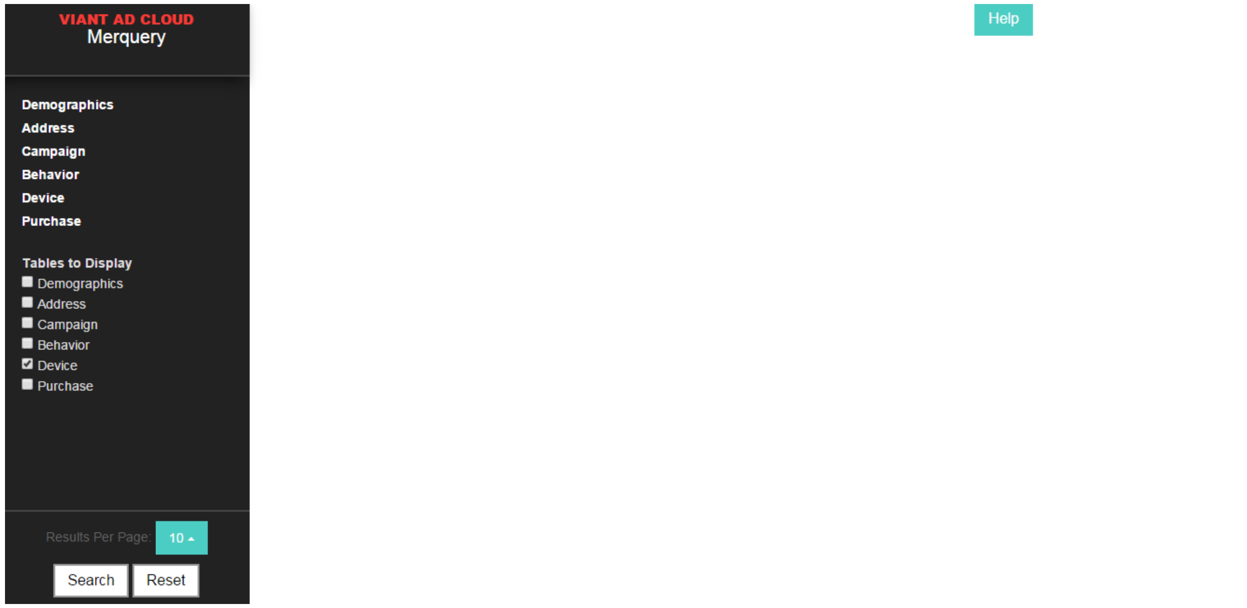
Possible solution and/or Trade-offs: There are no trade-offs for this particular heuristic.

Relationships:

No. HE 5	**Problem**

Name: Empty Search

Evidence:
Heuristic: Error prevention
Interface aspect: Nothing shown when the user clicks search.



Tables to Display

☐ Demographics
☐ Address
☐ Campaign
☐ Behavior
☑ Device
☐ Purchase

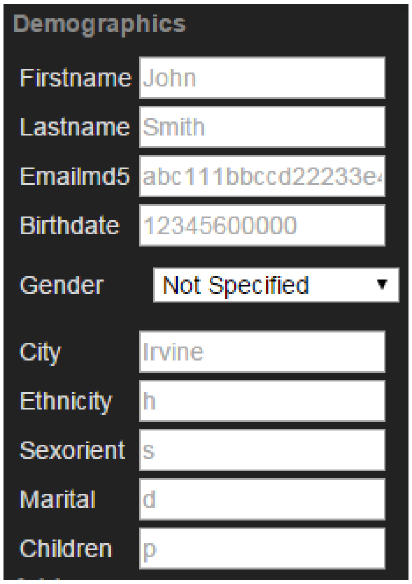
Explanation: The user does not fill out any of the fields, but unchecks some checkboxes. When he/she clicks search, nothing shows up. Numerous clicks still show the user nothings.

Severity or Benefit: Severity
Rating: 2
Justification (Frequency, Impact, Persistence, Weights):
Frequency: This is quite common since some users may just end up clicking search and they forget to add things in the fields.
Impact: This problem will be difficult to overcome if there is no indication or clue as to what the user is doing incorrectly.
Persistence: This problem usually arises one-time after the user figures out that they left the search fields blank. Then in the future, they will recall their mistake and end up remembering to add in fields.
How I weighted the factors: Although users will be able to learn their mistake after the first try, it is always better to inform the users of what is happening in the system rather than having them guess.

Possible solution and/or Trade-offs: A possible solution to this issue is to display a friendly message saying that the search fields are empty.

Relationships: No. HE 9

No. HE 6	**Good Aspect**
Name: Placeholders	
Evidence: Heuristic: Recognition rather than recall	



Demographics

Firstname	John
Lastname	Smith
Emailmd5	abc111bbccd22233e
Birthdate	12345600000
Gender	Not Specified ▾
City	Irvine
Ethnicity	h
Sexorient	s
Marital	d
Children	p

Explanation:
Each input box for searching contains a placeholder to indicate what kind of information each field takes. Once the user starts typing in the text field, the gray text will disappear. When the field is empty, the placeholder reappears to remind users how the input should be formatted.

Severity or Benefit: Benefit
Rating: N/A
Justification (Frequency, Impact, Persistence, Weights):
Frequency: The placeholders will always be in the text fields until users start entering their input.
Impact: Users will be given an idea of what type of data is allowed for each input field, which is important for error prevention.
Persistence: When users start writing in an input field, the placeholder text will disappear. Once the field is cleared or empty, the placeholder text will reappear.
How I weighted the factors: Error prevention and clarity for users is important.

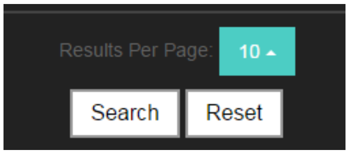
Possible solution and/or Trade-offs:
A problem with having set placeholder texts is that sometimes, the text is not clear so users will not know what "h" in "Ethnicity" would stand for or something similar.

Relationships:

No. HE 7	**Good Aspect**

Name: Reset

Evidence:
Heuristic: Flexibility and efficiency of use
Interface aspect: Users can click on the "Reset" button to return the left navigation to its original state.



Results Per Page: 10

Explanation:
Suppose a user types in multiple fields in the left navigation bar and realizes that these weren’t what he/she wanted to search for. Instead of going back and checking each section and deleting, he/she can click the reset button to clear the fields. This resets the left navigation to how it was when Merquery first loads.

Severity or Benefit: Benefit
Rating: N/A
Justification (Frequency, Impact, Persistence, Weights):
Frequency: This is quite common for both inexperienced and experienced users because they might want to run a new search and a quick and easy way to clear the fields would be resetting them.
Impact: The reset button simplifies the process of erasing the inputted fields which saves a user’s time.
Persistence: Once the users understand what the reset button is for, it makes it easier for users to overcome the problem of clearing fields.
How I weighted the factors: The benefits of using the reset allows the user to quickly search for a new profile after the user completed his/her initial search.

Possible solution and/or Trade-offs:
The trade-off for this is that users will not be able to pick and choose what is reset. For instance, if they just want to reset one category that they filled out, the reset button will not allow them to. In addition, there is no verification on whether the user wants to reset or not (in the case where the user accidentally clicks “reset” instead of “search”.

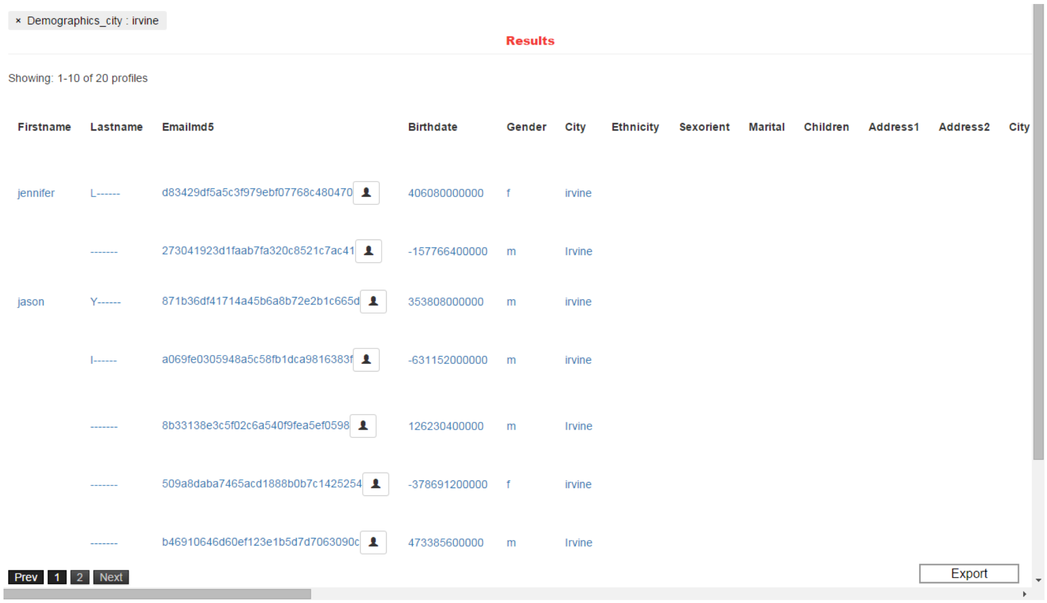
Relationships:

No. HE 8	**Problem**

Name: Results table size

Evidence:
Heuristic: Aesthetic and minimalist

design



× Demographics_city : irvine

Results

Showing: 1-10 of 20 profiles

Firstname	Lastname	Emailmd5	Birthdate	Gender	City	Ethnicity	Sexorient	Marital	Children	Address1	Address2	City
jennifer	L------	d83429df5a5c3f979ebf07768c480470	406080000000	f	irvine							
	-------	273041923d1faab7fa320c8521c7ac41	-157766400000	m	Irvine							
jason	Y------	871b36df41714a45b6a8b72e2b1c665d	353808000000	m	irvine							
	I------	a069fe0305948a5c58fb1dca9816383f	-631152000000	m	irvine							
	-------	8b33138e3c5f02c6a540f9fea5ef0598	126230400000	m	Irvine							
	-------	509a8daba7465acd1888b0b7c1425254	-378691200000	f	irvine							
	-------	b46910646d60ef123e1b5d7d7063090c	473385600000	m	Irvine							

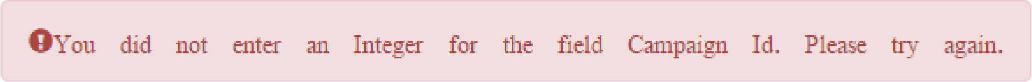
Explanation:
Because there is so much data to display in the table, the table scrolls horizontally for a long distance. This causes users to need to scroll far to view additional information, and because everything cannot be displayed in one screen, it might cause confusion for the users.

Severity or Benefit: Severity
Rating: 1
Justification (Frequency, Impact, Persistence, Weights):
Frequency: The problem is frequent because once the user searches for more than one category's worth of data, the table would be too wide to fit onto the screen.
Impact: Users might have problems trying to remember all of the data within a row that they are looking at or what information is hidden when they scroll past it.
Persistence: The problem is persistent because the purpose of the system is searching for data, and how the data is displayed is in the table.
How I weighted the factors: While it is a minor hinderance to have such a large table, functionality is not affected.

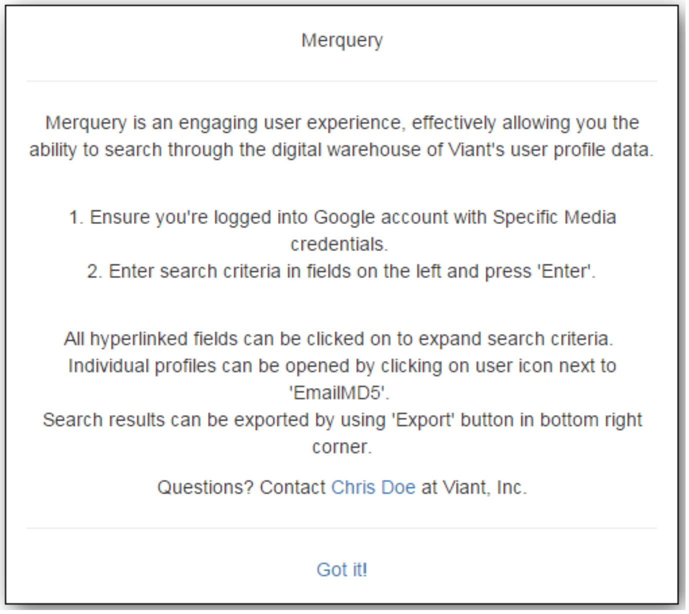
Possible solution and/or Trade-offs:
One possible solution is to consolidate data or completely delete some columns out. However, this might cause a bloated table or missing important information for the users.

Relationships:

No. HE 9	**Good Aspect**
Name: Error messages	

Evidence: Heuristic: Help users recognize, diagnose, and recover from errors  You did not enter an Integer for the field Campaign Id. Please try again.
Explanation: An error message will be displayed when users enter in the wrong input into a text field. Users will be notified what kind of input they put in and what the expected input for the field should be.
Severity or Benefit: Benefit ***Rating:*** N/A **Justification (Frequency, Impact, Persistence, Weights):** ***Frequency:*** Whenever users enter in the wrong input for a field, an error message will be displayed to tell users why their input was incorrect. ***Impact:*** Users will be able to recognize what is wrong with their input and the state of the system, which is important for them to recover from the error. ***Persistence:*** Errors will appear every time users enter in the wrong type of input (e.g. an integer when a field expects a string). ***How I weighted the factors:*** If there are no errors, users will not know why their queries did not work. Having understandable errors is important for users to recover from them and know not to do in the future.
Possible solution and/or Trade-offs: The errors might be confusing for users, so clearly listing the reasons why the input failed would be better in this case.
Relationships: No. HE 5

No. HE 10	**Good Aspect**
Name: Help page	
Evidence: Heuristic: Help and documentation	

 Merquery Merquery is an engaging user experience, effectively allowing you the ability to search through the digital warehouse of Viant's user profile data. 1. Ensure you're logged into Google account with Specific Media credentials. 2. Enter search criteria in fields on the left and press 'Enter'. All hyperlinked fields can be clicked on to expand search criteria. Individual profiles can be opened by clicking on user icon next to 'EmailMD5'. Search results can be exported by using 'Export' button in bottom right corner. Questions? Contact Chris Doe at Viant, Inc.
Explanation: When the user clicks on the help button, a pop-up will appear with instructions on how to use the system. Users will be given steps on how to start using the system, including how to authenticate and how to use the tool. Users with further questions are given an email to send their questions to.
Severity or Benefit: Benefit ***Rating:*** N/A **Justification (Frequency, Impact, Persistence, Weights):** ***Frequency:*** Because this is an internal tool, not many new users will be using the system. Once all users are familiar with the layout and how to use the system, there is little need for the help page. ***Impact:*** The help page is beneficial for quick answers to simple questions, and if users have a more complicated question not answered in the help page, there is an email provided. ***Persistence:*** Once users are familiar with using the system, they will no longer need the help page. However, if there are any questions that users still have, the help page offers an email. ***How I weighted the factors:*** The benefit of the help page is that users can quickly look up how to use the tool.
Possible solution and/or Trade-offs: Having a very simple help page might not offer enough help for the users, and having a specific person's email to redirect questions towards might inconvenience the person.
Relationships:

1.4 References

- 10 Usability Heuristics for User Interface Design
  - https://www.nngroup.com/articles/ten-usability-heuristics/
- UAR Template - Carnegie Mellon University
  - https://www.cs.cmu.edu/~bam/uicourse/UARTemplate.doc

Phase 2: User Testing

2.1 Scope

This test will be conducted on the data exploration tool, Merquery. Our tests will cover the variety of functions that Merquery is capable of. Most tests will be directly extracted from the requirements, with the remainder covering different aspects that have been added to our system (and may not have been part of the original requirements). We will conduct testing with potential users and have them evaluate our product and provide feedback.

2.2 Purpose

The purpose of this test is to determine the usability functionality of the tool. The following is a list of goals for this test:

- If users are unsure as to where to start, can they find a place where they can receive more information?
- Can users run a simple query?
- Can user run a specific query and get desired results?
- Are users able to sort the table?
- Can users navigate through the table to refine their search further?
- Can users access certain profile pages?
- Can users navigate back on their search?
- Are user needs being met? Is the system intuitive and easy to use?

In receiving feedback based on the goals for this test, we will move forward with fixing any present issues and consider the feasibility of new requirements/updated requirements that emerge during testing.

2.3 Materials

The following materials will be used for usability testing:

- Task sheet (one per tester)
- Laptops (to run our program)
- Notepad for observations
- Test results sheet

2.4 Users

The users will be people who work at Viant, potential users of our product. The aim is to have 3-5 users to test the data exploration tool. We aim to have Viant employees from a variety of fields, this will allow us to have opinions from different perspectives within the company.

2.5 Location

All testing is to be conducted at Viant. Times are not yet determined, but will be scheduled a week in advance.

2.6 Equipment

Depending on the number of users, there will be one laptop for each of them. Given that we are a team of five people, we will have a maximum number of five laptops. If the test users exceed the amount of laptops provided we will have two users per laptop.

2.7 Task List (Scenarios)

Included above in "Test Scenarios" section

2.8 Observations

Tests will be conducted by team members. Team members will observe Viant employees using the product and will write down any related observations.

2.9 Test Results

Test results will be collected from the sheets/scenarios handed out to and filled out by the test users. These results will be combined with observations and copied onto a google doc for viewing of all team members.

Results updated in the following doc: TestReportResults.pdf (included in Team Green Final Assignments folder)

2.10 References

- Sample Usability Test Plan
  - http://it.toolbox.com/blogs/<wbr />enterprise-solutions/sample-<wbr />usability-test-plan-17826
- System Usability Scale (SUS)
  - http://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html

Part III: Test Report

Listed below are the results collected from user testing on February 17,2016 and February 24, 2016

1. Left navigation

Feedback from user	Action Taken
Clicked enter, as is common in most pages, and program didn't search for results	Search button should be clicked when the "enter" key is pressed
Not made clear to user that left-nav is accordion style	Visual indication to show that left nav expands→ Hover functionality
Not intuitive to have search button on top	Moved search button to bottom
Unclear expectations for input in left-nav fields	added placeholders to make input expectation clear
Birthdate unclear in current format (timestamp)	(in progress) MSEC_TO_TIMESTAMP conversion for birthdate in backend, front end: add "from and to" allowing user can enter date range
No Error when all fields are empty and "search is clicked"	(in progress) Add Error message

2. Pagination

Feedback from user	Action Taken
If user wants to move through pages quickly, has to scroll all the way to bottom of page	Made pagination nav a fixed footer
Color scheme isn't intuitive	stylesheet changes

3. Export button

Feedback from User	Action Taken
Incorrect file extension displays when using a Windows computer (extension is .xlt -- throws an extension message to the user where they have to click ‘Yes’, but the file loads correctly)	Corrected the file so that the .xls/.xlsx extension is attached to the file and no longer requires user to click ‘Yes’ in order to download the results
File name is not descriptive	Changed the file name to “MerqueryResult”

4. Loading indicator

Feedback from User	Action Taken
There should be a loading indicator when the profile pop-up is being generated	Added a rotating gear .gif file that displays when the user clicks on the profile icon
Loading indicator isn’t centered when the pop-up is being displayed	IN PROGRESS - changing the stylesheet to ensure the indicator shows up in the correct location
Loading indicator should be consistent throughout (table results, pop-up)	Change the pointer of the loading indicator to refer to the same .gif and delete the unused version

5. Help page

Feedback from User	Action Taken
Can’t read all of the text, the help box is partially hidden by left nav	Resized the help window to scale with the page, no longer at fixed dimensions
Too much text	Removed some unnecessary text and shortened the number of instructions to increase readability

6. No Results message

Feedback from User	Action Taken
If there are no results, a message should be displayed	Added a friendly error message that displays ‘No Results’ when nothing is returned by the query

7. Breadcrumbs

Feedback from User	Action Taken
Breadcrumbs from previous searches display while new searches are going	IN PROGRESS - remove the breadcrumbs from previous search as soon as new search is created
Closing all of the breadcrumbs causes the code to break	IN PROGRESS - change the code so that when there are no breadcrumbs, the table is no longer displayed (no error message is needed)

8. Pop up

Feedback from User	Action Taken
Scrolling bug - user has to click out and then click back on profile to make sure the popup scrolls correctly	IN PROGRESS - change modal from third party plug-in to Bootstrap plug-in, should correct the problem
Too much data displaying on popup	Condensed the data down, limited the number of results returned from the query to ‘5’
“Undefined” shows up when there is no data in certain sections	IN PROGRESS - no action taken yet; “No Data” should be displayed when there is nothing in the result set
Resize the pop up to fit the window	IN PROGRESS - no action taken yet
Center the pop up	IN PROGRESS - no action taken yet
Data categories are off on the side - makes it difficult to understand what is being read;	IN PROGRESS - center the results and make the header bold so that it’s easier to understand what is showing
Scrolling in the background is enabled while the popup is showing, it should be disabled	IN PROGRESS - change popup z-axis so that it is the only thing on top and everything beneath it is disabled
Some sections are too long and should be scrollable	IN PROGRESS - edit the way the table sections function so that overflow after a particular length can be scrolled

9. **Error handling**

Feedback from User	Action Taken
Error messages should be more user friendly	Edited text so that it’s not the generic result given by Google BigQuery for errors

“Results” table title displays even though there are no results being returned	Edited code so that the table doesn’t display when there are no results being returned
User isn’t told what type of data to enter into search fields to actually get results	Error messages thrown also give help text to the user by telling them what types of entries need to be put into the field (string with example, integer with example)
No error message for when the user fails to enter anything into the search fields	System shows an error message when nothing gets entered into any of the fields

10. Query bugs

Feedback from User	Action Taken
Typing city in the ‘Address_City’ field doesn’t work	Made edits to code so that the field can deal with upper and lowercase entries

11. Sorting

Feedback from User	Action Taken
if it’s an integer like for birthdate, but it’s string and integers in the column, then the sorting varies like for Device id	IN PROGRESS - no action taken yet

12. UI Fixes

Feedback from User	Action Taken
Text coming out of the bottom of the ‘Export’ button	Resized the button so that text isn’t coming out of the bottom
Users don’t know that the table can be sorted	IN PROGRESS - Add arrows to each of the column headers in the results table so that user knows they can be searched
Sorting arrows aren’t spaced out enough away from the column headers	Add padding to the sorting arrows in the header
‘Export’ button is difficult to locate	IN PROGRESS - move ‘Export’ button up to the top of the page next to the ‘Help’ button

‘Help’ button color is strange	Recolor button
Page fonts are inconsistent (Arial, Arial Black, and Helvetica Neue)	Change the fonts
Result count should be in red and display underneath the “Results” header	IN PROGRESS - move the result count and recolor text
Breadcrumbs merge together at certain page resolutions	IN PROGRESS - check resolution of page higher resolution displays
When the user scrolls to the left and right to see the results table and then the user searches for more values in the left nav, the frame on the right side that displays the table should be resize so the user doesn’t have to scroll all the way to the left again.	IN PROGRESS - no action taken yet
Line beneath “Results” header is too small and doesn’t extend all the way to the end of the table	IN PROGRESS - resize the line break and extend to the end of the table

13. Traversal

Feedback from User	**Action Taken**
Fails for birthdate, Basically when results are shown on a table and then birthdate is clicked traversal doesn’t work an error message shows	IN PROGRESS - changing birthdate from msec to timestamp