

Table of Contents

Traceability and Acceptance Criteria.....	2-7
Application.....	8-9
Design Artifacts	10-14
Storyboard	10
Scenarios.....	11-12
UML Class Diagram	13
UML Activity Diagram	14
Testing	15-33
General Testing	15-16
Junit Tests	17-18
Tests Against Acceptance Criteria.....	19-28
Tests Against Scenarios.....	29-33
Sprint Review.....	34-36
Current Increment	34-35
Reflection on Feedback	36

Traceability & Acceptance Criteria

Throughout our document, you will see small circles with the User Story Index (like the one on the left) which is so we could link our design artefacts and application back to the user stories that created it. This table below is for traceability, where each of the user stories is traced back to the user requirement that inspired it, as well as the user story task itself. Alongside, there are the acceptance criteria we created for each user story, which is a specific condition of what our program needed to achieve in order for us to say that the user story is “done”.

User Story Index	User Requirement It Is Based Off	Increment	User Story	Acceptance Criteria
1	1	1	I want <it to read my CSV files>	Users can choose a folder with the data files in CSV files can be loaded into the application The CSV files can be read into the application
2	2	1	I want <a key metrics list>	Users can choose a folder with the data files in CSV files can be loaded into the application A list of metrics can be calculated from that data A list of metrics can be displayed from that data
3	3	1	I want <charts of rates over time>	Users can choose a folder with the data files in CSV files can be loaded into the application A graph (which compares a statistic to time) can be created A graph can be viewed
4	5	1	I want <histograms of click costs>	Users can choose a folder with the data files in CSV files can be loaded into the application A histogram of click costs can be created A histogram of click costs can be viewed
5	4	2	I want <to change the time granularity>	Users can choose a folder with the data files in CSV files can be loaded into the application A graph (which compares a statistic to time) can be created A graph can be viewed The time axis can be changed The graph that is viewed can be updated due to the time axis

User Story Index	User Requirement	Increment	User Story Task	Acceptance Criteria
It Is Based Off				
6	6	2	I want <to filter the statistic metrics>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>A list of metrics can be calculated from that data</p> <p>A list of metrics can be displayed from that data</p> <p>The user can filter which data is used for these metrics</p> <p>A list of new metrics can be re-calculated from the data and those filters</p> <p>The list of metrics displayed can be updated to the re-calculated metrics</p>
7	6	2	I want <filterable graphs>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>A graph (which compares a statistic to time) can be created</p> <p>A graph can be viewed</p> <p>The user can filter which data is used for this graph</p> <p>The graph can be re-created from the data and those filters</p> <p>The graph displayed can be updated to the re-created graph</p>
8	7	2	I want <to compare metrics>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>Two lists of metrics can be calculated from that data</p> <p>Two lists of metrics can be displayed from that data</p> <p>The user can filter which data is used for either of the lists of metrics</p> <p>A list of new metrics can be re-calculated from the data and those filters</p> <p>The list of metrics displayed can be updated to the re-calculated metrics</p>
9	7	2	I want <to compare graphs>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>Two graphs (which compare a statistic to time) can be created</p> <p>Two graphs can be viewed</p> <p>The user can filter which data is used for either of the graphs</p> <p>The graph can be re-created from the data and those filters</p> <p>The graph displayed can be updated to the re-created graph</p>

User Story Index	User Requirement	Increment	User Story Task	Acceptance Criteria
It Is Based Off				
10	8	2	I want <to redefine “bounce”>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>A list of metrics can be calculated from that data</p> <p>A list of metrics can be displayed from that data</p> <p>A user can update the calculation of bounce by the number of pages visited</p> <p>A user can update the calculation of bounce by the time spent on a page</p> <p>The list of metrics displayed can be updated with the re-calculated bounce rate</p>
11	9	ALL	I want <fast and responsive software>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>The CSV files can be read into the application</p> <p><i>The system loads quick enough that the user does not get frustrated by waiting a long time (to be checked with the supervisor if it is up to his and the company he represents standards)</i></p>
12	9	1	I want <it to handle large datasets>	<p>Users can choose a folder with very large data files in CSV files can be loaded into the application in a promptly matter</p> <p>The system can complete all other functionality even with a much larger csv file set</p>
13	1	ALL	I want <aesthetically pleasing GUI>	<p><i>The system is appealing to look at and use, and functionality is easy to deduce (to be checked with the supervisor if it is up to his and the company he represents standards)</i></p>
14	2	3	I want <to easily find and remove anomalies>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>The system can examine all of the data</p> <p>The system can remove any lines that could not be accurate</p> <p>The updated data can be used for all the calculations</p>

User Story Index	User Requirement	Increment	User Story Task	Acceptance Criteria
It Is Based Off				
15	2	3	I want <well known common statistics>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>A list of extra metrics can be calculated from that data (these metrics being useful commonly used metrics for statistical analysis)</p> <p>The extra metrics can be displayed from that data</p>
16	1	1	I want <easy access to recent files>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>The CSV files can be read into the application</p> <p>The application can then be closed, and re-opened</p> <p>Users can then choose a shortcut with the previous folder location</p> <p>Those CSV files can be loaded into the application</p> <p>The CSV files can be read into the application</p>
17	Extension 5	3	I want <to customise the appearance>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>Users can choose to open a menu to change the appearance</p> <p>Users are able to change the style mode to a dark theme</p> <p>Users are able to change the style mode to a high contrast theme</p> <p>These preferences will be saved and used throughout the program</p>
18	Extension 3	1	I want <it to save graphs to a file>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>A graph (which compares a statistic to time) can be created</p> <p>A graph can be viewed</p> <p>A graph can be saved onto the computer hard drive</p>
19	Extension 5	3	I want <a help button>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>The CSV files can be read into the application</p> <p>Users can get tooltips on actions on buttons on the software</p>

User Story Index	User Requirement	Increment	User Story Task	Acceptance Criteria
It Is Based Off				
20	Extension 1	1	I want <to load multiple csv files>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application The CSV files can be read into the application</p> <p>Inside the application, the user can choose to load another (or the same) dataset</p> <p>CSV files can be loaded into the application The CSV files can be read into the application</p> <p>The main screen can be created again</p> <p>This second main screen can be viewed in a separate window/tab</p>
21	6	3	I want <a search bar>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>The user can filter which data is used for the metrics/graph</p> <p>The user can save these filters to be able to quickly reuse them later on</p> <p>The user can search through saved filters by name to find one efficiently</p>
22	1	3	I want <an executable file>	Users can open this application by just using an executable file
23	6	2	I want <a favourites button>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>The user can filter which data is used for the metrics/graph</p> <p>The user can save these filters to be able to quickly reuse them later on</p>
24	1	2	I want <to view the csv file in app>	<p>Users can choose a folder with the data files in CSV files can be loaded into the application</p> <p>The CSV files can all be viewed on the application in an excel format</p>

For the acceptance criteria:

- ➔ Any steps in bold are specific to that particular user story's success, and they may be requirements in other user stories acceptance criteria, but are the key ones in that specific user story
- ➔ Any steps in italics are opinionated and are to be checked with the supervisor if it is up to his and the company he represents standards

4. REQUIREMENTS

The mandatory requirements are as follows:

- 1) The system should be able to read in data relating to a campaign given the three input files described above.
- 2) The system should be able to compute and display the following key metrics of a campaign:
 - a. Number of Impressions
 - b. Number of Clicks
 - c. Number of Uniques
 - d. Number of Bounces
 - e. Number of Conversions
 - f. Total Cost
 - g. CTR
 - h. CPA
 - i. CPC
 - j. CPM
 - k. Bounce Rate
- 3) The system should be able to display charts of these metrics over time. One data point on these charts should correspond to the metric computed over a specific time interval (e.g., one hour, one day or one week).
- 4) The user should be able to control the time granularity of these time charts.
- 5) The system should be able to display a histogram of the click costs (showing the distribution of costs).
- 6) The user should be able to filter the metrics and charts by:
 - a. Date range
 - b. Audience segments (by gender, age or income)
 - c. Context
- 7) There should be some functionality to directly compare metrics or charts with different filters applied to them (e.g., to compare the CTR within one audience segment to another, or to compare the number of uniques within two distinct time periods). This could be achieved by overlaying graphs or by displaying multiple graphs next to each other.
- 8) The user should be able to define how a bounce is registered (either depending on the time spent on the website or on the number of pages visited).
- 9) The software should be fast and responsive. It should be able to handle data sets containing millions of impressions and spanning several months. Once the initial data is loaded, updating the graphs (e.g., by applying filters or switching metrics) should take no more than a few seconds.

User Requirements For Reference

5. OPTIONAL EXTENSIONS

You are free to implement additional extensions beyond the requirements above. These should be discussed with your supervisor, but could include:

- 1) Ability to load and compare data from multiple campaigns.
- 2) Chart displaying the performance metrics per time of day or per day of the week.
- 3) Facility for saving summary charts to file (as an image or pdf file).
- 4) Printing functionality.
- 5) Ability to customise the appearance of the application (e.g., by modifying colour schemes or font sizes).

Application

22

	List 1	List 2
Number of Impressions	486104	486104
Number of Clicks	23923	23923
Number of Uniques	23806	23806
Number of Bounces	8733	8733
Number of Conversions	2026	2026
Total Cost (€)	118097.78	118097.78
CTR	0.049	0.049
CPA	58.291	58.291
CPM	242.948	242.948
Bounce Rate	0.365	0.365

19

17

19

14

Due to the fact there was no anomalies in any of the test data, we could not show a screenshot in the actual program where it changed it, so instead we added the Junit tests which were made to show it has been implemented.

Binomial Probability Distribution Calculator

Enter number of successes: 10

Enter number of trials: 20

Enter probability of success: 0.55

Calculate

0.15934945550

Binomial Probability Distribution Calculator

Enter number of successes: |

Enter number of trials: |

Enter probability of success: |

Calculate

Answer

Statistics Menu

Binomial Probability Distribution Binomial Cumulative Distribution

Poisson Probability Distribution Poisson Cumulative Distribution

15

Statistics Calculators

Open window of statistics 19

242.948 0.365

9

Load Additional Folder		Save Graph	
List 1	Number of impressions 486104	Number of Clicks 23923	
List 2	Number of impressions 486104	Number of Clicks 23923	

19

Load Additional Folder		Save Graph to PNG	
Males	Number of impressions 92345	Number of Clicks 4913	Number of Uniques 4885
List 2	Number of impressions 486104	Number of Clicks 23923	Number of Uniques 23806

19

Load Additional Folder		Save Graph to PNG		Open Histogram of click cost				Show CSV Data		Theme		Statistics Calculators	
Males	Number of impressions 92345	Number of Clicks 4913	Number of Uniques 4885	Number of Bounces 1786	Number of Conversions 416	Total Cost (£) 24327.13	CTR 0.053	CPA 58.479	CPM 263.437	Bounce Rate 0.364			
List 2	Number of impressions 486104	Number of Clicks 23923	Number of Uniques 23806	Number of Bounces 8733	Number of Conversions 2026	Total Cost (£) 118097.78	CTR 0.049	CPA 58.291	CPM 242.948	Bounce Rate 0.365			



19

Load Additional Folder		Save Graph	
List 1	Number of impressions 486104	Number of Clicks 23923	
List 2	Number of impressions 486104	Number of Clicks 23923	

19

Load Additional Folder		Save Graph	
List 1	Number of impressions 486104	Number of Clicks 23923	
List 2	Number of impressions 486104	Number of Clicks 23923	

19

Load Additional Folder		Save Graph to PNG		Open Histogram of click cost				Show CSV Data		Theme		Statistics Calculators	
Males	Number of impressions 92345	Number of Clicks 4913	Number of Uniques 4885	Number of Bounces 1786	Number of Conversions 416	Total Cost (£) 24327.13	CTR 0.053	CPA 58.479	CPM 263.437	Bounce Rate 0.364			
List 2	Number of impressions 486104	Number of Clicks 23923	Number of Uniques 23806	Number of Bounces 8733	Number of Conversions 2026	Total Cost (£) 118097.78	CTR 0.049	CPA 58.291	CPM 242.948	Bounce Rate 0.365			



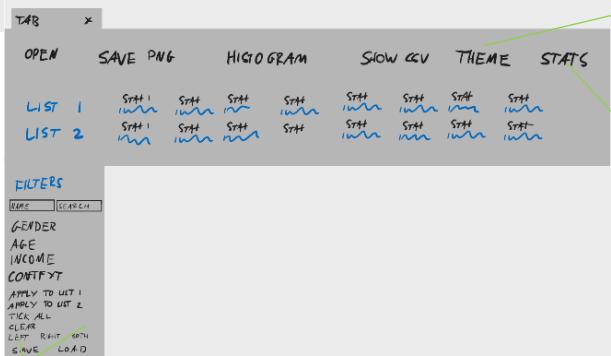
19

Load Additional Folder		Save Graph	
List 1	Number of impressions 486104	Number of Clicks 23923	
List 2	Number of impressions 486104	Number of Clicks 23923	

19

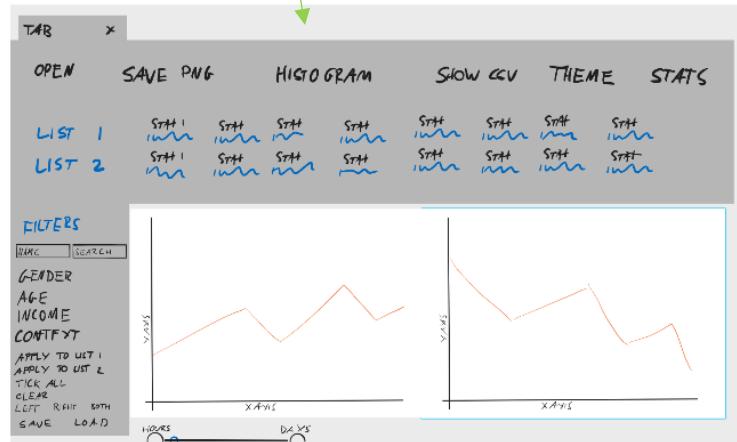
Load Additional Folder		Save Graph	
List 1	Number of impressions 486104	Number of Clicks 23923	
List 2	Number of impressions 486104	Number of Clicks 23923	

AD DASHBOARD

OPEN
RECENT

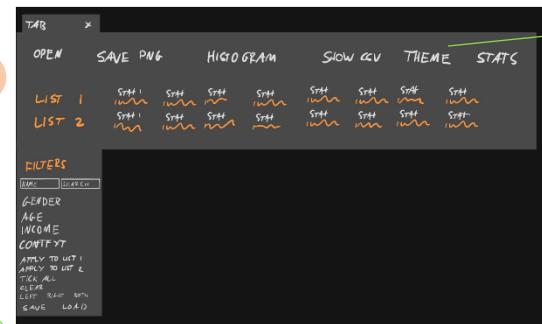
Search through saved filters by entered name

Compare graphs from List 1 and List 2. Click "Left" or "Right" button to expand graph to full screen 9



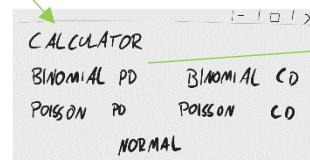
Storyboard

Dark mode



15

Common statistics



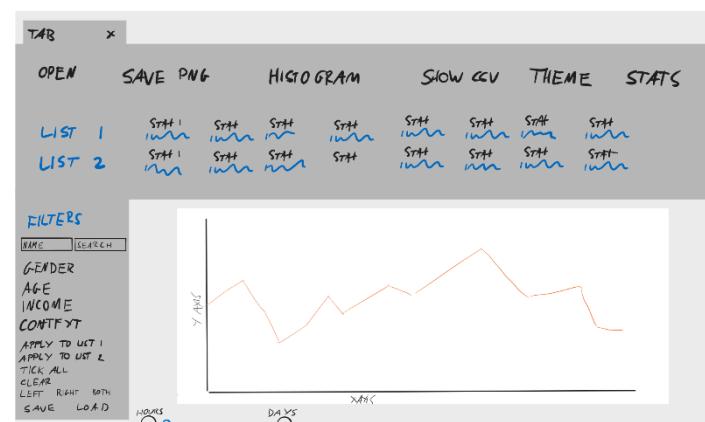
17

High contrast mode

Light mode

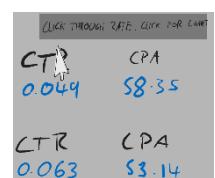


Name filter configurations which are seen next to metric lists



10

Hovering over any button reveals explanation and tooltip 19



Scenarios (Primary)



Anna

Age: 32 Years Old

Location: Chichester,
UK

Stakeholder: Primary

17

Anna has decided she would now like to customize the appearance of the rest of the software to make it less bright, in order for the graphs to be able to be standing out in the software.

- Anna opens up the software, and then chooses the most recently opened dataset
- The main screen is then loaded, with the usual display.
- Anna presses the customise the appearance button, which displays a new screen with different settings.
 - The first new screen she would get is dark mode
 - The second screen that appears is high contrast mode
 - Finally it loops back to the original screen
- Anna chooses to leave it on the dark mode



Dennis

Age: 17 Years Old

Location: London, UK

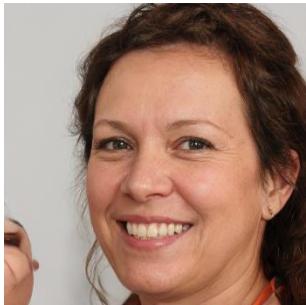
Stakeholder: Primary

15

Dennis has decided as well as the list of metrics provided on the main page, that he would also like to do some further statistical analysis of the data set and decided to use the extra statistics calculations in the software.

- Dennis opens up the software, then chooses the dataset he would like to use
- The main screen opens, with the usual statistics showing
- Dennis then presses the search new statistic button
- Then he can select the statistic type (and define which variables are used depending on the statistic) and the result is shown.

Scenarios (Secondary)



Claire

Age: 52 Years Old

Location: Sussex, UK

Stakeholder: Secondary

14

Claire would like to also remove any anomalies that may have occurred in the spreadsheet, such as errors in correlation between IDs, so that what is being processed is as valid as it can be. So, she would ask her analysts to use the remove anomalies and open function

-Analyst opens up the software, and then chooses the dataset they would like to use.

-Analyst then presses the open button, which opens up what looks like the same display, but it would be without any anomalies as that process would have run in the background



Eddie

Age: 44 Years Old

Location: Stockholm,
Sweden

Stakeholder: Secondary

19 22

Eddie is using this new software but sometimes does find it hard to remember the controls, or how to do certain things within the project. So, he would like to use the tooltips on what he is confused about, so he can easily check the functionality without resorting to the user guide

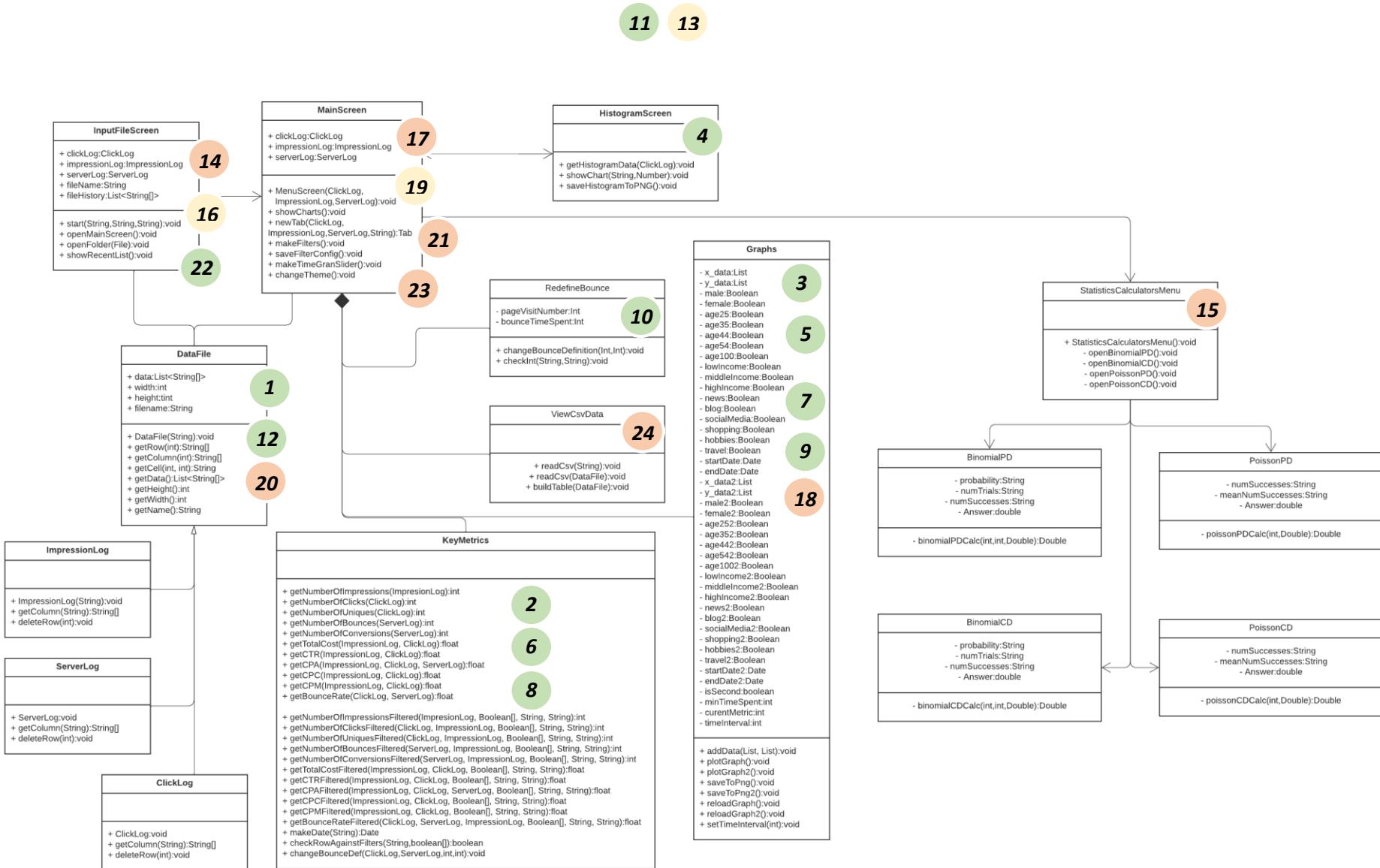
-Eddie opens up the software (using the executable file), and then chooses the most recently opened dataset

-The main screen is then loaded, with the usual display.

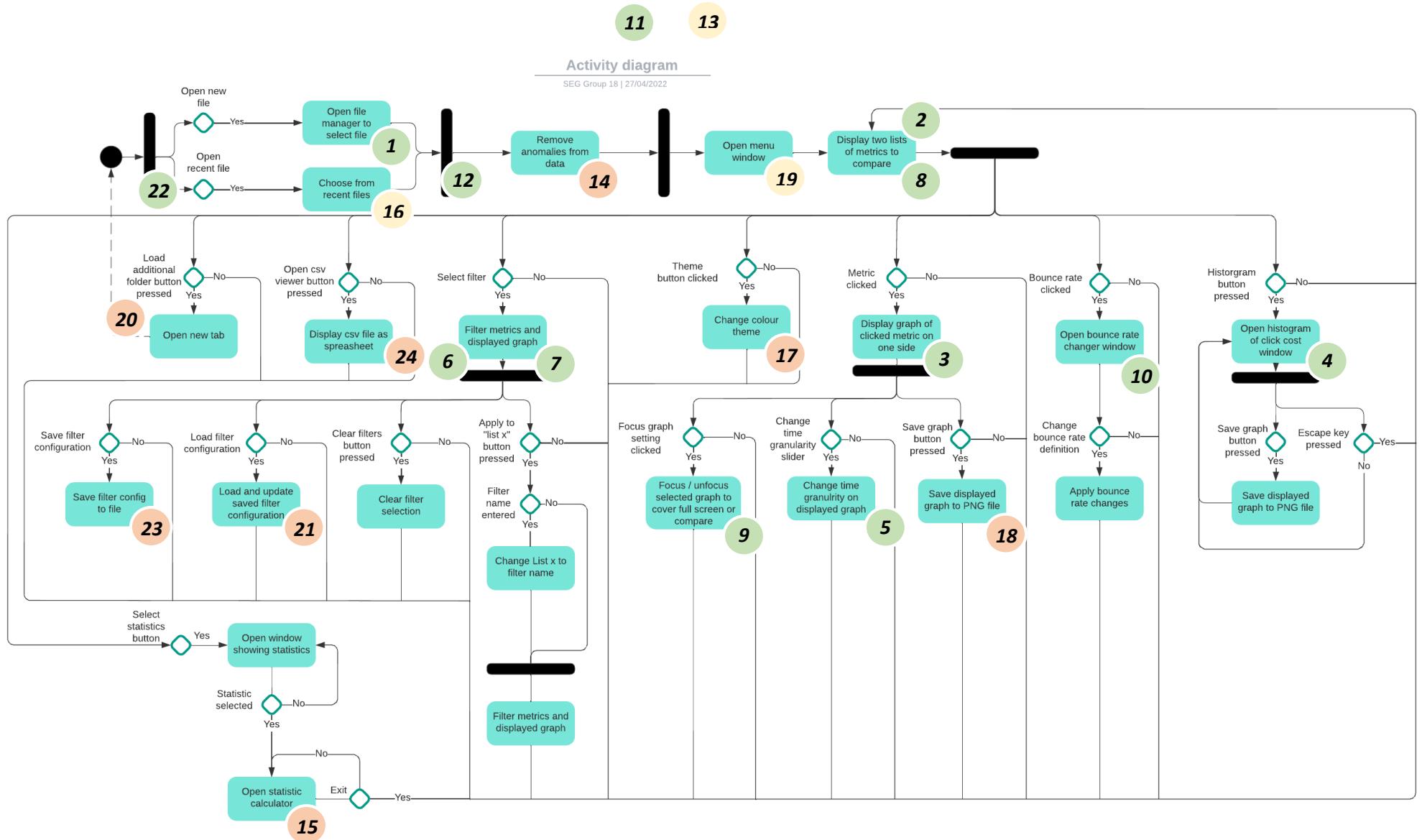
-Eddie then wants to make a graph for his clients, so he uses the tooltips on this screen to see how to use the filters for a graph he wants to use no filters on

-Eddie then makes the graph, saves it to a file (using the helpful tip in the tooltips)

UML Class Diagrams



UML Activity Diagram



Testing

Testing Method	What the test is	Why we would use it	What we have done	Which User Stories
Defect testing	Defect testing is looking for any unintended behaviour/input, what might make it crash, or make it so the user cannot complete their final task as it could not be used.	We use this in order to check our code against outlier code, unexpected mistakes and just anything that could cause our program to break.	While creating the search bar, we planned to test different inputs that could possibly make the program crash, from typos to irrelevant searches, to numbers as inputs, to fully ensure this will not cause the software to break. We also did thorough defect testing on our executable file to ensure it works completely	#21 A search bar #22 An executable file
Validation testing	Validation testing is tests specifically based on checking that the software actually meets the clients needs.	We would use this testing in order to check that our product will give our clients something worth their time, and also to ensure what we create matches what our clients asked us for.	We ran tests against our 4 scenarios, checking that what we have coded has allowed the 4 (primary and secondary) personas to be able to complete their tasks, as well as testing it against our acceptance criteria (all shown below). We have also checked all of our functionality with our supervisor to ensure what we have created is what our client needs/wants	#11 Fast and responsive software #13 Aesthetically pleasing GUI #17 To customise the appearance
Boundary testing	Boundary testing is checking if the program works up to and including any boundaries set on it and fails or does not accept values outside of the boundaries.	The reason we would use this is as it makes sure we have very clear guidelines to what values should be accepted, and which we expect to fail, which then works hand-in-hand with the partition testing.	Within the key metrics that are filtered, we have done some testing to make sure all boundaries have been tested. So, on top of this, we added boundary testing to the well-known common statistics (user story 15) and those unit tests can be seen below.	#15 Well known common statistics
Partition testing	Partition testing is a way of checking the largest range of DIFFERENT possible cases in software, so different possible	This would be used so that instead of having a very large set of tests of every value to be used, we can just pick one from each partition to test it with,	Within the new functions of finding any anomalies in the CSV files data that may break the program and removing them, we tested it against IDs. Our program checked that all IDs used in the click or server log had	#14 To easily find and remove anomalies

	accept/reject states are split into equivalent partitions, and then you just test it partition by partition.	decreasing the number of tests needed while still keeping high accuracy.	corresponding matches in the impression log provided, because if it didn't, then that data would be erroneous or corrupted, and need to be removed from the data. In order to test it worked to its full potential, we tested it with data which had IDs with 0 connections to the impression log, 1 connection to the impression log or 2 or more connections on the same ID to the same ID in the impression log	
Regression testing	Regression testing is making sure that any changes you make to your software doesn't break any previous functions/classes that you have previously made.	For us this will be used due to the fact that a lot of our new functions/classes either rely on or edit previous functions, and we would not only want to check that new software is tested, but the old functionality is still there and working the same.	The main use of regression testing for this increment was how it was applied to the help button, as the help button tooltips are a direct connection to every function we have made. So, we made sure that when the tooltips were added, not only were they an accurate representation of the functionality, but they also did not hinder the usability of the program.	#19 A help button

Junit Testing

```
24
25     @Test
26     void poissonCdCalc() {
27         //large values of x, l
28         assertEquals(BigDecimal.valueOf(0).round(new MathContext( setPrecision: 7, RoundingMode.HALF_UP)),
29                     PoissonCD.poissonCdCalc( x: 5000, BigDecimal.valueOf(5000.0)).round(new MathContext( setPrecision: 7, RoundingMode.HALF_UP)));
30         //small values of x, l
31         assertEquals(BigDecimal.valueOf(0.3678794).round(new MathContext( setPrecision: 7, RoundingMode.HALF_UP)),
32                     PoissonCD.poissonCdCalc( x: 0, BigDecimal.valueOf(1.0)).round(new MathContext( setPrecision: 7, RoundingMode.HALF_UP)));
33         //testing x > l
34         assertEquals(expected: 1, PoissonCD.poissonCdCalc( x: 1000, BigDecimal.valueOf(0.0)).intValue());
35     }
36 }
```

Run: SEG [clean,compile,javafxrun] × PoissonCDTest.poissonCdCalc ×
Tests passed: 1 of 1 test - 39 sec 23 ms
PoissonCDTest (com.example.eig) 39 sec 23 ms partTwo: 1 partThree: 462387260077893773543702433923003985719374864210714632543799910429938512398629820592044208480845
Process finished with exit code 0

```
24
25     @Test
26     void binPdBoundary() {
27         //testing large n
28         assertEquals(BigDecimal.valueOf(3.6437351549E-4).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)),
29                     this.functions.binomialPdCalc( x: 1000, n: 5000, BigDecimal.valueOf(0.185)).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)));
30         //testing small n, x and p
31         assertEquals(BigDecimal.valueOf(1.0).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)),
32                     this.functions.binomialPdCalc( x: 0, n: 1, BigDecimal.valueOf(0.000000001)).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)));
33         //testing invalid variables (x > n)
34         assertEquals(expected: 0, this.functions.binomialPdCalc( x: 1000, n: 4, BigDecimal.valueOf(1)).intValue());
35     }
36 }
```

Run: SEG [clean,compile,javafxrun] × StatisticsTest.binPdBoundary ×
Tests passed: 1 of 1 test - 175 ms
StatisticsTest 175 ms C:\Users\James\.jdks\openjdk-17.0.2\bin\java.exe ... java.lang.ArithmaticException: Division by zero
binPdBoundary 175 ms
Process finished with exit code 0

```
24
25     @Test
26     void poisPdBoundary() {
27         //large values of x, l
28         assertEquals(BigDecimal.valueOf(0.0).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)),
29                     this.functions.poissonPdCalc( x: 5000, BigDecimal.valueOf(5000.0)).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)));
30         //small values of x, l
31         assertEquals(BigDecimal.valueOf(1.0).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)),
32                     this.functions.binomialPdCalc( x: 0, n: 1, BigDecimal.valueOf(0.000000001)).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)));
33         //testing x > l
34         assertEquals(expected: 0, this.functions.poissonPdCalc( x: 1000, BigDecimal.valueOf(0.0)).intValue());
35     }
36 }
```

Run: SEG [clean,compile,javafxrun] × StatisticsTest.poisPdBoundary ×
Tests passed: 1 of 1 test - 205 ms
StatisticsTest 205 ms partTwo: 1 partThree: 402387260077893773543702433923003985719374864210714632543799910429938512398629820592044208486948480845
Process finished with exit code 0

```
24
25     @Test
26     void binomialCdCalc() {
27         //testing large n
28         assertEquals(BigDecimal.valueOf(0.99679135889).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)),
29                     BinomialCD.binomialCdCalc( x: 1000, n: 5000, BigDecimal.valueOf(0.185)).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)));
30         //testing small n x and p
31         assertEquals(BigDecimal.valueOf(0.999999999).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)),
32                     BinomialCD.binomialCdCalc( x: 0, n: 1, BigDecimal.valueOf(0.000000001)).round(new MathContext( setPrecision: 11, RoundingMode.HALF_UP)));
33         //testing invalid variables (x > n)
34         assertEquals(expected: 0, this.functions.binomialPdCalc( x: 1000, n: 4, BigDecimal.valueOf(1)).intValue());
35     }
36 }
```

Run: SEG [clean,compile,javafxrun] × BinomialCDTest.binomialCdCalc ×
Tests passed: 1 of 1 test - 21 sec 485 ms
BinomialCDTest (com.example.eig) 21 sec 485 ms binomialCdCalc 21 sec 485 ms C:\Users\James\.jdks\openjdk-17.0.2\bin\java.exe ... java.lang.ArithmaticException: Division by zero
binomialCdCalc 21 sec 485 ms
Process finished with exit code 0

The above is some boundary testing for the Statistics Calculators. There is a maximum value for the input variables, set at 5000, as the formulae for the 'CD' calculations require intensive recursion and the Binomial Distribution calculations use factorial calculation. These reasons make computation of the results difficult for most laptop processors, thus the limit.

For the tests, the boundaries on all inputs are $0 \leq$ and ≤ 5000 ; any other values cannot be entered in the actual program as regex limiters on the input boxes have been put in place.

	A	B	C
1	Date	ID	Click Cost
2	01/01/2015 12:01	1	11.7
3	02/01/2015 12:01	2	11.7
4	03/01/2015 12:02	3	0
5	04/01/2015 12:02	3	0
6	04/01/2015 12:04	4	9.2
7	05/01/2015 12:04	5	9.8
8	06/01/2015 12:04	6	0
9	06/01/2015 12:04	7	0
10	07/01/2015 12:04	8	0
11	08/01/2015 12:04	8	0
12	08/01/2015 12:05	9	14.6
13	09/01/2015 12:06	10	0
14	09/01/2015 12:06	11	8.2
15	10/01/2015 12:07	12	0
16	10/01/2015 12:09	13	9.6
17	11/01/2015 12:11	13	0
18	12/01/2015 12:11	14	13.4
19	12/01/2015 12:12	14	11.3
20	13/01/2015 12:14	15	0

Click Log Test Data

	A	B	C	D	E	F	G
1	Date	ID	Gender	Age	Income	Context	Impression Cost
2	01/01/2015 12:00	1	Male	25-34	High	Blog	0.002
3	01/01/2015 12:00	2	Female	35-44	Medium	News	0.003
4	02/01/2015 12:00	3	Female	>54	Medium	Shopping	0.002
5	03/01/2015 12:00	5	Male	<25	Low	Social Media	0.002
6	04/01/2015 12:00	6	Female	<25	Medium	Shopping	0
7	05/01/2015 12:00	7	Female	>54	Medium	News	0.002
8	08/01/2015 12:00	10	Female	45-54	Low	Social Media	0.002
9	09/01/2015 12:00	11	Female	>54	Medium	Shopping	0.003
10	10/01/2015 12:00	12	Female	45-54	Medium	Blog	0
11	11/01/2015 12:00	13	Female	25-34	Low	Shopping	0.003
12	13/01/2015 12:00	15	Female	<25	Low	Shopping	0.003

Impression Log Test Data

	A	B	C	D	E
1	Entry Date	ID	Exit Date	Pages Viewed	Conversion
2	01/01/2015 12:01	1	01/01/2015 12:05	7	No
3	01/01/2015 12:01	2	01/01/2015 12:02	1	No
4	02/01/2015 12:02	3	02/01/2015 12:05	10	No
5	02/01/2015 12:02	4	02/01/2015 12:06	3	No
6	03/01/2015 12:04	5	03/01/2015 12:04	1	No
7	04/01/2015 12:04	6	04/01/2015 12:05	4	No
8	05/01/2015 12:04	7	05/01/2015 12:05	4	Yes
9	06/01/2015 12:04	8	06/01/2015 12:09	10	No
10	07/01/2015 12:04	9	07/01/2015 12:07	7	No
11	08/01/2015 12:04	10	08/01/2015 12:06	15	Yes
12	09/01/2015 12:05	11	n/a	10	No
13	10/01/2015 12:06	12	10/01/2015 12:06	1	No
14	11/01/2015 12:07	13	11/01/2015 12:08	1	No
15	12/01/2015 12:07	14	12/01/2015 12:07	1	No
16	13/01/2015 12:09	15	13/01/2015 12:10	1	No

Server Log Test Data

For User Story 14s Testing

```
import ...

class ClickLogCleanDataTest {
    ImpressionLog impressionLog;
    ClickLog clickLog;

    @BeforeEach
    void setUp() throws IOException, CsvException {
        this.impressionLog = new ImpressionLog( filename: "C:\\Users\\jmboy\\IdeaProjects\\UserStory14s\\src\\main\\resources\\impressionLog.csv" );
        this.clickLog = new ClickLog( filename: "C:\\Users\\jmboy\\IdeaProjects\\UserStory14s\\src\\main\\resources\\clickLog.csv" );
    }

    @Test
    void clean_data() {
        clickLog.clean_data(impressionLog);
        assertEquals( expected: 13, clickLog.getLength());
    }
}
```

ClickLogCleanDataTest

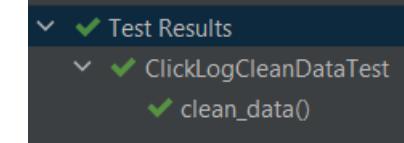
```
import ...

class ServerLogCleanDataTest {
    ImpressionLog impressionLog;
    ServerLog serverLog;

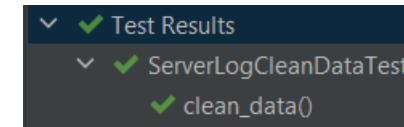
    @BeforeEach
    void setUp() throws IOException, CsvException {
        this.impressionLog = new ImpressionLog( filename: "C:\\Users\\jmboy\\IdeaProjects\\UserStory14s\\src\\main\\resources\\impressionLog.csv" );
        this.serverLog = new ServerLog( filename: "C:\\Users\\jmboy\\IdeaProjects\\UserStory14s\\src\\main\\resources\\serverLog.csv" );
    }

    @Test
    void clean_data() {
        serverLog.clean_data(impressionLog);
        assertEquals( expected: 11, serverLog.getLength());
    }
}
```

ServerLogCleanDataTest



ClickLogCleanDataTest Passed



ServerLogCleanDataTest Passed

Testing Against Acceptance Criteria

User Story 1 Acceptance Criteria

Users can choose a folder with the data files in
CSV files can be loaded into the application
The CSV files can be read into the application data

PASSED

User Story 2 Acceptance Criteria

Users can choose a folder with the data files in
CSV files can be loaded into the application
A list of metrics can be calculated from that data
A list of metrics can be displayed from that data

PASSED

User Story 3 Acceptance Criteria

Users can choose a folder with the data files in
CSV files can be loaded into the application
A graph (which compares a statistic to time) can be created
A graph can be viewed

PASSED

The screenshot shows the 'Ad Auction Dashboard' window. On the left, there's a 'Select folder' dialog box with a green arrow pointing to the 'Recent' section. The main dashboard displays various metrics for 'List 1' and 'List 2'. A large red 'PASSED' stamp is overlaid across the middle of the screen.

List	Number of Impressions	Number of Clicks	Number of Uniques	Number of Bounces	Number of Conversions	Total Cost (£)	CTR	CPA	CPM	Bounce Rate
List 1	486104	23923	23806	8733	2026	118097.78	0.049	58.291	242.948	0.365
List 2	486104	23923	23806	8733	2026	118097.78	0.049	58.291	242.948	0.365

A line chart titled 'Graph of Number of Impressions over time' shows the number of impressions per day from January 1, 2015, to January 14, 2015. The Y-axis ranges from 0 to 45,000. The data shows a general upward trend with some fluctuations, ending at approximately 15,000 on January 14.

The screenshot shows the 'Ad Auction Dashboard' window again. A green arrow points to the 'Number of impressions' button in the 'List 1' row. A tooltip appears over the button with the text 'Number of impressions, Click to open the chart'.

PASSED

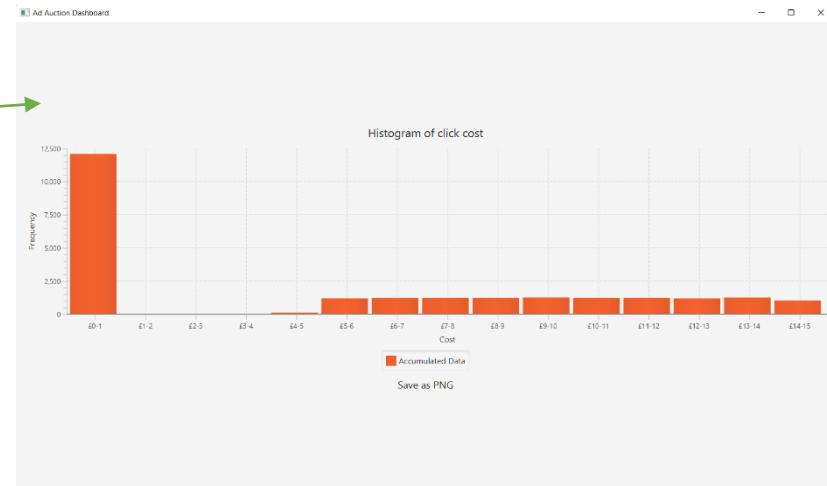
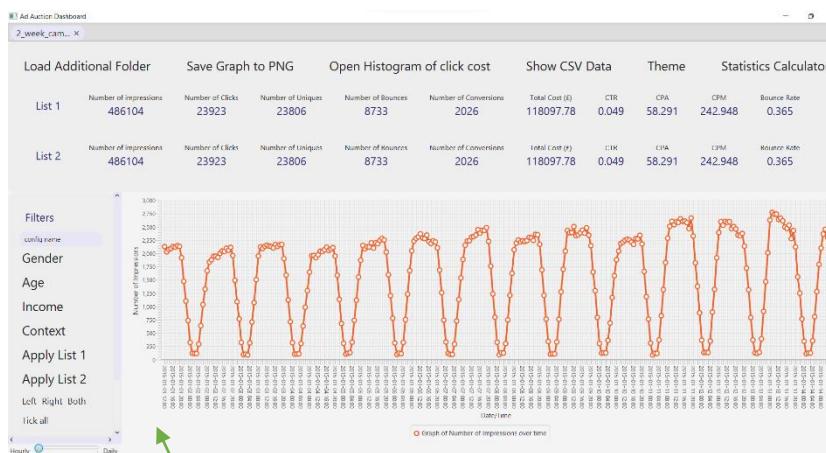
User Story 4 Acceptance Criteria

Users can choose a folder with the data files in

CSV files can be loaded into the application

A histogram of click costs can be created

A histogram of click costs can be viewed



PASSED

User Story 5 Acceptance Criteria

Users can choose a folder with the data files in

CSV files can be loaded into the application

A graph (which compares a statistic to time) can be created

A graph can be viewed

The time axis can be changed

The graph that is viewed can be updated due to the time axis



PASSED

User Story 6 Acceptance Criteria

Users can choose a folder with the data files in

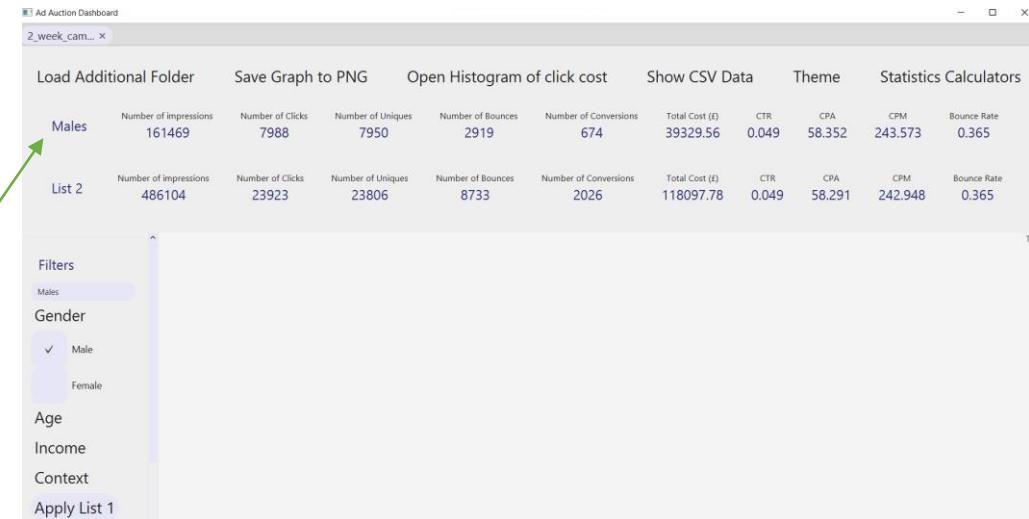
CSV files can be loaded into the application
A list of metrics can be calculated from that data

A list of metrics can be displayed from that data

The user can filter which data is used for these metrics

A list of new metrics can be re-calculated from the data and those filters

The list of metrics displayed can be updated to the re-calculated metrics



PASSED

User Story 7 Acceptance Criteria

Users can choose a folder with the data files in

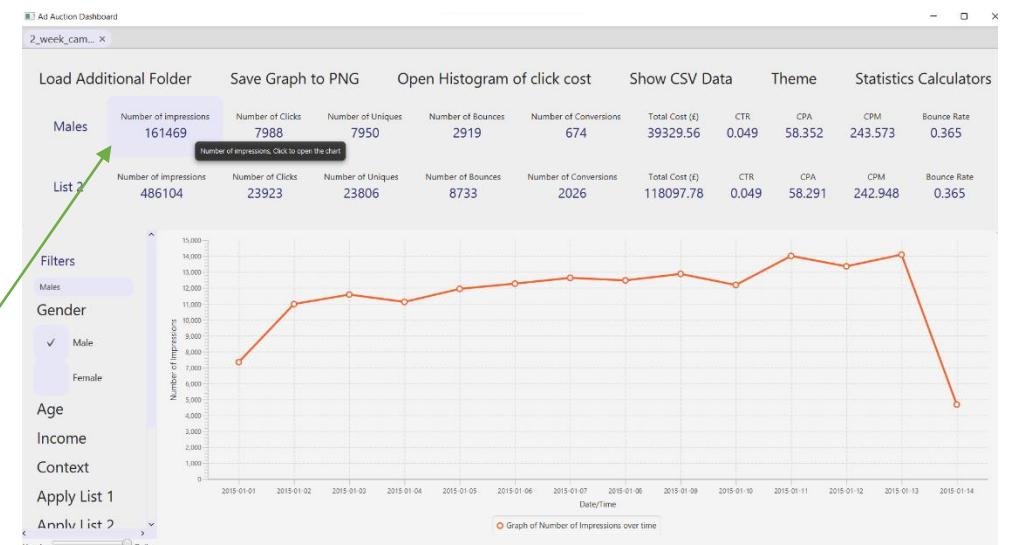
CSV files can be loaded into the application
A graph (which compares a statistic to time) can be created

A graph can be viewed

The user can filter which data is used for this graph

The graph can be re-created from the data and those filters

The graph displayed can be updated to the re-created graph



User Story 8 Acceptance Criteria

Users can choose a folder with the data files in
 CSV files can be loaded into the application
Two lists of metrics can be calculated from that data
Two lists of metrics can be displayed from that data
The user can filter which data is used for either of the lists of metrics
 A list of new metrics can be re-calculated from the data and those filters
 The list of metrics displayed can be updated to the re-calculated metrics

PASSED

Load Additional Folder

Males Number of impressions 161469

Females Number of impressions 486104

Filters

Gender

Male

Female

Age

Income

Context

Apply List 1

Apply List 2

Load Additional Folder

Males Number of impressions 161469

List 2 Number of impressions 486104

Filters

Gender

Male

Female

Age

Income

Context

Apply List 1

Apply List 2

Load Additional Folder

List 1 Number of impressions 486104

List 2 Number of impressions 486104

Filters

Gender

Male

Female

Age

Income

Context

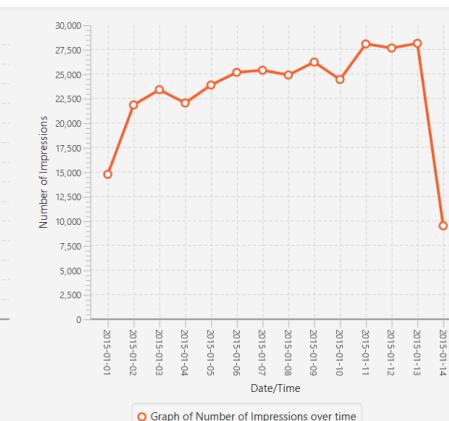
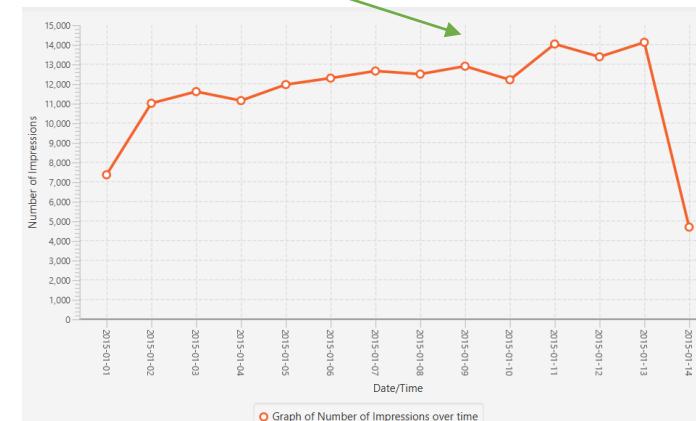
Apply List 1

Apply List 2

User Story 9 Acceptance Criteria

Users can choose a folder with the data files in
 CSV files can be loaded into the application
Two graphs (which compare a statistic to time) can be created
Two graphs can be viewed
The user can filter which data is used for either of the graphs
 The graph can be re-created from the data and those filters
 The graph displayed can be updated to the re-created graph

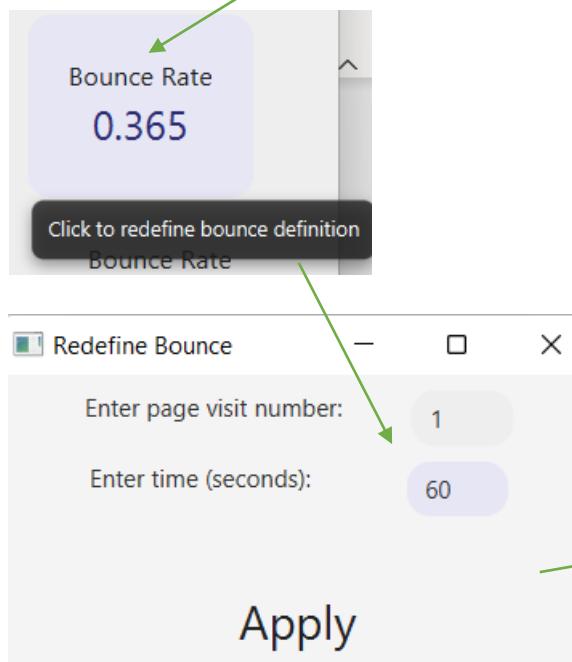
PASSED



Load Additional Folder		Save Graph to PNG		Open Histogram of click cost		Show CSV Data		Theme		Statistics Calculators	
Males	Number of impressions 161469	Number of Clicks 7988	Number of Uniques 7950	Number of Bounces 2919	Number of Conversions 674	Total Cost (£) 39329.56	CTR 0.049	CPA 58.352	CPM 243.573	Bounce Rate 0.365	
Females	Number of impressions 324635	Number of Clicks 15935	Number of Uniques 15856	Number of Bounces 5814	Number of Conversions 1352	Total Cost (£) 78768.17	CTR 0.049	CPA 58.26	CPM 242.636	Bounce Rate 0.365	

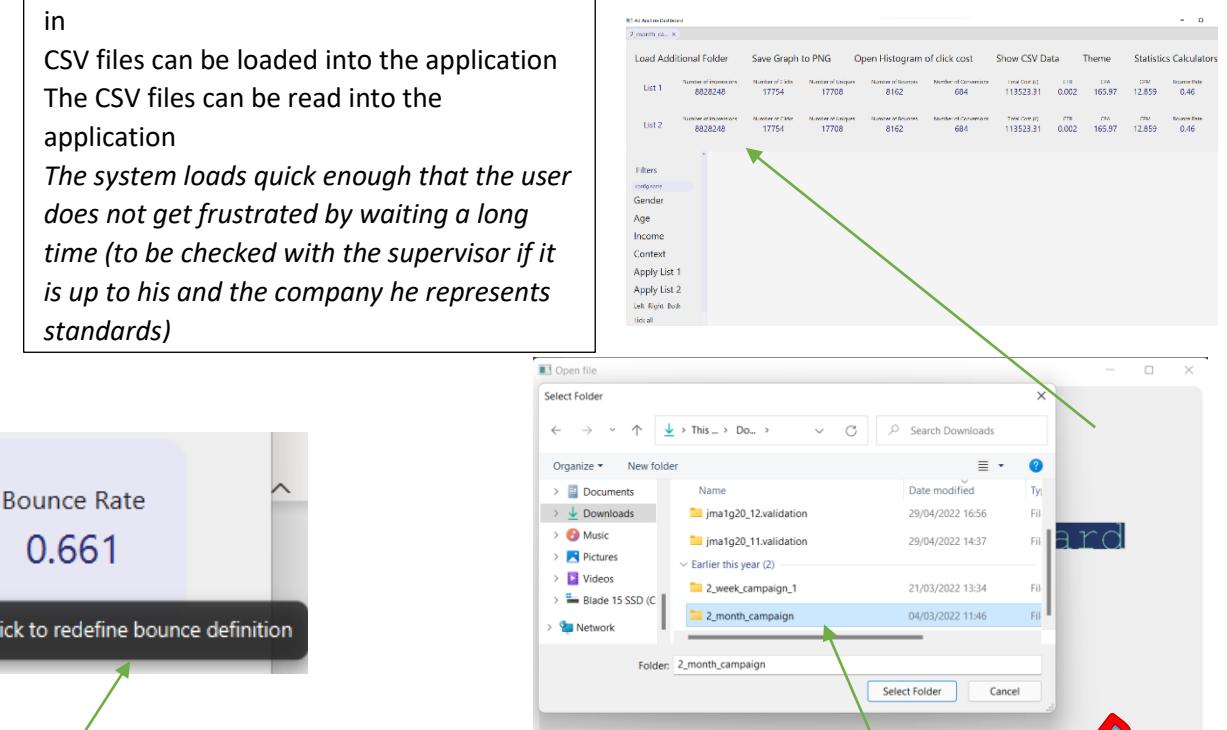
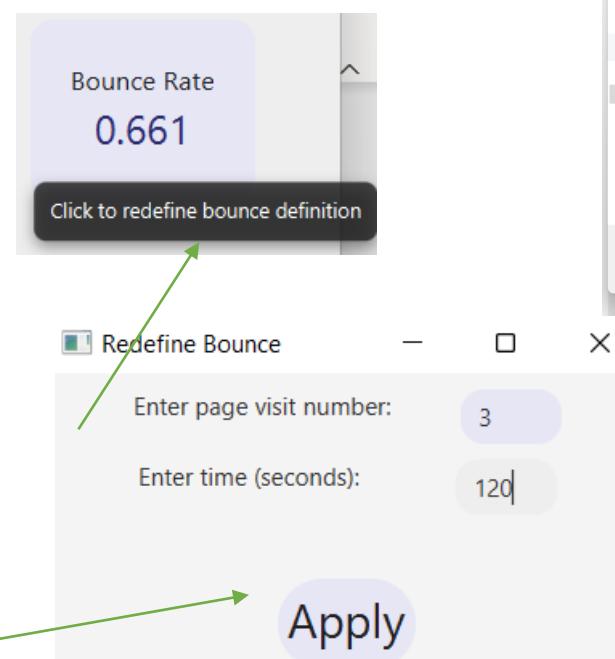
User Story 10 Acceptance Criteria

Users can choose a folder with the data files in
CSV files can be loaded into the application
A list of metrics can be calculated from that data
A list of metrics can be displayed from that data
A user can update the calculation of bounce by the number of pages visited
A user can update the calculation of bounce by the time spent on a page
The list of metrics displayed can be updated with the re-calculated bounce rate



User Story 11 Acceptance Criteria

Users can choose a folder with the data files in
CSV files can be loaded into the application
The CSV files can be read into the application
The system loads quick enough that the user does not get frustrated by waiting a long time (to be checked with the supervisor if it is up to his and the company he represents standards)



User Story 12 Acceptance Criteria

Users can choose a folder with very large data files in
CSV files can be loaded into the application in a promptly matter
The system can complete all other functionality even with a much larger csv file set

User Story 13 Acceptance Criteria

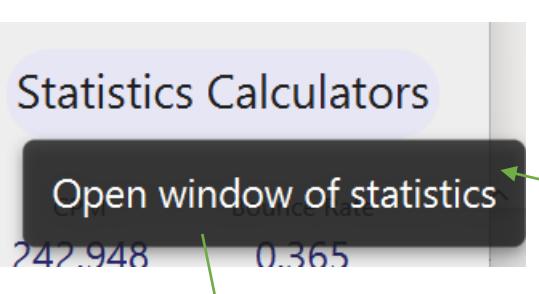
The system is appealing to look at and use, and functionality is easy to deduce (to be checked with the supervisor if it is up to his and the company he represents standards)

User Story 14 Acceptance Criteria

Users can choose a folder with the data files in
 CSV files can be loaded into the application
The system can examine all of the data
The system can remove any lines that could not be accurate
The updated data can be used for all the calculations

Due to the fact there was no anomalies in any of the test data, we could not show a screenshot in the actual program where it changed it, so instead we added the Junit tests which were made to show it has been implemented.

PASSED

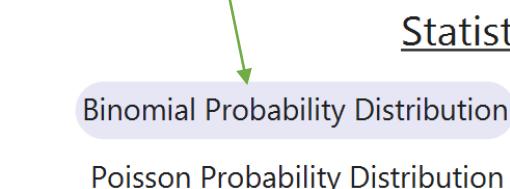


User Story 15 Acceptance Criteria

Users can choose a folder with the data files in
 CSV files can be loaded into the application
A list of extra metrics can be calculated from that data (these metrics being useful commonly used metrics for statistical analysis)
The extra metrics can be displayed from that data

PASSED

PASSED



Statistics Menu

Binomial Cumulative Distribution
 Poisson Cumulative Distribution

Binomial Probability Distribution Calculator

Enter number of successes:

Enter number of trials:

Enter probability of success:

Calculate

Answer

Binomial Probability Distribution Calculator

Enter number of successes:

Enter number of trials:

Enter probability of success:

Calculate

0.15934945550

PASSED

User Story 16 Acceptance Criteria

Users can choose a folder with the data files in

CSV files can be loaded into the application

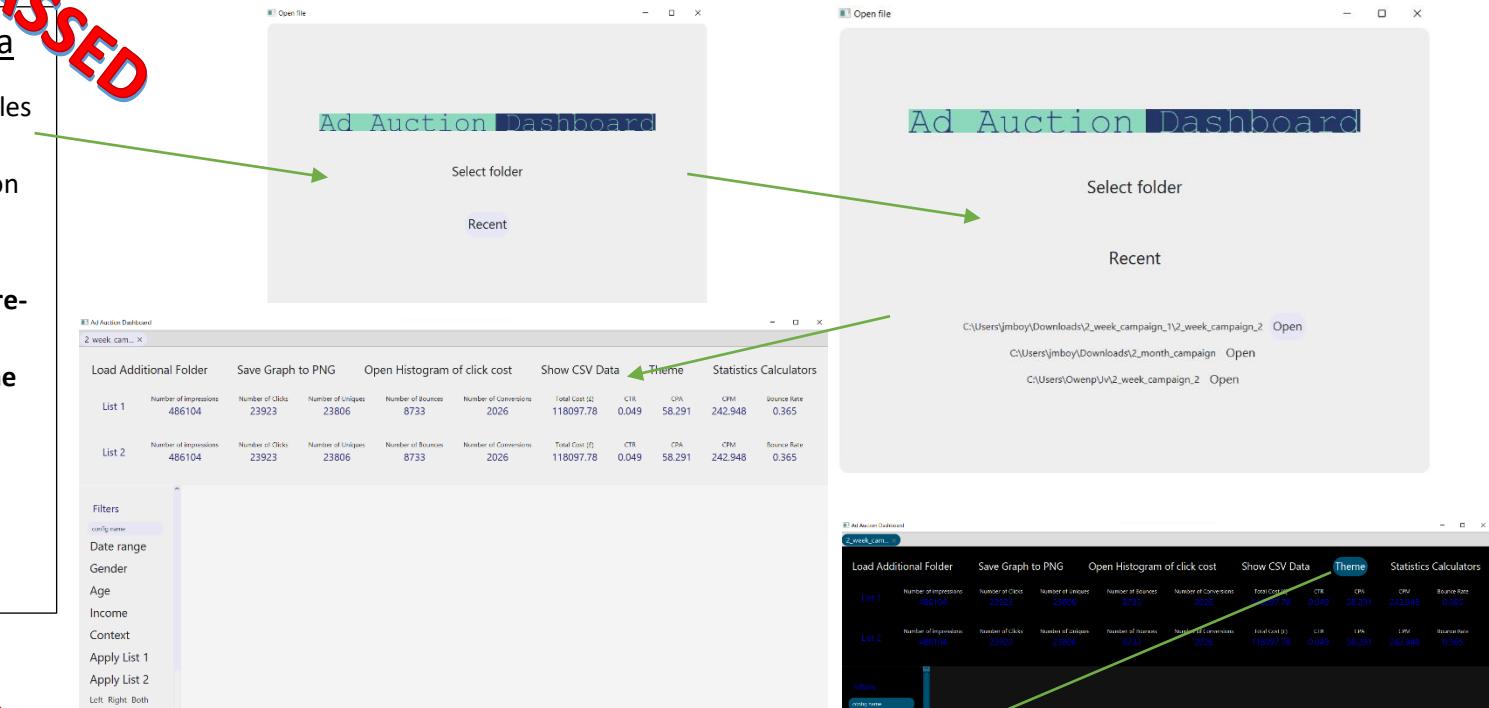
The CSV files can be read into the application

The application can then be closed, and re-opened

Users can then choose a shortcut with the previous folder location

Those CSV files can be loaded into the application

The CSV files can be read into the application



PASSED

User Story 17 Acceptance Criteria

Users can choose a folder with the data files in

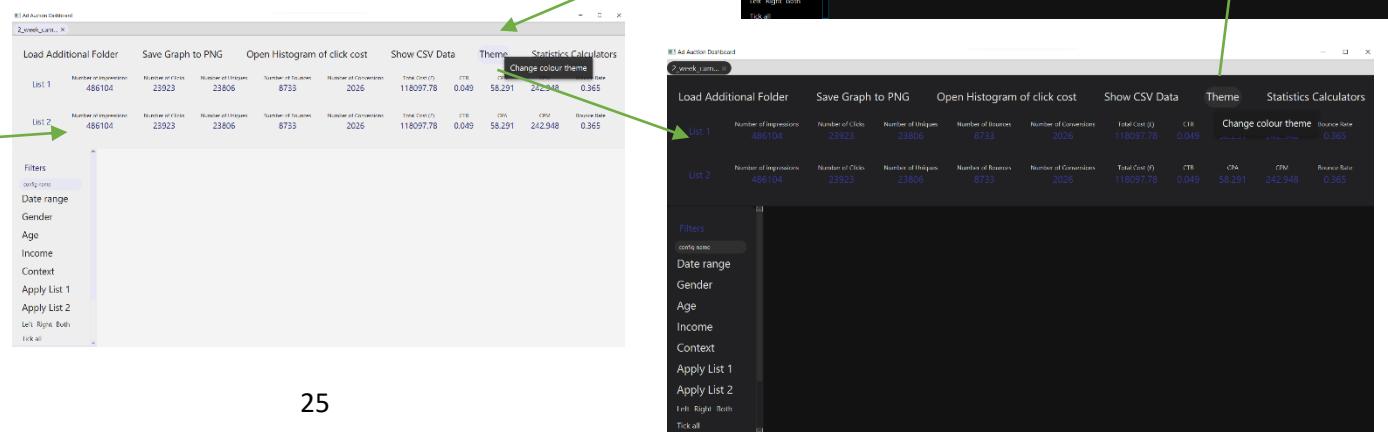
CSV files can be loaded into the application

Users can choose to open a menu to change the appearance

Users are able to change the style mode to a dark theme

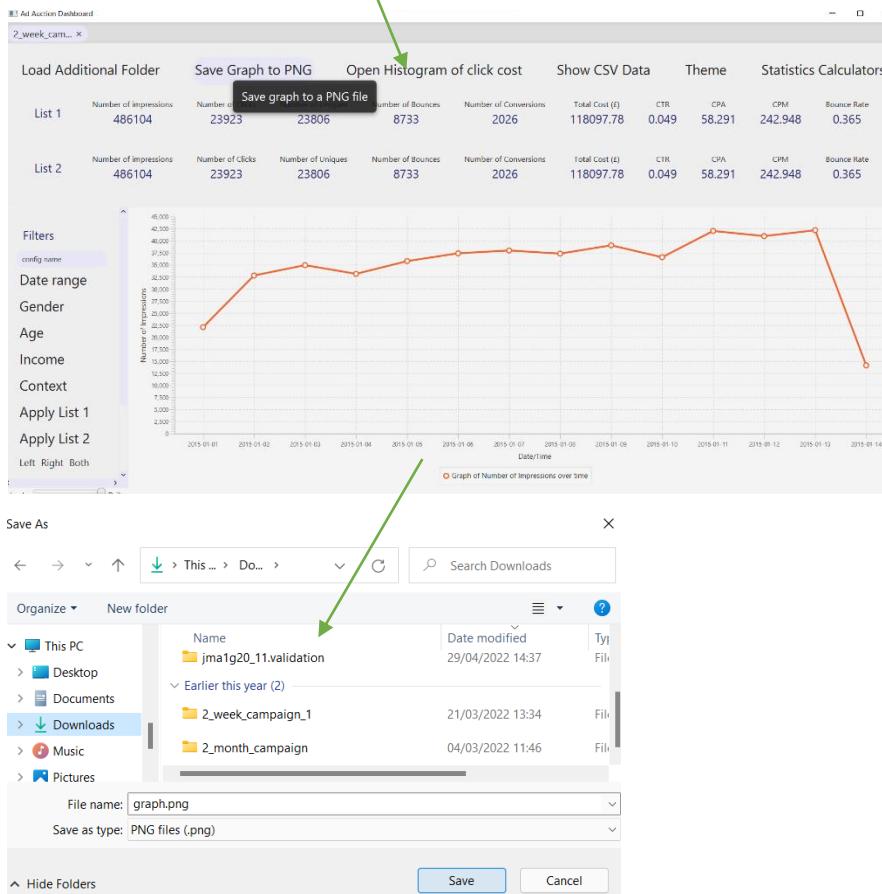
Users are able to change the style mode to a high contrast theme

These preferences will be saved and used throughout the program



User Story 18 Acceptance Criteria

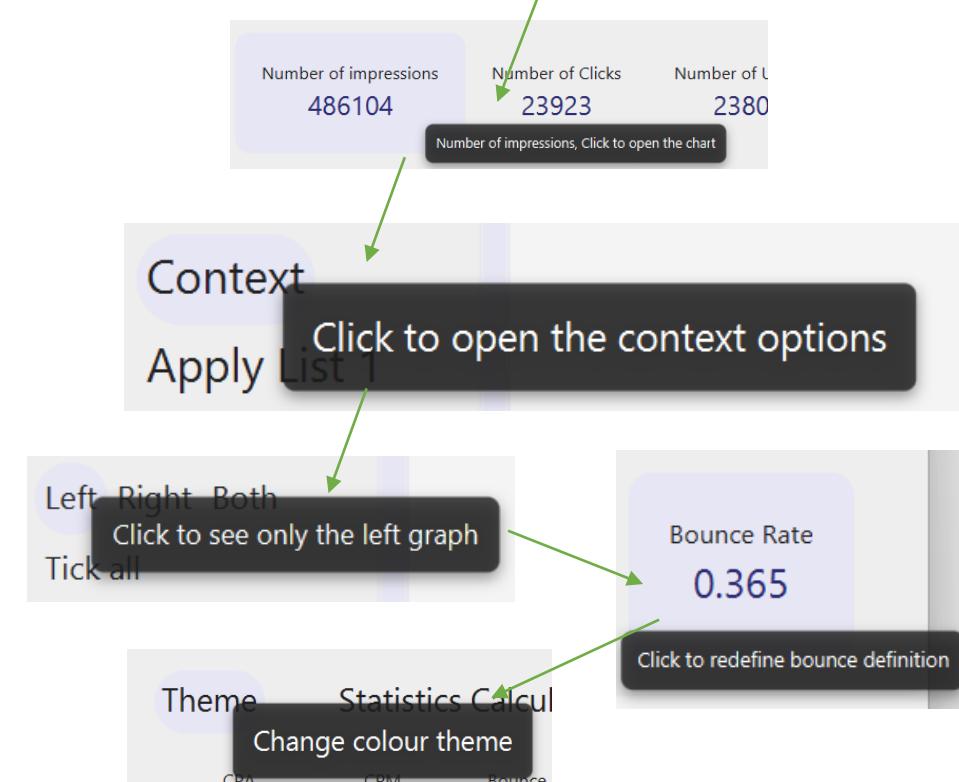
Users can choose a folder with the data files in
CSV files can be loaded into the application
A graph (which compares a statistic to time) can be created
A graph can be viewed
A graph can be saved onto the computer hard drive



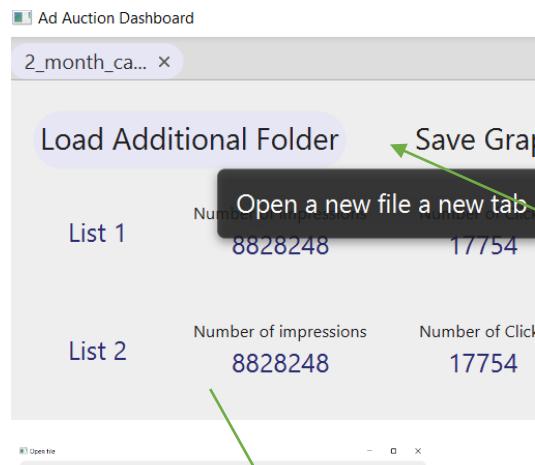
PASSED

User Story 19 Acceptance Criteria

Users can choose a folder with the data files in
CSV files can be loaded into the application
The CSV files can be read into the application
Users can get tooltips on actions on buttons on the software



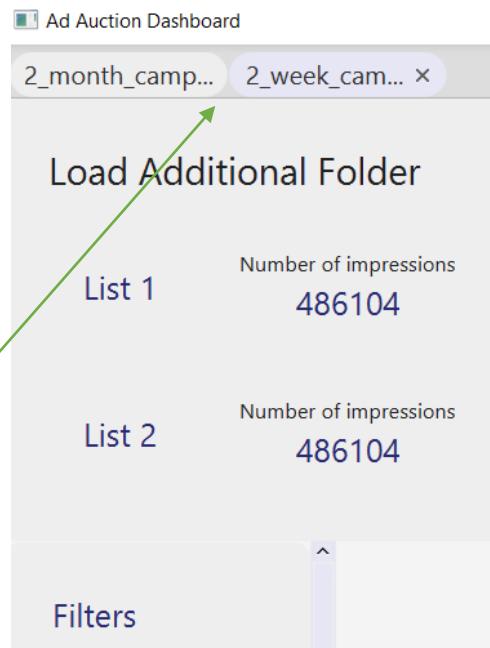
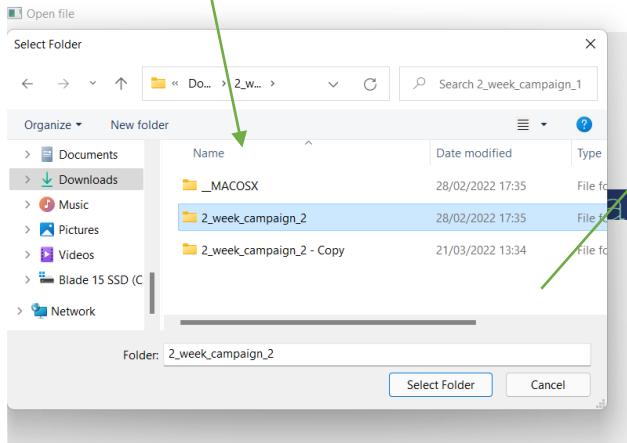
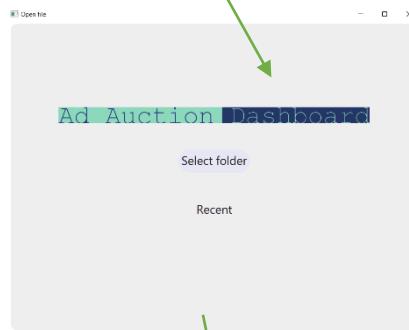
PASSED



User Story 20 Acceptance Criteria

Users can choose a folder with the data files in
 CSV files can be loaded into the application
A list of extra metrics can be calculated from that data (these metrics being useful commonly used metrics for statistical analysis)
The extra metrics can be displayed from that data

PASSED



User Story 21 Acceptance Criteria

Users can choose a folder with the data files in
 CSV files can be loaded into the application
 The user can filter which data is used for the metrics/graph
 The user can save these filters to be able to quickly reuse them later on
The user can search through saved filters by name to find one efficiently

Obsolete

Due to the fact we implemented the loading a configuration to open a file explorer window, there was no longer a necessity of a search bar. This user story was only ranked as a "could do", and we checked this with our supervisor, and he completely agreed with this conclusion.

ad_auction_dashboard.jar

User Story 22 Acceptance Criteria

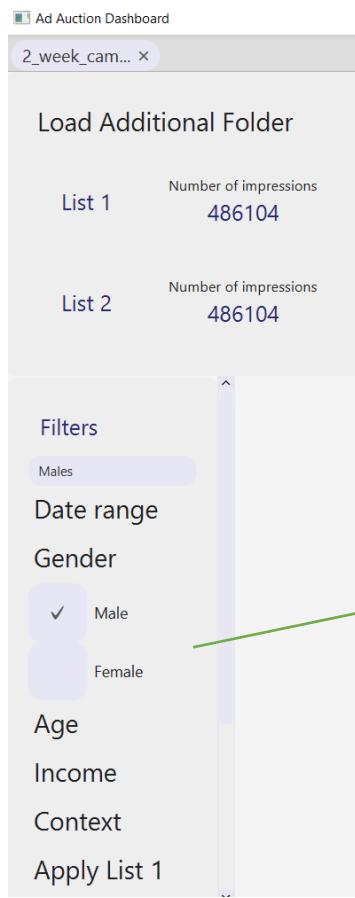
Users can open this application by just using an executable file

PASSED

User Story 23 Acceptance Criteria

Users can choose a folder with the data files in
CSV files can be loaded into the application
The user can filter which data is used for the metrics/graph
The user can save these filters to be able to quickly reuse them later on

PASSED



Ad Auction Dashboard

2_week_campaign_1

Load Additional Folder

List 1 Number of impressions 486104

List 2 Number of impressions 486104

Filters

Males

Date range

Gender

Male

Female

Age

Income

Context

Apply List 1

Apply List 2

Left Right Both

Tick all

Clear

Save Configuration

Load Configuration

Save configuration



Ad Auction Dashboard

2_week_campaign_1

Load Additional Folder

List 1 Number of impressions 486104

List 2 Number of impressions 486104

Filters

Female

Age

Income

Context

Apply List 1

Apply List 2

Left Right Both

Tick all

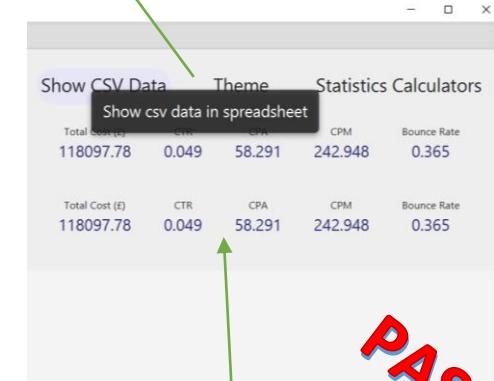
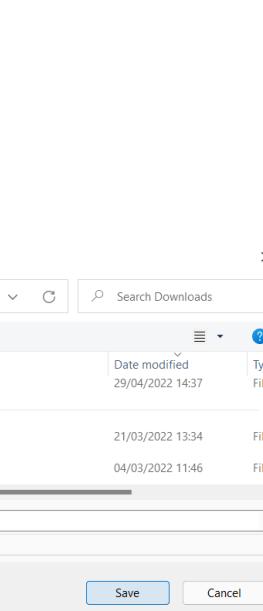
Clear

Save Configuration

Load Configuration

Save configuration

Date	ID	Click Cost
2015-01-01 12:01:21	8895519749317550080	11.794442
2015-01-01 12:01:33	6487139546184700800	11.718663
2015-01-01 12:02:25	1277480883056123904	0.000000
2015-01-01 12:02:57	3777890599251549184	0.000000
2015-01-01 12:04:02	6202006224593186816	9.340521
2015-01-01 12:04:15	7375754838836782080	9.827456
2015-01-01 12:04:12	1162345557141671936	0.000000
2015-01-01 12:04:11	8370837523317244928	0.000000
2015-01-01 12:04:28	5217170615204436992	0.000000
2015-01-01 12:04:35	9205559084602150912	0.000000
2015-01-01 12:05:51	4033231570324092928	14.570611
2015-01-01 12:06:48	5024355283837802496	0.000000
2015-01-01 12:06:56	4176769234609457152	8.167711
2015-01-01 12:07:40	6194744146071669760	0.000000
2015-01-01 12:09:46	5901145211964154880	9.557479
2015-01-01 12:11:11	3334649978696392704	0.000000
2015-01-01 12:11:21	1204592395485832192	13.430781
2015-01-01 12:12:47	7928515372481310720	11.311353
2015-01-01 12:14:57	2372317742582114304	0.000000
2015-01-01 12:15:53	5874214096119492608	14.543623
2015-01-01 12:16:10	8918212838267728896	14.941834



Show CSV Data

Show csv data in spreadsheet

Theme

Statistics Calculators

Total Cost (€)	CTR	CPA	CPM	Bounce Rate
118097.78	0.049	58.291	242.948	0.365
Total Cost (€)	CTR	CPA	CPM	Bounce Rate
118097.78	0.049	58.291	242.948	0.365

User Story 24 Acceptance Criteria

Users can choose a folder with the data files in
CSV files can be loaded into the application
The CSV files can all be viewed on the application in an excel format

PASSED

Testing Program Against Scenarios (Anna From Sprint 2)



Anna

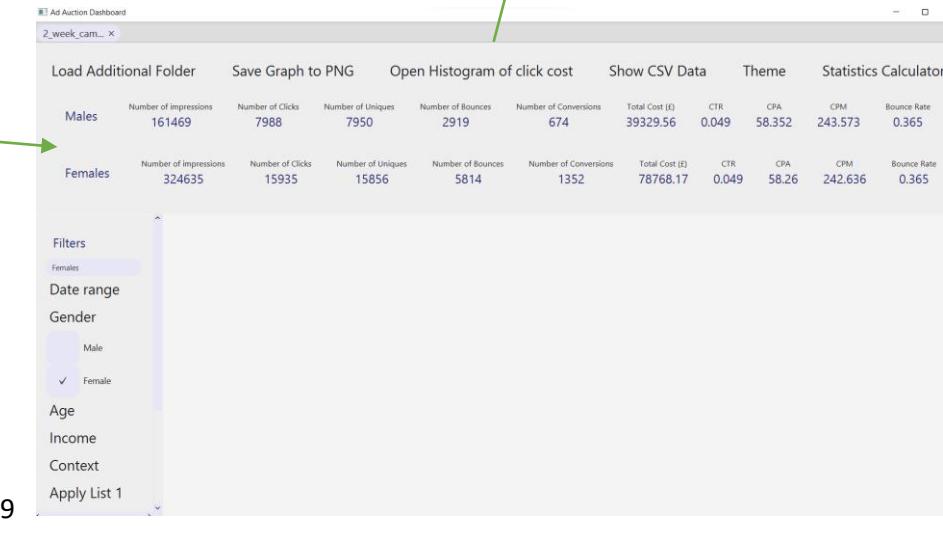
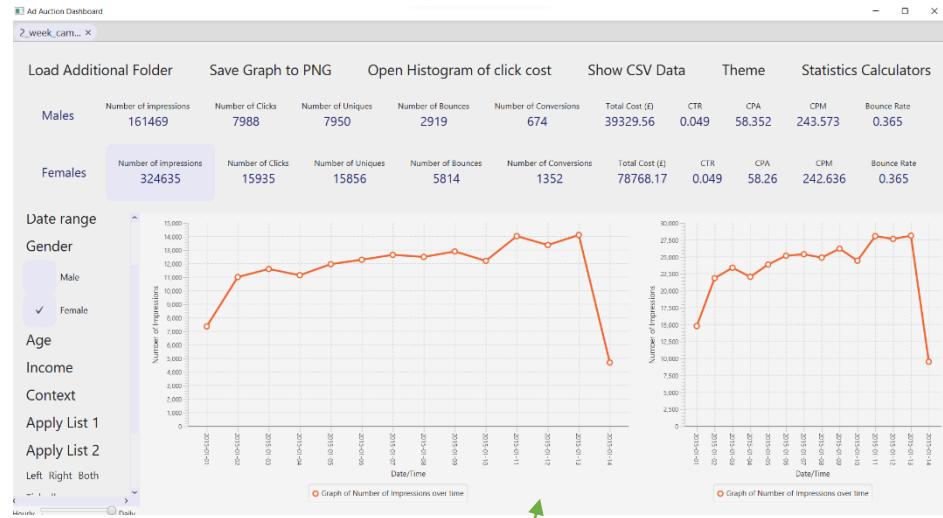
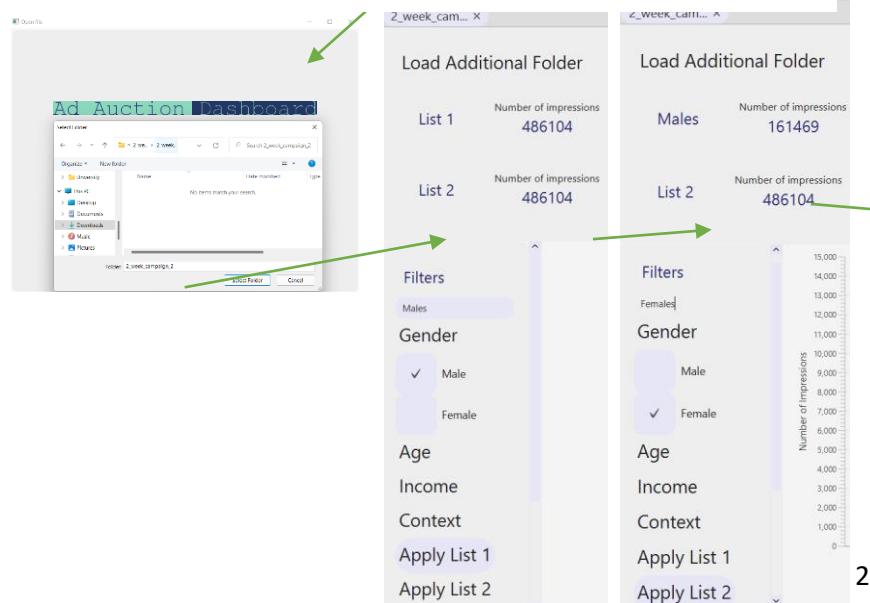
Age: 32 Years Old

Location: Chichester,
UK

Stakeholder: Primary

9

- Anna opens the application and she selects a folder containing ad data
 - She then uses the filter checkboxes to create 2 different sets of data to make graphs of
 - She then selects "total number of clicks", and the 2 different graphs is then shown on the screen next to each other to be compared



Testing Program Against Scenarios (Anna)



Anna

Age: 32 Years Old

Location: Chichester,
UK

Stakeholder: Primary

17

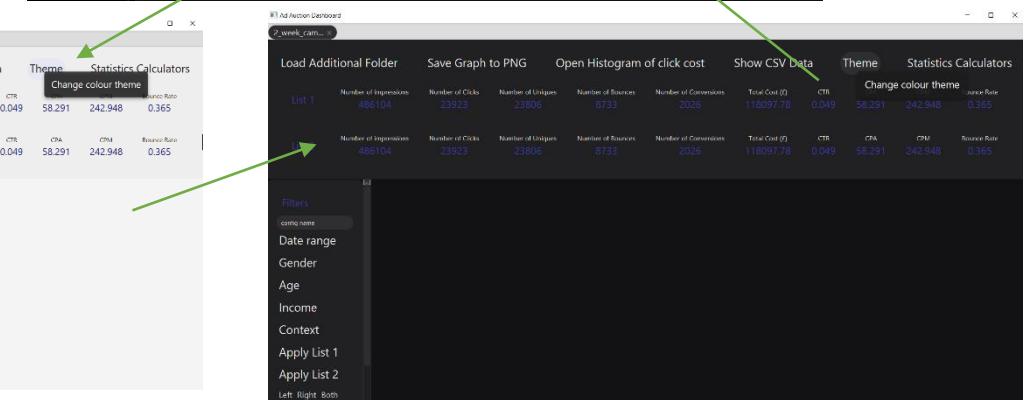
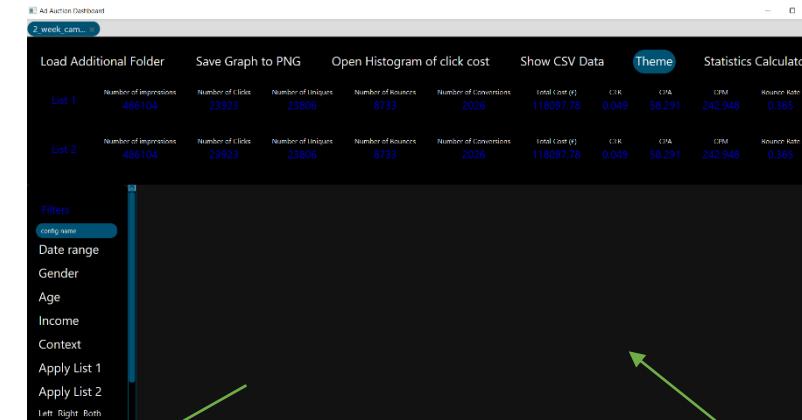
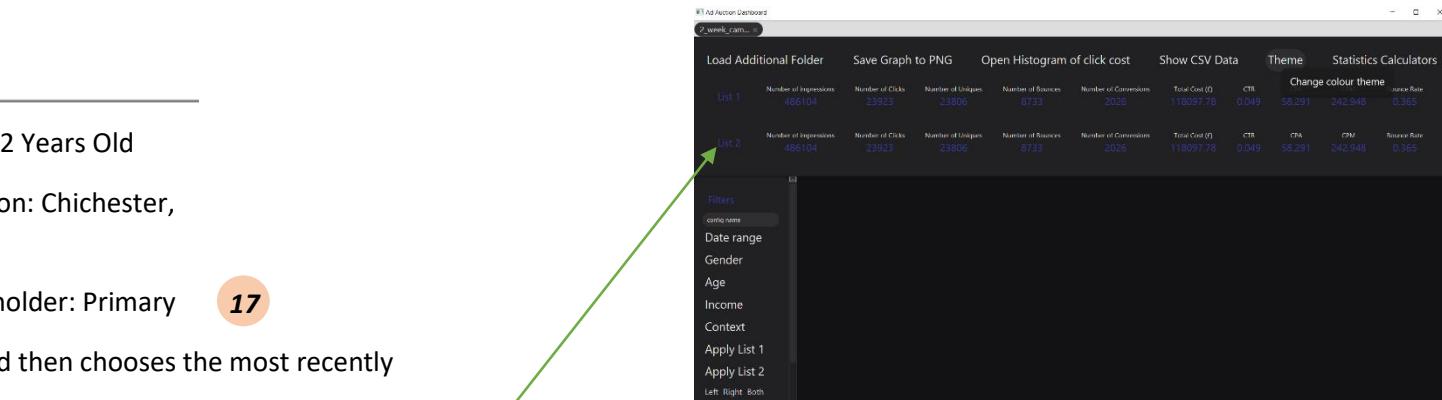
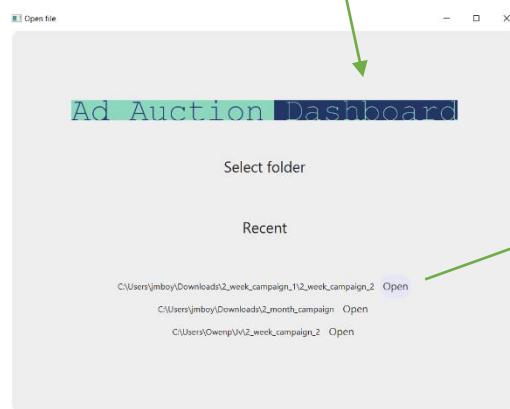
-Anna opens up the software, and then chooses the most recently opened dataset

-The main screen is then loaded, with the usual display.

-Anna presses the customise the appearance button, which displays a new screen with different settings.

- The first new screen she would get is dark mode
- The second screen that appears is high contrast mode
- Finally it loops back to the original screen

-Anna chooses to leave it on the dark mode



Testing Program Against Scenarios (Dennis)



Dennis

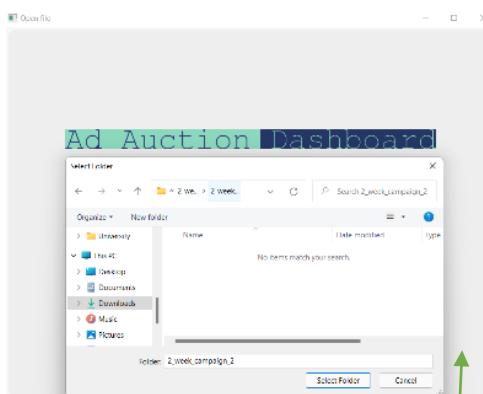
Age: 17 Years Old

Location: London, UK

Stakeholder: Primary

15

- Dennis opens up the software, then chooses the dataset he would like to use
- The main screen opens, with the usual statistics showing
- Dennis then presses the search new statistic button
- Then he can select the statistic type (and define which variables are used depending on the statistic) and the result is shown.



Statistics Calculators

Open window of statistics

242.948

0.365

Binomial Probability Distribution Calculator

Enter number of successes: 10

Enter number of trials: 20

Enter probability of success: 0.55

Calculate

0.15934945550

Binomial Probability Distribution Calculator

Enter number of successes: |

Enter number of trials: |

Enter probability of success: |

Calculate

Answer

Statistics Menu

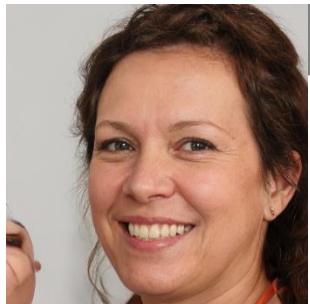
Binomial Probability Distribution

Poisson Probability Distribution

Binomial Cumulative Distribution

Poisson Cumulative Distribution

Testing Program Against Scenarios (Claire)



Claire

Age: 52 Years Old

Location: Sussex, UK

Stakeholder: Secondary

14

-Analyst opens up the software, and then chooses the dataset they would like to use.

-Analyst then presses the open button, which opens up what looks like the same display, but it would be without any anomalies as that process would have run in the background

The screenshot shows the Ad Auction Dashboard interface. At the top, there are several menu options: Load Additional Folder, Save Graph to PNG, Open Histogram of click cost, Show CSV Data, Theme, Statistics Calculators, and Change colour theme. Below these are two data tables labeled List 1 and List 2, each displaying various metrics such as Number of Impressions, Number of Clicks, Number of Uniques, Number of Bounces, Number of Conversions, Total Cost (€), CTR, CPA, CPM, and Bounce Rate. A green arrow points from the 'Select Folder' dialog box to the 'Load Additional Folder' menu item. Another green arrow points from the 'Filters' section to the 'Change colour theme' menu item.

	List 1	List 2
Number of Impressions	486104	486104
Number of Clicks	23923	23923
Number of Uniques	23806	23806
Number of Bounces	8733	8733
Number of Conversions	2026	2026
Total Cost (€)	118097.78	118097.78
CTR	0.049	0.049
CPA	58.291	58.291
CPM	242.948	242.948
Bounce Rate	0.365	0.365

Select Folder

Organize Now folder

Date range

Gender

Age

Income

Context

Apply List 1

Apply List 2

Left Right Both

Testing Program Against Scenarios (Eddie)



Eddie

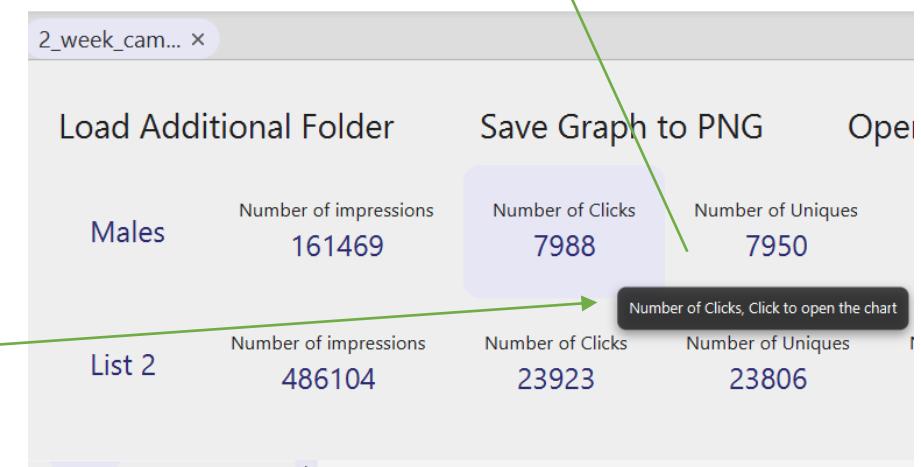
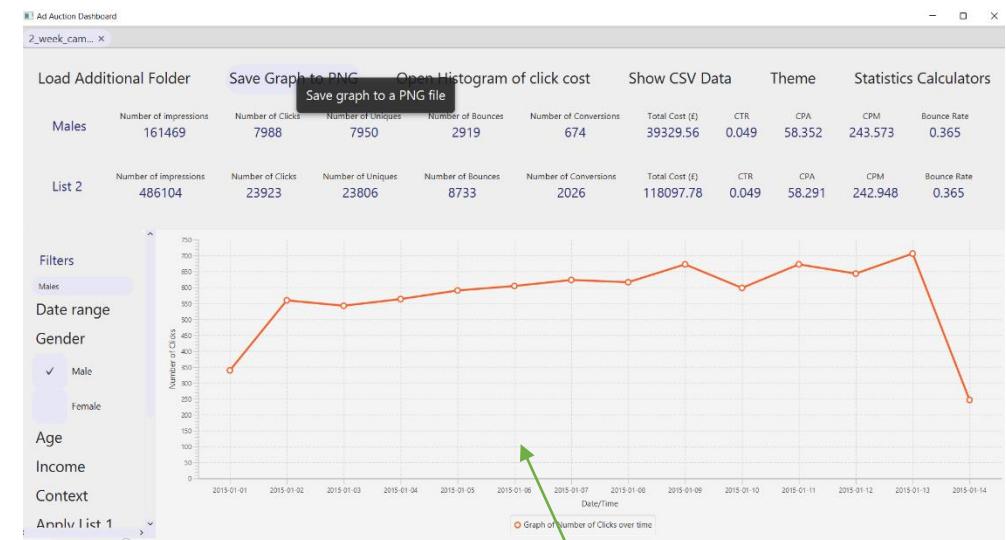
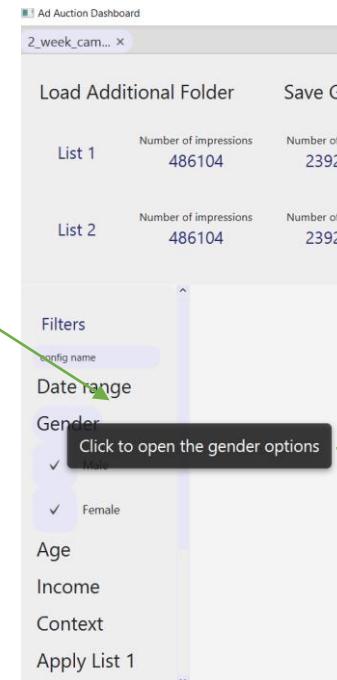
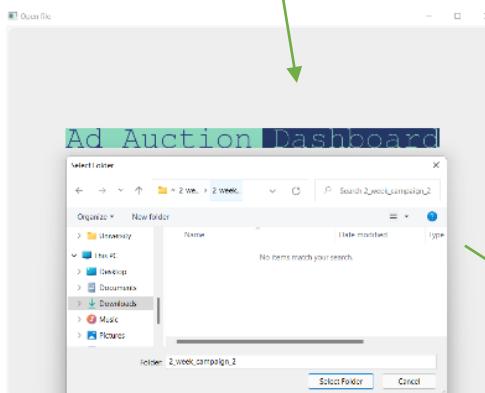
Age: 44 Years Old

Location: Stockholm,
Sweden

Stakeholder: Secondary

19 22

- Eddie opens up the software (using the executable file), and then chooses the most recently opened dataset
- The main screen is then loaded, with the usual display.
- Eddie then wants to make a graph for his clients, so he uses the tooltips on this screen to see how to use the filters for a graph he wants to use no filters on
- Eddie then makes the graph, saves it to a file (using the helpful tip in the tooltips)



Sprint Plan For Current Increment

Who?	James	JMA	Owen	Philipp	Sandor	Tianyu	ALL
What?	15. Well known common statistics	14. To easily find and remove anomalies	21. A search bar	17. To customise the appearance	9. To compare graphs 22. An executable file	19. A help button	11. Fast and responsive software 13. Aesthetically pleasing GUI

BURNDOWN CHART

■ To compare graphs

■ A help button

■ Well known common statistics

■ Combining all different tasks into the one file

■ An executable file

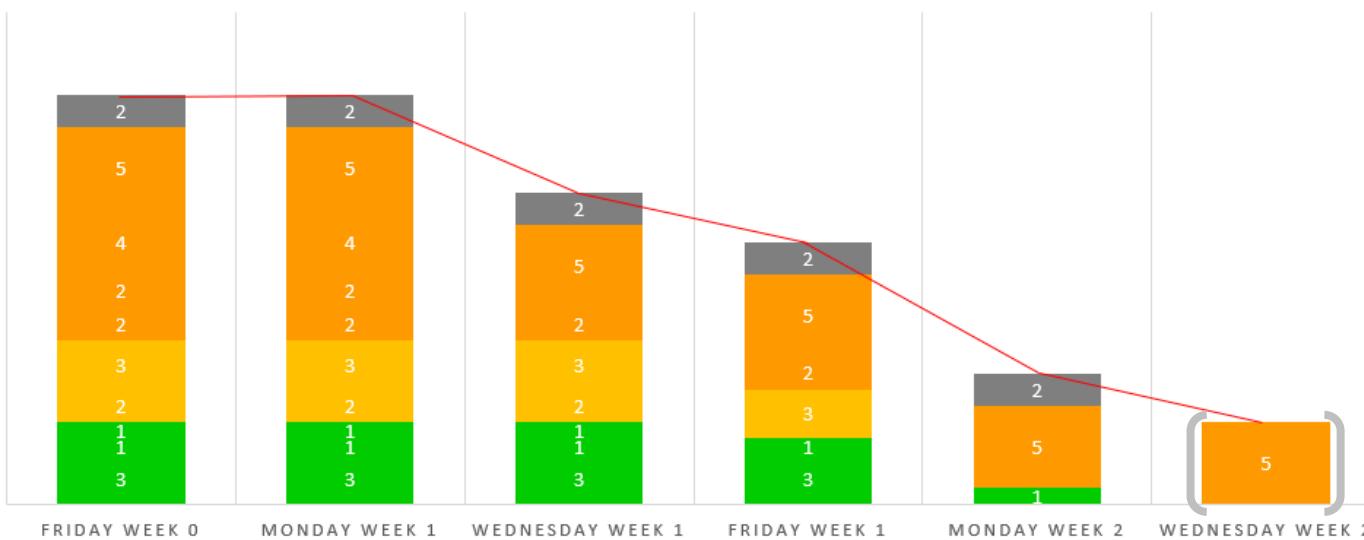
■ Aesthetically pleasing GUI

■ To customise the appearance

■ Fast and responsive software

■ To easily find and remove anomalies

■ A search bar



Sprint Leaders – JMA and Tianyu

Split into 2 groups:

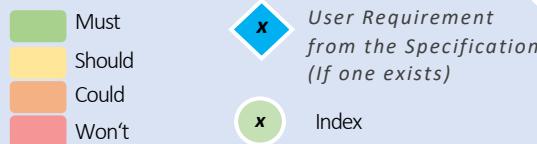
--- Statistics/Graphs (James, **JMA**, Sandor)

--- Usability (Owen, Philipp, **Tianyu**)

Task	Estimate Time	Actual Time	Completed Date
	(MAX)		
9	6 hours	8 hours	30/04/2022
11	2 hours	2 hours	28/04/2022
13	12 hours	12 hours	01/05/2022
14	4 hours	4 hours	26/04/2022
15	6 hours	12 hours	02/05/2022
17	8 hours	7 hours	27/04/2022
19	4 hours	5 hours	29/04/2022
21	10 hours	N/A	N/A
22	2 hours	1 hour	04/05/2022
Merge	4 hours	3 hours	03/05/2022
Total	56 hours	54 hours	04/05/2022

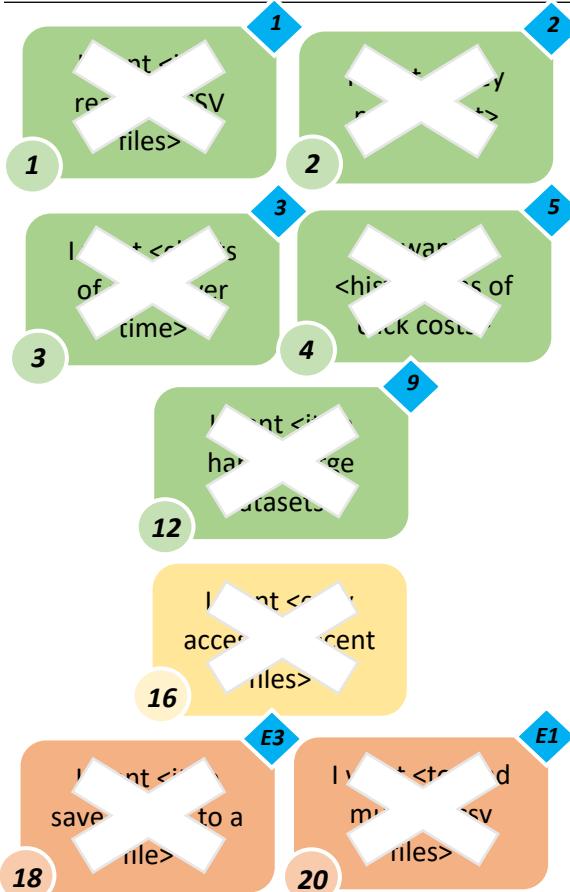
Definition of “done”: When we say a task is “done”, we mean that we have tested the user story against all the tests we designed for it in the testing section, and the acceptance criteria we made for it, and it had passed, so the user story it was for is then completed in our program.

Key

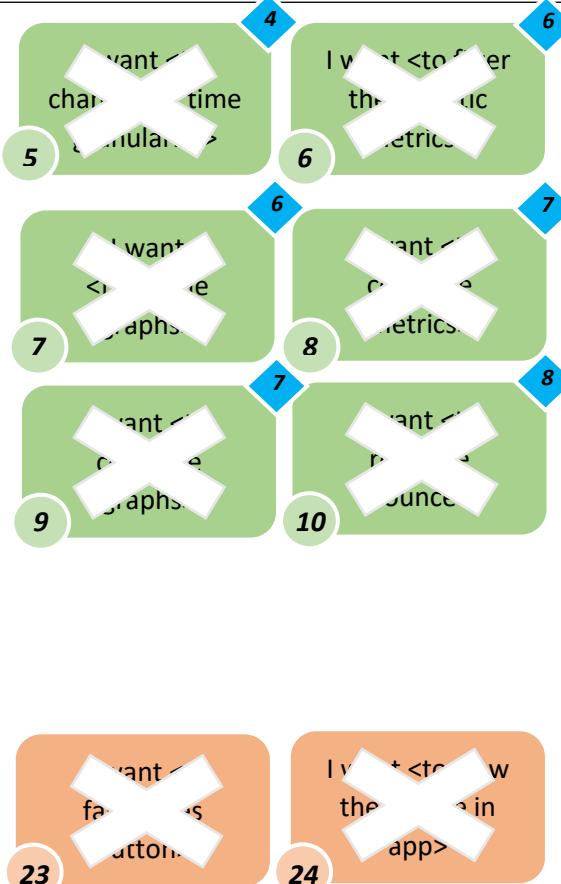


User Stories In Increments

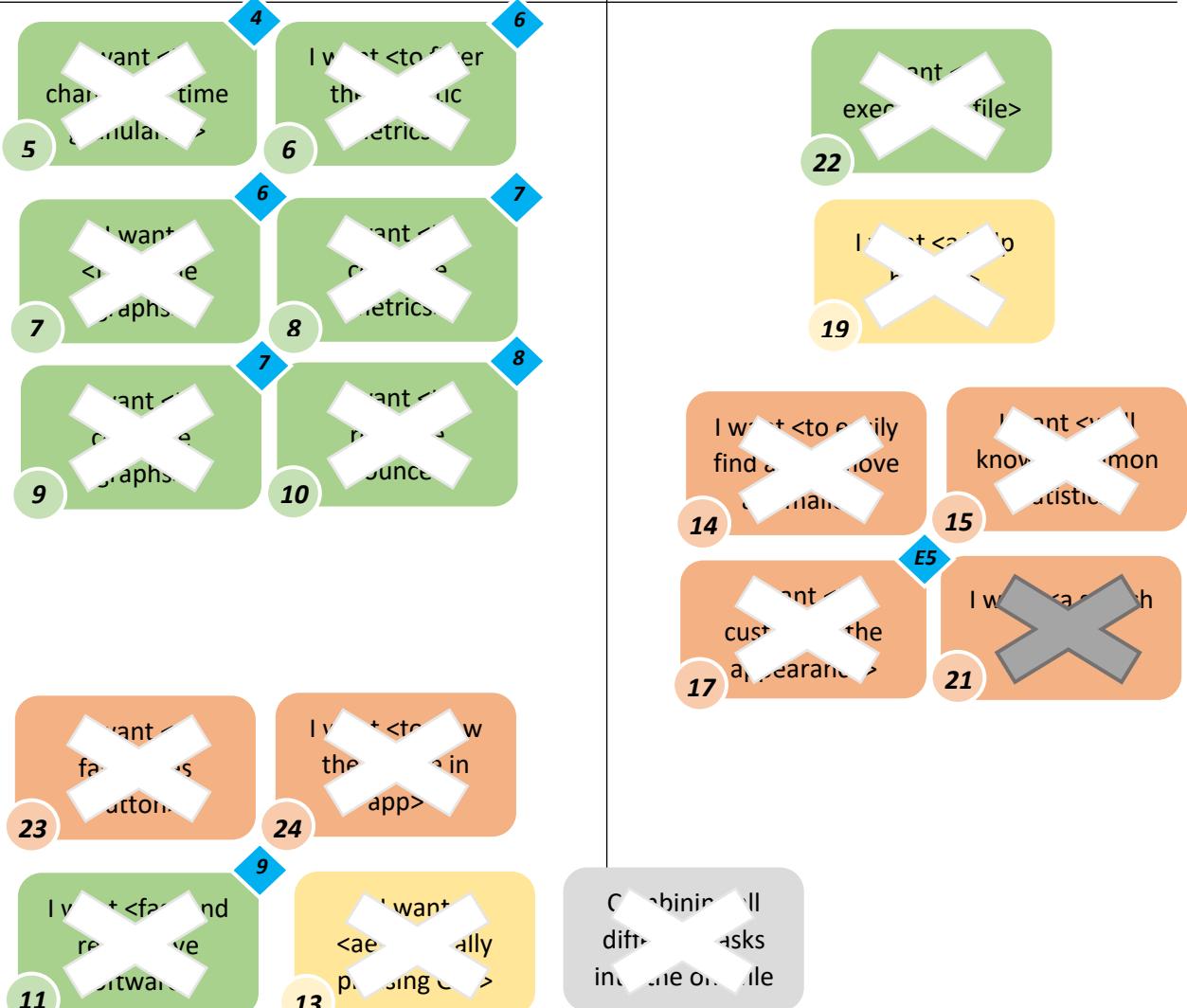
Increment 1



Increment 2



Increment 3



Included In All Increments

Reflection on Feedback

Feedback this is based on	Our implementation
<p>The application does not necessarily need two identical menus that do the same thing, it seems to be a little bit tedious and space on the window could better be used without it. Try and keep similar controls in one place, otherwise, it might become a little bit confusing.</p>	<p>We have changed the implementation of the two filter menus so that only one of them are shown, with two buttons below it instead to make it more useable. Likewise, all buttons use the same formatting in order to keep it clear to their purpose (Application section of the document)</p>
<p>I would like to see in your scenarios more of the reasons *why* a particular persona wants to do what they're doing. How does their relevant demographic affect how they go about doing this? Might they do something that breaks your application or causes some unintended behaviour?</p>	<p>We have edited our scenarios to make them more informative and detailed, in looking into the motives of the people using the software, to highlight any possible extra bugs or problems that we may have not already addressed (page 11-12)</p>
<p>A sentence to explain that the little circles are direct links to how you've modelled user stories (where appropriate) in your design artefacts would be helpful.</p>	<p>In order to specifically define what the little circles are representing, we have added not only a definition of them, we have also created a table which links every user story to their respective user requirement (page 2)</p>
<p>Make sure all design elements in the documents are clearly and correctly an honest interpretation of my original spec on the document.</p>	<p>All of our design artefacts are shown as the interpretation we took from the requirements, and have also been verified and checked by our supervisor to make sure they are how he envisions the project. It is also labelled with where the user stories are being referenced/fulfilled in our project (Design section of the document)</p>
<p>For testing, follow the following acceptance criteria generation examples and map it to pass or fail classification {followed by instructions on how to make acceptance criteria}</p>	<p>For each of the user stories we have, we have made an acceptance criteria assessment based on the guidelines given, as well as our supervisor's opinions, which we then used to test our code against each user story (on pages 2-7, and page 19-28)</p>
<p>Make sure for next time that your definition of done is modified slightly to ensure that each User Story has met its acceptance criteria</p>	<p>The definition of done has been updated to reflect that each user story passes its acceptance criteria (page 34)</p>