PLASMA VISUALIZATION GROUP

PlasmaGraph

Software Requirements Specifications

Daniel E. Quintini & Gerardo A. Navas

21/11/2013

This document explains the necessity of developing a software tool that can help in the analyzing of experiment's data for the plasma research team at the Polytechnic University of Puerto Rico as well as specifying what is expected of said software.

Contenido

1. Introduction 2
1.1 Purpose 2
1.2 Scope 2
1.3 Definitions, acronyms and abbreviations 3
1.4 References 3
1.5 Overview 3
2. Overall description 3
2.1 Product perspective 3
2.1.1 System Interfaces 3
2.1.2 User Interfaces 4
2.1.3 Hardware Interfaces 5
2.1.4 Software Interfaces 5
2.1.5 Memory 5
2.1.6 Site Adaptation Requirements 6
2.2 Product functions 6
2.3 User characteristics 6
2.4 Constraints 6
2.5 Assumptions and dependencies 6
3. Specific requirements 6
3.1 External interfaces 6
3.2 Functions 7
3.3 Performance requirements 13
3.4 Logical database requirements 13
3.5 Design constraints 13
3.6 Software system attributes 13
3.6.1 Reliability 13
3.6.2 Maintainability 13
3.6.3 Portability 13
3.7 Organizing the specific requirements 13
3.7.1 Objects 14
3.7.2 Features 14
3.8 Additional comments 14

1. Introduction

1.1 Purpose

This document specifies and explains the necessity of developing a software product hereinafter referred to as “PlasmaGraph” or “the product”. It also specifies the features and functions that the product delivers as well as providing a general explanation of what PlasmaGraph can and can’t do. The information contained in this document is intended to help Angel E. González-Lizard, Ph. D hereinafter referred to as “the client” to understand the product. Another use for this document can be as a starting point and guide of reference along the development process for Daniel E. Quintini and Gerardo A Navas hereinafter referred to as “the developers”.

1.2 Scope

PlasmaGraph is intended to assists in the job of analyzing figures collected in experiments by providing a tool capable of analyzing a file with tabular data and produce a visual representation of the numbers within that file. In other words, a program with a graphical user interface that makes chart graphs.

Chart graphs help people spot pattern easier and faster. But because there are many ways of making a graph with the same data, PlasmaGraph is going to need an additional file that specifies the following missing information:

- Name of the graphic
- Horizontal headers
- Vertical headers
- Columns
- Data order (ascending /descending)
- Type of chart (bar, line, pie)

This file will be referred to as “template” and each time a new one is created, it will be saved in PlasmaGraph’s database so it can be used later.

Templates can be modified or deleted. Also, once the chart graph is created, the person using this product will be able to save it in his or her local file system.

In addition to all this, PlasmaGraph is able to catch and fix certain errors in the data that needs to be represented visually.

What the product does:

- Create a chart graph.
- Provide a graphical interface so the user can interact with the product.
- Create and save templates.
- Modify or delete templates.
- Alert about possible errors in the data that has to be graphed.

- Fix error or suggest solutions.

What the product does not do:

- Process files that are not formatted as tabular data.
- Catch 100% of possible errors.
- Fix 100% of detected errors.

1.3 Definitions, acronyms and abbreviations

*.png, GUI, software, *.csv, database, encapsulation, OS, mb, data integrity, tabular data, java, widget toolkit, programming language, code conventions, XML, distributed version control system

1.4 References

1.5 Overview

The rest of this document is divided in two sections which are overall description and specific requirements. The first describes the general factors that affect PlasmaGraph and its requirements. The second contains the requirements in a level of detail sufficient enough to enable the developers to design and test a system that meets the objectives defined.

The overall description of the product is divided into five subsections that defines the product's perspective and functions, user characteristics, constrains, assumptions and dependencies. Meanwhile the specific requirements section spawns eight subsections covering external interfaces, functions, performance requirements, logical database requirements, design constrains, software systems attributes, organization of the specific requirements and any additional comments.

2. Overall description

PlasmaGraph is a software tool that can read a file with tabular data and create a chart graph. Because the tabular data alone won't be enough information to make the chart graph; PlasmaGraph also uses other files called templates. Templates have this additional information needed to make the graph.

2.1 Product perspective

There are several data visualization software products in the market like Tableau and Visua.ly but they are expensive and come with too many options which results in a steeper learning curve for the person using the product. PlasmaGraph in the other hand is specialized to meet the client's requirements making it simpler and easier to use.

2.1.1 System Interfaces

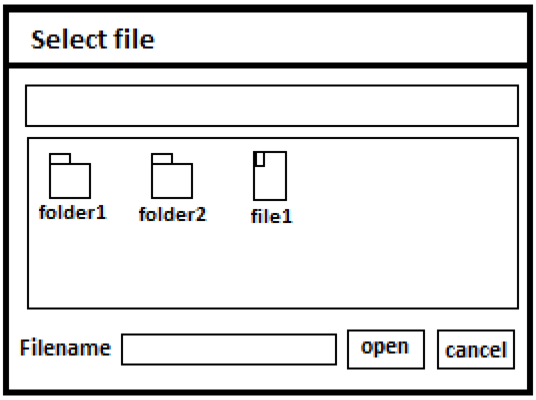
PlasmaGraph is a composite of interfaces working together in order to transform a set of data into an image that is a graphical representation of that data. The two predominant interfaces are the user interface and the software interface:

Graphical User Interface (also referred as GUI): This is the interface that interacts with the person using the program. It is made of a series of windows populated with fields and buttons that enable the user to tell the program what to do.

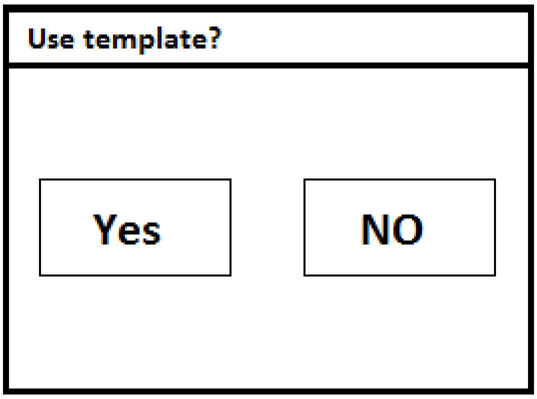
System Controllers: It is made by a collection of functions that receive data from or send data to the GUI in order to perform the operations necessary to generate the chart graph or report the problem.

Database Abstraction: This interface encapsulates all the processes necessary for communication with the database so System Controllers can use it to insert, update, retrieve or delete information.

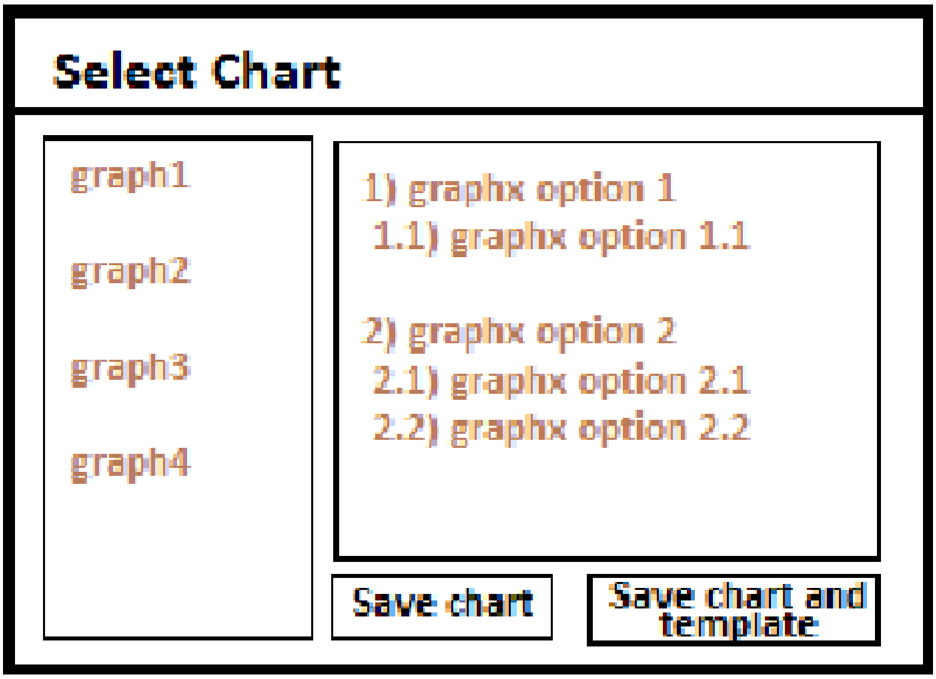
2.1.2 User Interfaces



When PlasmaGraph starts, the user is going to see a window asking him/her to select a file. Using this window, the user can select the file containing the data that needs to be graphed.



Once the user selects the file, another window is displayed asking if a template is going to be used. When the selection is made, the system proceeds to check the file for problems. If a problem is found, a message with a brief description of the problem is shown to the user. If the option selected was ‘Yes’, the system will provide another window with a list of charts to select from and once the chart is chosen, a set of options is provided inside the window so the user can control how the data is going to be displayed in the chart (charts and their configurations are TBD until a meeting with Angel E. González-Lizard, Ph. D is scheduled to resolve this problem). If the option selected when asked to use a template is ‘No’, then the same window with a list of charts will appear but the chart and all its options will be already selected. This last window also has two buttons, one to save the chart and another to save the chart and the options used as new template.



When the user proceeds to save the chart/template, the system will use the data provided to make an image file and display a window asking the user where to save the image. If a problem occurs, then a window with a brief description of the problem is shown and the PlasmaGraph program is terminated.

2.1.3 Hardware Interfaces

The minimum recommended hardware requirements for PlasmaGraph are as follow:

- 15’’ Monitor with a resolution of (1200x700) and a refresh rate of 30Hz.
- Any keyboard and mouse compatible with your OS.
- Two 3.20GHz processors.
- 1GB of available memory.

2.1.4 Software Interfaces

PlasmaGraph on the following external packages and software systems:

- Java SE runtime environment 7u45 available at http://www.oracle.com/technetwork/java/javase/downloads/jre7-downloads-1880261.html.
- Opencsv 2.3 availiable at http://sourceforge.net/projects/opencsv/
- JFreeChart 1.0.16 availiable at http://sourceforge.net/projects/jfreechart/files/

2.1.5 Memory

TBD (for this section to be determined, a prototype of PlasmaGraph should be evaluated first)

2.1.6 Site Adaptation Requirements

For PlasmaGraph to function, Java 7 must be installed and running.

2.2 Product functions

This product performs the following functions:

- Load a file
- Create chart graph
- Create template
- Save chart graph
- Save template
- Update template
- Delete template

[use case dia]

2.3 User characteristics

The product is used by Angel E. González-Lizard, Ph. D as a tool to analyze experiment's data from the plasma laboratory at the Polytechnic University of Puerto Rico. The user needs to know how to interact with a window based interface and how to read the chart types provided.

2.4 Constraints

The product will only recognize a handful of errors in the data provided by the user and even then it won't be able to fix all the problems (specifics of these problems are TBD until a meeting is scheduled with Angel E. González-Lizard, Ph. D to resolve the issue). It will also limit the type of charts to bar, line and pie and the amount of rows and columns to TBD

2.5 Assumptions and dependencies

For PlasmaGraph to function, Java 7 must be installed and running.

3. Specific requirements

3.1 External interfaces

CSV Sheet
Description:
Input source:
Valid format:
Units of measure:
Timing:
Relationships:
Window format:
Data format:
Command format:
End message:

3.2 Functions

Load File

Actors:

- User
- OS

Scenarios:

	Principal	Alternate: (A1) “Original file had errors”	Exception: (E1) “The user didn’t want to load the file”	Exception: (E2) "Error Loading file"	Exception: (E2) "Bad data"
1	The system asks User to specify the address of the file.				
2	The user gives a file’s address to the system.				
3	OS asks the system for a file’s address.				
4	The system gives the address specified by the user to OS.				
5	OS gives the system the data contained in the file.				
6	The system tells the user that the file was loaded successfully.	The system tells the user that the file has one or more errors.	The system tells the user that the file has one or more errors.	OS gives an error message to the system.	The system tells the user that the file isn’t valid.
7	X	The user tells the system to fix the errors.	The user tells the system not to load the file.	The system tells the user that an error occurred while loading the file.	X
8	X	The system tells the user that the file was fixed and loaded successfully.	X	X	X

Make Graph Chart

Actors:

- User
- Database

Scenarios:

	Principal	Alternate: (A1) “The user creates a new template”	Exception: (E1) "Unknown template"	Exception: (E7) "Template isn’t compatible with data set"
1	Load File.	Load File		
2	The system asks the user for a template’s name.	The system asks the user to make a new template.		
3	The user gives a template’s name to the system.	Make Template.		
4	The database asks the system for the name of a template.	The database asks the system for the name of a template.		
5	The system gives the name of the template specified by the user to the database.	The system gives the database the name of the template that the user just made.		
6	The database gives the system the template that matched the name provided.	The database gives the system the template that matched the name provided.	The database gives the system a message telling it that there isn’t any template with the name provided.	
7	The system tells the user that the graph chart was created.	The system tells the user that the graph chart was created.	The system tells the user the same message it received from the database.	The system tells the user that the graph chart wasn’t created because the template provided isn’t compatible with the data set in the file loaded.

Save File

Actors:

- OS

Pre-conditions:

- The system has an address.
- The system has a file.

Scenarios:

	Principal	Exception: (E1) "Invalid address"	Exception: (E2) "Error saving the file"
1	OS asks the system for an address.		
2	The system gives an address to OS.		
3	OS asks the system for a file.	OS tells the system that the address isn't valid.	
4	The system gives OS a file.	x	
5	OS tells the system that the file was saved.	x	OS tells the system that it wasn't able to save the file.

Save a Graph Chart

Actors:

- User

Scenarios:

	Principal
1	Make Graph Chart
2	The system asks the user for an address.
3	The user gives an address to the system.
4	Save File
5	The system tells the user that the graph chart was saved in the address provided.

Fill Template

Actors:

- User

Pre-conditions:

- Load File

Scenarios:

	Principal
1	The system asks the user for a template's name.
2	User gives a template's name to the system.
3	The system asks the user which columns are going to be included in the graph.
4	The user tells the system which columns he/she wants to graph.
5	The system asks the user what are the names he/she wants the columns to have.
6	The user tells the system the names of the columns.
7	The system asks the user which type of graph chart does he/she wants to make.
8	The user tells the system which type of graph chart he/she wants.

Make Template

Actors:

- User
- OS

Scenarios:

	Principal	Exception: (E1) “The template wasn’t created”
1	Fill template.	
2	Database asks the system for a template.	
3	The system gives a template to the database.	
4	The database tells the system that the template was saved.	The database tells the system that the template wasn’t saved.
5	The system tells the user that the template was created.	The system tells the user that the template wasn’t saved.

Update Template

Actors:

- User
- Database

Scenarios:

	Principal	Exception: (E1) "Database didn’t find the template"	Exception: (E2) "The template wasn’t updated"
1	The system asks the user for a template’s name.		
2	The user gives a template’s name to the system.		
3	The database asks the system for a template’s name.		
4	The system gives the database the name provided by the user.		
5	The database gives the system the template that matches the name provided.	Database tells the system that it didn’t find the template.	
6	Fill Template.	The system tells the user that there isn’t any template matching the name provided.	
7	Database asks the system for a template.	x	
8	The system gives the database the template that the user filled in step 6.	x	
9	Database tells the system that the template was updated.	x	Database tells the system that there was an error trying to update the file.
10	The system tells the user that the template was updated.	x	The system tells the user that the file wasn’t updated

3.3 Performance requirements

PlasmaGraph may only be used by one user at a time. The CSV sheet loaded for generating the graph chart needs to be less than 15mb and the time the system takes to create the file should be no more than 120 seconds.

3.4 Logical database requirements

The only information to be stored in a database is that of the template generated when making a chart graph. This information will have a unique identifier number so it can be easily retrieved later. The user is the one that will decide to both store the template in the database and retrieve it. Note that if the template was not able to be used for generating the chart graph, then it won't be able to be stored in the database.

3.5 Design constraints

At least 90% of the product's source code should be written in Java programming language using Oracle's code conventions which can be retrieved at the following address:
http://www.oracle.com/technetwork/java/codeconv-138413.html

The user interface is to be made using Java's primary GUI widget toolkit "Swing", the graph charts using the library JFreeChart 1.0.16 availiable at http://sourceforge.net/projects/jfreechart/files/ and Opencsv 2.3 found at http://sourceforge.net/projects/opencsv/ which can be very helpful to manipulate data in 'csv' format. Finally, the database should be implemented using XML.

3.6 Software system attributes

This section lists a series of attributes that PlasmaGraph should have and therefore they must be objectively verified.

3.6.1 Reliability

The product is to be submitted to a series of tests before a release version is delivered. These tests will evaluate how much does the system takes to generate a chart graph and how does it handles each scenario described in section 3.2.

3.6.2 Maintainability

This product will use only java standard libraries with the exeptions of JFreeChart 1.0.16 and Opencsv 2.3. A library's source code can't be altered so the original documentation of each one can stay as true to its software as possible.

PlasmaGraph uses git which is a distributed version control system that helps to keep track of code revisions and allows many developers to work on the same project.

3.6.3 Portability

Because this product runs in the Java Virtual Machine so its portability is determined by this piece of software. To know if your system supports Java you can follow the link below:
www.java.com/en/download/help/sysreq.xml

3.7 Organizing the specific requirements

This section contains a detail description of each requirement of the system and organizes them in a manner optimal for understanding.

3.7.1 Objects

The objects contained in the product are:

- User Interface
- Chart graph tool
- Tabular data manipulation tool
- Templates
- Database

3.7.2 Features

This product can perform the following tasks:

- Fix common errors in tabular data sets.
- Create and save templates for generating chart graphs.
- Save chart graphs as image files in the user's file system.

3.8 Additional comments