Mass Spectrometer Interface

A Desktop Application for Reading Instrument Data

Cousins Photosynthesis Lab in the School of Biological Sciences at WSU



**Team Linnaea Borealis**

****

Kyler Kupp, Erik Holtrop

**Note**: Recall that this writing assignment says:

Length = 3 pages text + appendixes as needed.

Some materials do not count towards this 3 page minimum. These excluded parts include:  
 Cover page  
 Table of contents  
 Pictures  
 Images  
 Use-case UML diagrams

Posted as a single self‐contained file (no links to outside resources.)

Posted as a PDF file to both Blackboard and your team’s Assignments GitLab repository

Typed single‐spaced.

Typed with black text.

Typed with #11 font size.

Typed using Arial font.

Typed with one inch margins on sides, top and bottom.

**(Please erase this page in your final document.)**

**TABLE OF CONTENTS**

[**I.**](#_gjdgxs) **Introduction 4**

[**II.**](#_30j0zll) **System Requirements Specification 4**

[II.1.](#_1fob9te) Use Cases 4

[II.2.](#_3znysh7) Functional Requirements 4

[*II.2.1.*](#_2et92p0) *[The name of the module/component/part] 4*

[*II.2.2.*](#_tyjcwt) *[The name of the next module/component/part] 5*

[II.3.](#_3dy6vkm) Non-Functional Requirements 5

[**III.**](#_1t3h5sf) **System Evolution 5**

[**IV.**](#_4d34og8) **Glossary 5**

[**V.**](#_2s8eyo1) **References 5**

# Introduction

When plants breathe, they take carbon dioxide (CO2) out of the air and replace it with oxygen (O2). Determining what affects plants’ respiration rate, or their breathing rate, is incredibly valuable data. These factors point backwards in time, reflecting causes for evolutionary trends, and forwards in time, providing opportunities to improve agriculture. We can use a mass spectrometer to measure this breathing rate. The mass spectrometer measures the amount of CO2 and O2 in the air, and so the respiration rate is the change in each of those.

The Cousins Photosynthesis Lab in the School of Biological Sciences at Washington State University uses one of these mass spectrometers. These instruments are complicated devices, requiring complex calculations for calibration. The lab uses proprietary software from the mass spectrometer’s manufacturer, but that software outputs massive amounts of data over the course of a multi-hour lab, most of which isn’t needed. This problem has been partially solved with the creation of a Python desktop application, but this application is not perfect. Our task is to improve this application. This application currently faces small bugs, and only works for one instrument. The application is also in process of a UI upgrade. There’s also a few non-spectrometer instruments in the lab that are provide similar data, that would be easier to use if their data-streams were combined.

# System Requirements Specification

In this section you will describe the features, functions and other specifications that are requirements for your product. You will also specify the client/stakeholder need(s) that requirement maps to. If you find a stakeholder that your first assignment did not identify, feel free to add them here as required.

Please refer to Section 4.4 in the CptS 322 & CptS 422 textbooks like “Object-Oriented Software Engineering” to refresh your knowledge on software requirements.

## Use Cases

If applicable, provide some major use-cases that illustrate scenarios for using your product[[1]](#footnote-1). Use cases tell a story about how an end user interacts with the system under a specific set of circumstances. You may illustrate the use-cases with UML diagrams.

For each use case, identify the related requirements (you may directly refer to the requirements listed in Section II.2).

(Note that the diagrams will not be counted for the 3 pages text length specified for this document.)

## Functional Requirements

List the functional requirements in this section[[2]](#footnote-2).

Include a subsection for each main part/module of your product and list the requirements for the module in that subsection. (*Please note that we are not considering any design issues yet. Each module (subsection) refers to a major part/functionality of the product, not to sub-section in the architecture. This classification of requirements is intended to improve the readability of the document.*)  
  
Generally, functional requirements are expressed in the form "system must do <requirement>”

Briefly describe each requirement and specify the client/stakeholder need(s) that requirement maps to. Each requirement should appear in ONLY ONE sub-section of the document.

Here is an example template for requirement specification: (*the requirement template is formatted with blue for readability. Please remove the color formatting in your document*.)

### [The name of the module/component/part]

List your requirements for this project module here. Each listed requirement MAY INCLUDE the following items:

**[Enter a Concise Requirement Name]:** [provide a concise description, in clear and easily understandable language to specify the requirement]

**Source**: [specify who/what originated this requirement, specify the client/stakeholder need(s) that requirement maps to]

**Priority**: specify the priority of this requirement, you may use the following scale to specify priority of the requirements:

Priority Level 0: Essential and required functionality

Priority Level 1: Desirable functionality

Priority Level 2: Extra features or stretch goals

Repeat the above for each requirement.

### [The name of the next module/component/part]

List your requirements for the next project module here based on the requirements template described above. Include a subsection for each part.

## Non-Functional Requirements

**Modern Visuals:**

The system shall look like a modern 2020’s program. PyQt5’s baseline UI elements meets this standard

**Python For Extendibility:**

The system shall use Python since it is one of the most common languages among Biology students, including those that use the Cousins Photosynthesis Lab. This will enable extension and maintenance.

**Documentation For Extendibility:**

The system shall have documentation for code that’s thorough enough to allow college senior skill level software engineers to extend and maintain it.

**Documentation For Non-Technical Users:**

The system shall have documentation to enable non-technical users to make full use of it. For example, an explanation of how to create executables of each module.

**Accuracy:**

The system shall accurately calculate and portray data. The instruments have their own levels of accuracy, so maintenance of significant figures should be enough.

**Compatibility:**

The system shall use the established data format, which is versatile enough to work with other devices.

**Non-Destructive:**

The system shall not overwrite or otherwise destroy any data.

# System Evolution

The Cousins Lab’s relationship with WSU places it in a position to enlist Computer Science students to work on this codebase each year as part of their capstone. We see this in the project’s history, having two teams as previous maintainers. The sponsor liaison, Dr. Cousins, has voiced an intention to have software created for many of the instruments in the lab. With these considerations, we should make our software compatible with different machines if possible, and make it maintainable and extendable by software engineers with the skill and education level of a college senior.

We can see what this looks like from the issues and opportunities presented by previous teams. For instance, the modules written from the lab’s primary Mass Spectrometer, may be usable for the second one, provided we write software to reformat its data stream. In another case, some of the existing code lacks basic files to enable maintenance, like a requirements.txt for relevant Python libraries or the context files for executable creation.

# Glossary

Define technical terms used in the document.

# References

Cite your references here.

For the papers you cite give the authors, the title of the article, the journal name, journal volume number, date of publication and inclusive page numbers. Giving only the URL for the journal is not appropriate.

For the websites, give the title, author (if applicable), date accessed, and the website URL.

Please use either IEEE or Chicago format for your references. Keep in mind that Google Scholar or BibTeX can help you easily format your citations for periodicals and journals.

1. https://en.wikipedia.org/wiki/Use\_case [↑](#footnote-ref-1)
2. https://en.wikipedia.org/wiki/Functional\_requirement [↑](#footnote-ref-2)