Requirements 2D-GC visualization

Baksteen, Sarah
S3145034

Brink, Wouter
S3862348

Gros, Oane
S2972778

Jager, Maarten de
S2957906

Jong, Michiel de
S2550768

Strik, Oliver
S3100693

university of groningen

Client
Rohrbach, Léon
Figueiredo, Monique

TA
Argyrousis, George

Software Engineering
University of Groningen
March 31, 2019

Version 3.0

1 Introduction

This document elaborates the requirements for the project 2D-GC visualization. This project is commisioned by the Chemical Engineering department of the University of Groningen and is part of the course *Software Engineering*.

Problem

2-dimensional Gas Chromatography (2D-GC) is a chemical technique to accurately identify and quantify small traces of compounds in samples. This analytical technique is used in Groningen to aid in the search for new biofuels. For this research, accurate and interactive software for 3D-visualization and analysis is necessary.

The software which is currently being used by the Chemical Engineering department is limited in functionalities. This means that there is information which cannot be accessed yet with the current technology.

Solution

We aim to build a desktop app to visualise 2D-GC data in one, two and three dimensional spaces, and try to work in comparative analysis, where the same analysis can be run smoothly on multiple samples.

Terminology

This subsection defines important terms commonly used in this doucment.

2D-GC
Two dimensional gas chromatography (see GC).

GC
GC is an abbreviation for gas chromatography. This is a chemical technique to accurately identify and quantify small traces of compounds in samples.

Integration
Calculating the total amount of area under a specific region in the given data.

2 Stakeholders

This section gives an overview of all stakeholders involved in this project.

There are two stakeholders in this project: Chemical analysts of the RUG Chemical Engineering department and the developers.

The chemical analysts are the clients of the project. These are the users of the program. The program is going to be a tool for their research. The chemical analysts determine the requirements of the project.

The second type of stakeholder are the developers. These are students following the course *Software Engineering*. The developers are responsible for writing the software and making sure that the desired functionalities are being implemented.

3 User Stories

- As a chemical analyst, I want to find the quantity of a specific compound in multiple samples, and compare these quantities.

- As a chemical analyst, I want to quickly view the outcome of a diagnostic test by visually assaying the 3D graph.
- As a chemical analyst publishing research, I want to have a way to show the data in a visually appealing manner, so that a saved image would be appropriate for publication.
- As a chemical analyst, I want to be able to be able to find the quantity of a larger chemical group in my sample by freeform selection of groups of peaks.
- As a developer, I want to be able to extend the program to work with GCxGC-MS, where an extra identification dimension is added on top of the 2D-GC data.

4 Funcitonal requirements

Critical Functional Requirements

- Show 2D-GC output .txt files as interactive 3-dimensional graphs, where the user can use intuitive controls to visually inspect and interact with the graph. These intuitive controls will include dragging to rotate the graph and scrolling to zoom in or out or scale the graph.
- Show 2D-GC output .txt files as 2-dimensional graphs, with similar controls: scrolling to zoom in, and dragging to move across the zoomed in graph.
- It should be possible to integrate a specific area in the data. The user should be able to draw an area over the 2D graph space and receive the calculated integral for that space.

Important Functional Requirements

- Be able to save the image with any attached user inputs such as integration or labels as an image file, or as a specific format from which the graph and any user inputs would be able to be reloaded.
- Be able to select a freeform space to integrate over, with supplied extra integration parameters, such as how high to clip data.
- Changing the colors of the visualizations between multiple presets, and optionally change the midpoint of the color range, so more color resolution is brought to different levels of the data.
- Be able to save peak data in a database, with any attached information, such as which sample it was from, integration value, labels etc.. The user should be able to retrieve previously saved information in a look-up table while looking at a new sample.
- Automatic peak finding

Useful Functional Requirements

- Pipelining for comparisons between samples, where a single selection region or other user inputs could be preset into the program and done on multiple data files simultaneously.
- Compound recognition/matching with the internal database, where older labelled data (which was identified with mass spectrometry) would automatically be recognized if it's already in the internal database, and new data shows a peak at the same position.
- Interpolation over peaks to make them look smoother and prettier.

Won't Do

- Linking to an outside Mass-Spectrometry (MS) library. This is something which would be useful, as the client may switch from an FID detector to an MS detector in the near future. However, this would add a layer of complexity, as MS data adds an extra dimension to the data. We think it best to focus on a stable program for FID visualization, as this can also be used for GCxGC-MS data.

5 Non-functional requirements

- **Extensibility:** this is essential, as the client is preparing to switch to a different detection system, which alters the dimensionality of the data. We have to make sure that this extension would be easily incorporated in the program.
- **Technology:** the program needs to be able to run on older versions of operating systems, as the measurement software itself is run on windows XP.
- **Maintainability:** the software has to be designed in such a way that the program is easily maintainable. After the course *Software Engineering* has ended, the developers will not provide further maintenance to the software.
- **Scalability:** The amount of information in the database will most likely grow with time. Therefore, the database has to be scalable to a certain extend.

6 Meeting Log

Date	**Meeting log**
25/02/2019	Talked about and discussed the requirements document.

7 Change Log

Date	**Author**	**Change log**
15/02/2019	Oane Gros	Initial version
25/02/2019	All	Introduction and MoSCoW
04/03/2019	Wouter Brink, Oliver Strik	Added meeting and change logs
04/03/2019	Wouter Brink	Added title page
08/03/2019	Oane Gros	Added basic user stories and non-functional requirements
10/03/2019	Oane Gros	Extended requirements to include more detailed description, and some alterations based on the first physical meeting with the client
31/03/2019	Wouter Brink	Added Terminology and Stakeholder sections.
31/03/2019	Wouter Brink	Restructured the document.
31/03/2019	Wouter Brink	Added Maintainability and Scalability to the non functional requirements.