VISUALIZING MICE BEHAVIOR WITH HEAT AND VECTOR MAP GENERATION

SOFTWARE REQUIREMENTS DOCUMENT

Compiled by: Kevin Hartnett, Colton Hall, Parker Rowland
Version 1.0
02/08/2017

TABLE OF CONTENTS

1. INTRODUCTION

1.1. Purpose of the Software Requirements Document

The software requirements document details all information pertinent to the required functionality that the project must possess once implemented. The information within this document provides an in-depth description of project background, stipulations, and other details in order to give the reader a high level overview of the product. The intended audience of this document is comprised of the clients, stakeholders, and developers of the system.

1.2. Scope of the System and Subsystems

The system involves several key subsystems that must interoperate to produce the visualizations. A graphical user interface (GUI) provides the user with the means to operate the system. A data management subsystem enables the user to load datasets into the system for the purpose of producing visualizations. The heat and vector mapping component utilizes the necessary mathematical and scientific formulae for rendering images and animations of the loaded data set. Finally, an export manager gives the user the ability to save the rendered heat and vector map graphics as image files for use outside of the system. The requirements within this document of the system and its subsystems originate from discussions with the client.

1.3. References

Dr. Polack provided the necessary descriptions of functionality that the system should possess at its implementation.

1.4. Overview of Remainder of Document

The remainder of this document is roughly divided into four sections of information. The first is an overall description of the program that includes background information, descriptions of the client and user, as well as a list of use cases that demonstrate the system's intended usage. The second section is devoted to the requirements and non-requirements of the system, which is an analysis of the system's capabilities in closer

detail. The penultimate section provides a list of assumptions made about the system and its usage. The last section contains an appendix of information containing a glossary of terms and a list of contributions to this documentation made by each team member.

2. OVERALL DESCRIPTION

2.1. Client Characteristics

The client is an interdisciplinary team of biology, psychology, and computer science researchers investigating the feasibility of developing an artificial intelligence model for group mouse behavior. To help determine the presence of any significant patterns in the experiment data, the client needs a tool that can generate heat maps and vector maps (line paths) of mouse activity over time. References to the client of the system in this document are defined, unless otherwise noted, as references to our point of contact with the client, Dr. Polack, who is one of the researchers.

2.2. User Characteristics

The user of the system is defined as one of the researchers that comprise the client group. The user is characterized as being a professional academic in their respective field of study with a basic understanding of computer operation.

2.3. Data Source Characteristics

The data sets that the system is capable of processing contain the raw data captured from experiments conducted by the researchers. The experiment involves a fixed enclosure divided into grid sections, each with an Radio Frequency Identification (RFID) component, in which a group of one to eight mice traverses for a period of time. If a particular mouse remains stationary over a grid section for a period of time greater than a specific minimum threshold duration (such as 100 milliseconds, for example), a data entry of the amount of time the mouse remains in that grid section begins and is committed to the data set once the mouse leaves that grid section. If a particular mouse moves faster than the minimum threshold duration through a grid section, a data entry with a

zero value is committed to the data set once the mouse leaves that grid section. In this way, the data sets are comprised of the stationary activity and transient movements of each individual mouse over a period of time. The format of the data sets is Comma Separated Value (CSV).The sizes of the data sets are not upper bounded, and include files as large as 35 megabytes (approximately 70,000 data entries).

2.4. Product Functions

The product functions of the system allow the user to load and visualize input data as well as create export images of the generated visualizations. The user interacts with the system through the graphical user interface to access its functions. The following use cases are descriptions of how particular core functions of the system work:

2.4.1. Load Dataset

Description: A user wants to load a new data set into the system.

Main Flow:

User indicates that they want to load a new data set. The system opens the operating system's file explorer, prompting the user to select a CSV file to load. If the file selected is in CSV format and if the data within the file passes a validation check, the system prompts the user to enter the grid size associated with the experiment data in the file. The system then prompts the user for a name to save the data set as, and adds the file path and grid information to its internal database for quick use in future sessions.

Alternate Flow A:

If the user selects a file that is not a CSV file, then the system displays message to the user stating that the selected file type is invalid and that only CSV files can be accepted.

Alternate Flow B:

If the data within the selected CSV file fails to pass the validation check, then the system displays a message to the user stating that the selected file type is invalid and that only CSV files with experiment data will be accepted.

2.4.2. Delete Dataset

Description: A user wants to delete a previously loaded data set.

Main Flow:
User indicates that they want to delete a data set from the System's list of previously loaded data sets. The system displays the Data Set Manager section, which contains a list of previously loaded data sets. If the user selects the delete GUI control next to a particular data set, the system will prompt the user with a confirmation dialog asking if they would really like to delete the data set. If the user selects the affirmative choice, the system deletes the file association of the data set from its internal database and displays a success message to the user.
Alternate Flow A:
The system has no previously loaded data sets.
Alternate Flow B:
If the user selects either the negative or cancel choice in the prompt, then the system cancels the delete file association operation.

2.4.3. Generate Visualization

Description: A User wants to generate a heat map or vector map visualization.
Main Flow:
User indicates that they want to generate a visualization graphic. The system allows the user to specify certain parameters influencing the visualization, including map type, number of mice to graph, and static / dynamic graphic generation ("at once" vs. animated over time). Once the user has selected their parameters, the user activates the GUI control that begins the visualization generation process. The system generates the visualization, and the ability to export the visualization becomes available to the user.

2.4.4. Export Visualization

Description: User wants to save a visualization in an image file format.
Main Flow:
The user indicates that they want to export the current or selected visualization as an image file. The system opens the operating system's file navigation window asking the user what to name and where to save the image. The system generates the image, saves it

to the specified location given by the user, and displays a success message to the user.

Alternate Flow A:

If the user closes the operating system's file navigation window, then the system cancels the image export operation.

2.5. General Constraints

The system is to be implemented in Java at the request of the client. There are no time constraints regarding the generation of visualizations, nor in the loading and processing of data sets. Additionally, there is no upper bound on the amount of data that the system must be able to process. All data sets that the system is able to process are in CSV format.

3. Requirements

The requirements of this program are broad in nature; they extend from very general requirements that lay out the absolute basics (type and parameters) to specific implementation details. Because it’s targeted largely at non-technical users, it has many GUI requirements to make the crucial functions accessible. It must be able to handle exceptions without crashing, and it has to be able to perform data analysis after proper parsing to generate vector graphs and heatmaps of changes of mouse positions. Finally, the program must be able to render the output as an image and print.

3.1. General Requirements

Requirement 3.1.1.

The program shall track the paths of a variable number of mice based on heat signatures detected by the application. The program shall be designed and written as a Java application, being formatted correctly to run on Mac and Windows. All dataset grid dimensions shall be set by the user and returned to the display screen.

Rationale: The program must be able to run cross-platform.

3.2. Capacity Requirements

Requirement 3.2.1.

The program shall upload any size of CSV file. The capacity used from the program will vary based on the csv files uploaded, but shall be handled depending on the system's specifications and free space.

> Rationale: The program must not be bounded by a certain upper limit of size of the chosen CSV file.

3.3. User Interface Requirements

Requirement 3.3.1.

The program shall have a graphical user interface that interacts and responds based on the user's input to the system. There shall also be a menu bar with several options that include: "Load Dataset", "Pick Previous Dataset", "Set Grid Dimensions", and "Manage Dataset". In the "Pick Previous Dataset" option, the program shall remember both the grid size and the grid filenames for easy access to reopen. In the "Set Grid Dimensions" option, the user shall be able to set the number of rows and columns for the intended grid size. In the "Manage Dataset" option, the user shall be able to edit individual dataset entries from within the program. In addition to the menu bar the program shall also have textbox options that include: "Start", "Stop", and "All". The "Start" and "Stop" option shall be used to denote the time in which to track the mice. The "All" option shall be used to draw the lines tracking the mice paths from the very first instance. The side of the program shall include three buttons a user may toggle; "Pick Mice to Display", "Heat Map", and "Vector Map". The option to "Pick Mice to Display" shall give a list of the mice's RFID tags currently in the system, and allow the user to toggle which mice are displayed. The option "Heat Map" shall allow the user to toggle the heat map on the display screen. The option "Vector Map" shall allow the user to view either a dynamic or static graph with the mice as vertices.

> Rationale: The program must be easy to use, because it has to be accessible from a non-technical standpoint.

3.4. Error Handling Requirements

Requirement 3.4.1.

The program shall provide a mechanism to validate that the file a user has selected is readable and contains all required information. This requirement shall be triggered upon the user selecting the “Load Dataset” menu option. It must make sure that the file is in the proper format and that the contents are all proper.

Rationale: This exception handler needs to be in place to ensure smooth running of the program. If it is not there, the program will behave unpredictably. Data validation in the beginning will satisfy the preconditions of every other component of the software, so there won’t be any worry down the line of whether sound data was read in at the beginning of the program.

3.5. Input Requirements

Requirement 3.5.1.

The program shall only accept valid CSV files as datasets. It will not accept Microsoft-proprietary files, such as XLS and XLSX files. The data validation is covered separately in requirement 4.1.

Rationale: There must be a standard type of input for the system. While some research files are XLS and XLSX files, the file input is a CSV file that has been generated by other files. This removes any reliance on proprietary file types, and the project only needs position and string data, both of which are provided in a minimal CSV file. This input must be error-checked via Requirement 3.5.1.

3.6. Processing Requirements

Requirement 3.6.1.

The program shall have the ability to generate a heatmap based on input data.The heatmap shall have an eight-bit color scheme, based on traditional colors of a heatmap; colors must include red, orange, yellow, green, blue, purple, and shades therebetween. The heatmap should increase in “hot” color the longer a mouse remains in that position. For the purposes of this program, if a mouse transiently passes through a grid piece, no heat map data is generated. *The heatmap may not contain the*

color “cyan” or any of its derivatives. The heatmap must be overlayed on the existing grid, and the client made no requests regarding resolution of the heatmap.

> Rationale: An automatically generated heatmap will give researchers an idea as to where mice tend to stay the longest. The heatmap also gives an instant visualization of the dataset that is initially loaded into the program.

Requirement 3.6.2.

The program shall have the ability to generate a map of travel based on the input data. The travel data must contain the following: vertices and edges that represent the approximate path that each mouse took. Each mouse’s path must be a separate color. These paths must resemble natural motion, and they cannot be straight lines . This must be able to be overlaid on an existing heat map, and it must still be a clearly visible diagram. This may be drawn in simulated real-time, or it may be output as one static drawing at the end.

> Rationale: This will allow researchers to instantly visualize where specific mice were traveling at certain times. The natural motion gives a realistic representation, and it can make experimentation more precise. It also adds more functionality to the program to make its output more meaningful.

Requirement 3.6.3.

The program shall have the ability to show movement using time weighting rather than mean location. While drawing out a trail of mice movement, the program must utilize all of its data points to recreate as lifelike of a rendition of a mouse’s movements as possible. The final paths should appear rounded and not linear.

> Rationale: This is so that meaningful research can be done on where the mice are going. Estimated lines do not help, and they cannot give a realistic representation.

3.7. Output Requirements

Requirement 3.7.1.

The program shall be able to output its final results, as selected by the user, to an image file. After the manipulations have been done, the program must be able to capture a rendition of what it's currently displaying. This may be the heatmap and/or the final result of the mouse travel animation. The program should generate an image of at least 300 DPI in a JPG or PNG format. This does not need to be saved in a specific folder, nor do there have to be any graphical options to do so. This is solely for internal purposes. The image should be stored in a temporary location.

> Rationale: This creates a basis for the program to print the results from. It makes it easy for the next requirement to be implemented, and it standardizes the output for any future manipulations that may be required.

Requirement 3.7.2.

The program shall have the ability to print out the output. After the processing has finished, and the heatmap and/or the mouse travel animation have completed, the program should have the ability to print the output. Upon printing, Requirement 7.1 shall be executed, creating an image. Once the image has been created, the program shall send it to the printer spool so that the results may be physically put on a piece of paper.

> Rationale: This requirement ensures that researchers will always be able to put their results into physical notebooks, prevents against data loss, and makes the output extendable.

4. Non-Requirements

In this section is stated the different Non-Requirements, intended by the client, that will not be implemented in the program. The Non-Requirements section will include details involving Network, Security, and other Miscellaneous activities.

4.1. Network

Non-Requirement 4.1.1.

The program shall not need any network access to function. There shall be no need to connect to any source outside the program other than the files being uploaded. The program shall not be able to connect to any cloud service to print information.

4.2. Security

Non-Requirement 4.2.1.

The program shall not need any security protocol as there is no access to any network. There shall not be any account information that reveals any personal details of the user. The shall be no malicious activity that can be reported from this program.

4.3. Miscellaneous

Non-Requirement 4.3.1.

The program shall not be bound by any generation time that is dependent on displaying information the user has entered.

5. Assumptions

- The machine(s) this program is run on must have the Java Runtime Environment installed. No machine that does not have this installed must be supported.

- The datasets loaded into the program are of consistent format so they can be parsed and output reliable data.

- It is assumed that the researchers using this program are using a computer with enough onboard memory and processing power to store and parse/evaluate very large files and make real-time renderings of said files.

6. APPENDICES

6.1. Glossary

Comma Separated Value (CSV): A type of data format associated with text that consists of rows of data separated by line breaks. Within the rows of data, individual values are delimited by a value separator, which can be any character so long as the character does not appear within the values of the row data and only if that delimiter is used consistently throughout the file.

Data Set: A collection of related sets of information that is composed of separate elements but can be manipulated as a unit by a computer (Google definition).

Graphical User Interface (GUI): The visual component of a software program that allows users to interact with the software.

Heat Map: A graphical representation of the aggregate intensity of specific data parameters projected onto a data matrix, where the axes are particular parameters. In this project, the axes of the heat map reflect the dimensions of the mouse enclosure used in the experiment.

Radio Frequency IDentification (RFID): RFID is a technology that implements tags (transponders) containing information and readers that capture that data from the tags (transceivers). A reader sends a radio signal broadcast specific to the RFID application, and if there are any tags in the vicinity of the broadcast, the tags send their information back to the reader in the form of passive backscatter or active radio transmission. RFID is commonly used in asset tracking applications. In this project, RFID is responsible for generating the data of mouse location and duration of time spent within the sectors of the enclosure.

System: The system is the software program being built for the client.

User: The user is the person using the system.

Vector: A quantity having direction as well as magnitude, especially as determining the position of one point in space relative to another (Google definition).

Vector Map: A graphical representation of the path vectors traveled over time. In this project, the vector map will show the paths traveled by individual mice within the enclosure.

6.2. Team Member Contributions

All team members met throughout various times within the deadline to discuss the layout and the intended information to comprise this Requirements Document. Team members may have also contributed to this document in other minor details that are not listed in the following.

Parker Rowland
Met with Dr. Polack to discuss the layout of the program and all requirements needed for the program to run as intended. Filled out the information written in the “Table of Contents”, “Introduction”, “Overall Description”, and “Glossary” sections of the Requirements Document

Kevin Hartnett
Met with Dr. Polack to discuss the layout of the program and all requirements needed for the program to run as intended. Filled out the information written in the “Requirements” (3.4-3.7) and “Assumptions” sections.

Colton Hall
Filled out the information written in the “Requirements” (3.1-3.3), “Non-Requirements”, and “Team Member Contributions” sections.