

Jack of All Spades

Eric Jestel, Alex Ayer, Bryce Roy, Chris Letourneau, Conall Gouveia

System Requirements Specification

Chemistry Instrumentation: Control, Data Acquisition and Offline Analysis

For Dr. Mitchell Bruce with the Chemistry Department at the University of Maine, Orono

11/12/2025

V1.1

Table of Contents

1. Introduction	1
1.1. Purpose of This Document	1
1.2. References	1
1.3. Purpose of the Product	1
1.4. Product Scope	2
2. Functional Requirements	3
Use Case Tests	11
3. Non-Functional Requirements	22
4. User Interface	26
5. Deliverables	27
6. Open Issues	28
Appendix A - Agreement Between Customer and Contractor	29
Appendix B – Team Review Sign-off	30
Appendix C – Document Contributions	31

1. Introduction

This document presents the system requirements for Chemistry Instrumentation: Control, Data Acquisition and Offline Analysis. This project is being undertaken by a team of seniors at the University of Maine in order to fulfill the capstone requirements of a Computer Science degree. This is being undertaken in collaboration with the chemistry department at the University of Maine.

1.1. Purpose of This Document

Within this document, the plan for the project is laid out. The requirements are laid out in order to clarify the exact parameters of the project, as well as defining the conditions required for success. This document was written by the team and was reviewed by the client to ensure that all parties have a common understanding. Thus, future materials can reference this document when working on future parts of the project. Contained within is a description of the project, functional and non-functional requirements, what deliverables are expected, and any decisions yet to be made.

1.2. References

- *UML Distilled*, by Martin Fowler.

1.3. Purpose of the Product

The UMaine chemistry department recently purchased a UV-vis Spectrometer, as well as an IR Spectrometer. These instruments will be used in introductory chemistry labs by students. This means that the instruments must be able to process many samples in each 3 hour lab period, as each student must prepare and analyze their own sample. For a given student to operate and analyze the data from one of these instruments can take upwards of fifteen minutes, meaning that additional instruments would need to be purchased to enable all students to complete the lab. However, a majority of that time is spent on analysis, so allowing students to store the data, then analyze it later on their own will greatly reduce the time needed.

In order to split up the operation of the machine and the analysis, a custom software will present a student-friendly user environment. This environment will allow a student to operate the machine using preset settings, look at their measurements, then upload the data to the InterChemNet server for later analysis. In this way, students will be able to quickly process samples in the lab period while having time to analyze their samples without time constraints.

1.4. Product Scope

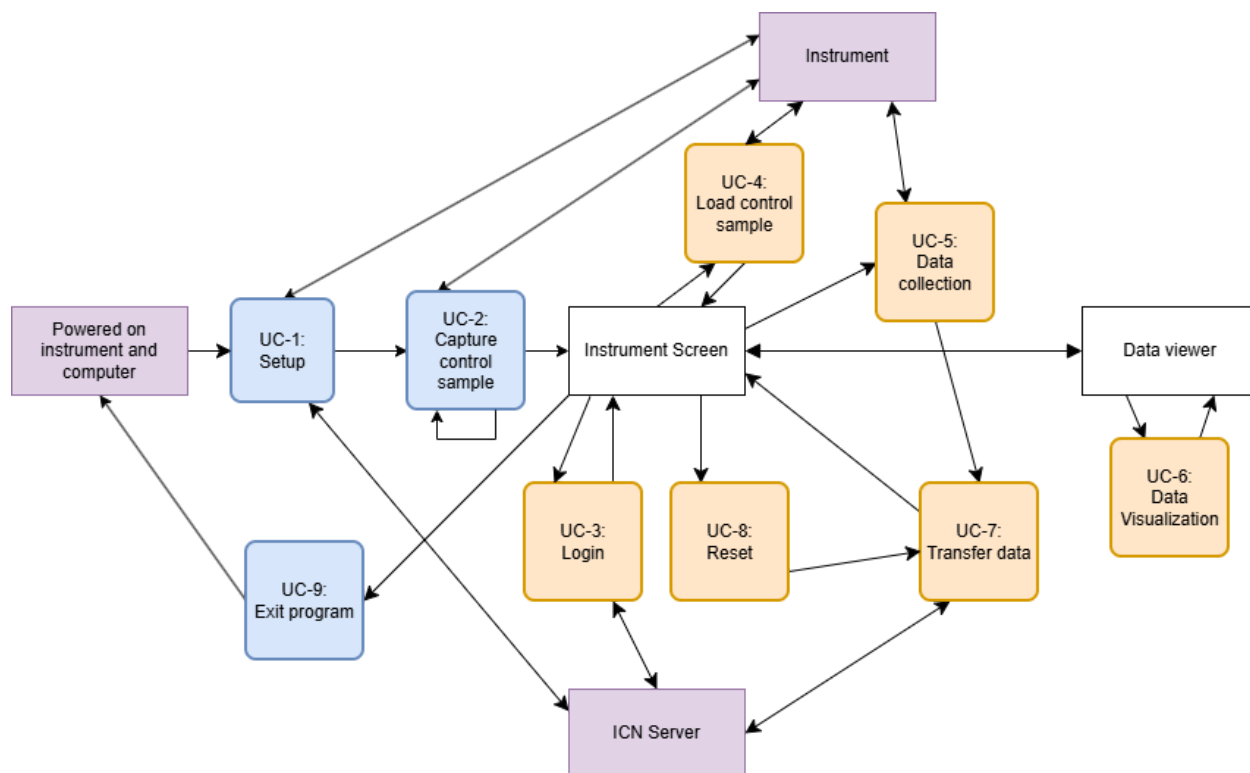


Figure 1. State Diagram of the System. In this figure, the use cases are labeled as “UC-X” and colored depending on the primary user. In blue are the tasks that the lab manager will do. In orange are the use cases that students will engage with. Additionally, 3 external tools or states are represented in purple. There is 1 primary screen associated with the system, the instrument screen, which has the data viewer as a component of it. Both are shown in white.

Here, the system diagram and primary use cases are displayed. The lab manager tasks will be performed once per day or per lab session. The student tasks make up the primary cycle of the machine, representing tasks that each student will perform during a lab period. The system consists of 3 parts: the computer, the instrument, and the InterChemNet (ICN) server. Our program, running on the computer, will interact directly with the instrument to perform measurements and with the ICN server to store the measurements taken. The instrument screen is the primary means through which students will interact with the program. First, they will login, then capture data, view the data through the data viewer section, and finally reset the screen once complete.

2. Functional Requirements

This section details the functional requirements, presented as use cases. Each use case represents one process involved with the system, such as login, or capturing a measurement. Presented below are the presented use cases for this system, along with a set of tests that will be used to validate the success of each case.

Number	UC - 001	
Name	Setup	
Summary	The TA/professor (“Lab manager”) must start the software and run our program. The lab manager must connect to the machine in the software. The ICN server is pinged and the connection is verified.	
Priority	5	
Preconditions	The machine must be physically connected to the computer. The machine must be powered on. The program is loaded on the computer’s hard drive.	
Postconditions	The program will be running and on UC-002. The machine will be accessible to the program. The connection to the ICN server is verified.	
Primary Actor	Lab manager	
Secondary Actors	Machine, ICN server	
Trigger	The user clicks on the program executable	
Main Scenario	Step	Action
	1	Program is started
	2	Program checks for connection to instrument.
	3	Program pings ICN server
	4	The program goes to the UC-002
Extensions	Step	Branching Action
	2a	Instrument is not connected : Display “Instrument not connected. Please connect instrument”
	3a	ICN server doesn’t respond : Display “ICN server failed to respond. Retry or Continue anyways?”
Open Issues	None	

Number	UC - 002	
Name	Capture control sample	
Summary	The TA/professor (“Lab manager”) must capture a “blank” (control sample). They must repeat this process until satisfied with the result. The software must store the blank.	
Priority	3	
Preconditions	Setup is complete. A blank sample is loaded into the machine.	
Postconditions	A blank has been captured. The machine is on the login screen	
Primary Actor	Lab manager	
Secondary Actors	Machine	
Trigger	TA clicks “Capture blank”	
Main Scenario	Step	Action
	1	TA clicks capture blank
	2	“Check that the Blank Sample is inserted” pops up. TA clicks “capture blank” button
	3	Program instructs instrument to capture a reading
	4	Program displays the reading and “keep blank” or “capture again”
	5	If TA clicks “keep blank”, store the reading in a local file
Extensions	6	If TA clicks “capture again”, go to 2
	Step	Branching Action
	2a	Instrument error: Display “Instrument error” and whatever error is reported
Open Issues	None	

Number	UC - 003	
Name	Login	
Summary	The student has to enter their ICN username. The information must be validated.	
Priority	3	
Preconditions	The program must be on the instrument screen. All other logins must be cleared. The computer must be connected to the internet. ICN server must be running	
Postconditions	The student is logged in. Instrument screen is displayed.	
Primary Actor	Student	
Secondary Actors	ICN server	
Trigger	User enters login information and clicks “login”	
Main Scenario	Step	Action
	1	User enters ICN username into login box
	2	User clicks “login”
	3	The system sends credentials to the ICN server, then receives validation.
	4	The system greys out the username box and highlights UC-008 button (reset).
	4	The system displays “Successfully login”
Extensions	Step	Branching Action
	1a	If another user is logged in : Username field is greyed out and can’t be edited. Highlight UC-008 button
	3a	If username isn’t recognized: Display “username not recognized”. Clear username field
	3b	If ICN server does not respond or times out: Display “ICN server not responding. Check internet connection”
Open Issues	Should the machine be usable with a “fake” account, like “demouser”, for demos and such? Should the machine take multiple usernames as inputs?	

Number	UC - 004	
Name	Load control sample	
Summary	The student must either select a control sample or capture a new blank	
Priority	5	
Preconditions	The user is logged in to their ICN account. System is on the instrument page.	
Postconditions	The user has a control sample selected for that session. System is on the instrument page.	
Primary Actor	Student	
Secondary Actors	Instrument	
Trigger	User clicks “Advanced options”	
Main Scenario	Step	Action
	1	User clicks “Advanced options”, then “Select blank manually”
	2	System displays 3 buttons: “Use active blank”, “Select old blank from file”, and “Capture new blank”
	3	User clicks “Use active blank”: The blank captured by TA for that session is selected. System goes to 6
	4	User clicks “Capture new blank”: Execute UC-002, but do not overwrite session blank. System goes to 6
	5	User clicks “Select old blank from file”: File selector pops up. User selects a valid spectra file. System goes to 6
	6	System displays “blank selected” and returns to instrument page. Until reset or “select blank” clicked again, that blank will be utilized automatically.
Extensions	Step	Branching Action
	4a	If the instrument reports an error: Display “Instrument error” and then the reported error
Open Issues	None	

Number	UC - 005	
Name	Data collection	
Summary	The student must capture the data. The data must be stored locally.	
Priority	5	
Preconditions	The student must be on the instrument page. The student must be logged in.	
Postconditions	The student will be on the instrument page. Data will be displayed in the data viewer. UC-007 will execute	
Primary Actor	Student	
Secondary Actors	Instrument	
Trigger	Student clicks “Capture sample”	
Main Scenario	Step	Action
	1	User clicks “capture sample”
	2	System displays “Make sure sample is inserted” popup and “Capture” button
	3	System interacts with instrument to record data
	4	The system writes the data into a local file in sample folder, with details about the sample and the user in the file name
	5	System displays “Sample captured”
	6	The system displays the data in the data viewer window.
	7	The system executes UC-007
Extensions	Step	Branching Action
	4a	If the instrument reports an error: Display “Instrument error” and then the reported error
Open Issues	File name formatting? How should data be displayed?	

Number	UC - 006	
Name	Visualize data	
Summary	The student can view the data in a graph. The student can adjust visualization options of the data.	
Priority	5	
Preconditions	The user is on the instrument page. The user has captured at least 1 sample.	
Postconditions	The user remains on the instrument page. The data is formatted according to the student specifications.	
Primary Actor	Student	
Secondary Actors	None	
Trigger	The student is on the instrument page with data captured	
Main Scenario	Step	Action
	1	Data is shown in a graph. Student selects one of the visualization options (set bounds, show/hide sample, reset).
	2	The program adjusts the visualization according to what was selected and returns to 1.
	3	Student is done adjusting the visualization. Student enters into other use case
Extensions	Step	Branching Action
		None
Open Issues	What visualization options are available?	

Number	UC - 007	
Name	Transfer data	
Summary	All data files that are stored and not yet transferred are sent to the ICN server, then tagged as sent.	
Priority	4	
Preconditions	The student has just captured a sample or has executed UC-008.	
Postconditions	The student will be on the instrument page. ICN will have received the data	
Primary Actor	Student	
Secondary Actors	None	
Trigger	User clicks “reset” User completes UC-005	
Main Scenario	Step	Action
	1	User clicks “reset” or completes UC-005
	2	System reads all files in sample folder. For each file without a “sent” tag, perform steps 3 and 4
	3	Send the file to the ICN server, based on the username in the file name.
	4	Add “sent” tag to the file.
	5	Display “All files sent”
Extensions	Step	Branching Action
	3	If ICN fails to acknowledge file receipt, do not perform step 4.
	5	If any files failed to send, display “ICN did not receive” and the file names that did not send
Open Issues	None	

Number	UC - 008	
Name	Reset	
Summary	The student has captured all their data and is done with the machine. The system clears all data, sends anything leftover, and clears the password field	
Priority	5	
Preconditions	The user is on the instrument page. The user is logged in.	
Postconditions	The user is on the instrument page. Nobody is logged in.	
Primary Actor	Student	
Secondary Actors	ICN server	
Trigger	The student clicks “reset” from the instrument page.	
Main Scenario	Step	Action
	1	Confirmation popup and “continue” “go back” buttons appear
	2	Clear all local variables and send “reset” commands to all components
	3	Call UC-007
Extensions	Step	Branching Action
	1a	Student clicks “go back”: Return to instrument page
Open Issues		

Number	UC - 009	
Name	Exit Program	
Summary	The program is done being used for the lab period. Any connections to the machine are terminated, all remaining files are flagged, and any old files are deleted.	
Priority	3	
Preconditions	The machine is on the instrument screen. The “reset” button has been pressed.	
Postconditions	The software is closed.	
Primary Actor	Lab manager	
Secondary Actors	None	
Trigger	User clicks “exit program”	
Main Scenario	Step	Action
	1	System checks for any files not tagged as “sent” and displays a list.
	2	System deletes any files timestamped more than 2 weeks ago
	3	System terminates any connections to the instrument
	4	System closes all components.
Extensions	Step	Branching Action
Open Issues	How long should files be saved for?	

Use Case Tests

Number	UCT - 001a	
Name	Setup Test	
Related to	UC - 001	
Description	Test if the software launches properly with the instrument attached	
Priority	4	
Test	Step	
	1	Initiate software on a Windows 11 computer with the instrument connected
	2	Verify that software has no critical initialization errors
	3	Verify that software connects to lab machine

Number	UCT - 001b	
Name	No Instrument Setup Test	
Related to	UC - 001	
Description	Test if the software displays an error while launching without the instrument attached	
Priority	2	
Test	Step	
	1	Initiate software on a Windows 11 computer without the instrument connected
	2	Verify that software displays an error, that is could not connect to the machine

Number	UCT - 002a	
Name	Capture Blank Test	
Related to	UC - 002, UC - 004, UC - 005	
Description	Test if a user can collect a background sample	
Priority	5	
Test	Step	
	1	Initiate a Blank Capture
	2	Verify a reminder window pops up
	3	Click "Capture Blank" button
	4	Verify that a sample was produced
	5	Click "Save Sample"
	6	Initiate a Sample Capture with the same blank sample inserted into the machine
Open Issues	7	Verify that absorption does not deviate from 0 by more than 0.005.
	Exact values might need to be changed with some testing	

Number	UCT - 002b	
Name	Capture Blank Capture Again Test	
Related to	UC - 002, UC - 004, UC - 005	
Description	Test if a user can collect a background sample	
Priority	5	
Test	Step	
	1	Initiate a Blank Capture
	2	Verify a reminder window pops up
	3	Click “Capture Blank” button
	4	Verify that a sample was produced
	5	Click “Capture Again”
	6	GOTO: UCT - 002a Step 2

Number	UCT - 002c	
Name	Disconnected Instrument Test	
Related to	UC - 002, UC - 004, UC - 005	
Description	Tests that an error is displayed if the user tries to collect a sample when the instrument is disconnected	
Priority	2	
Test	Step	
	1	Disconnect instrument from computer
	2	Initiate a sample capture / blank capture
	3	Verify a reminder window pops up
	4	Click “Capture” button
	5	Verify that an error message pops up

Number	UCT - 003a	
Name	Valid Login test	
Related to	UC - 003	
Description	Tests if a user can log in with a correct username	
Priority	3	
Test	Step	
	1	Input valid username into the login box
	2	Verify that the user is logged in

Number	UCT - 003b	
Name	Invalid Login test	
Related to	UC - 003	
Description	Tests that a user can not log into an account with an incorrect username	
Priority	5	
Test	Step	
	1	Input invalid username into the login box
	2	Verify that a user is not logged in

Number	UCT - 004a	
Name	Capture New Blank Test	
Related to	UC - 004, UC - 002, UC - 005	
Description	Tests that a user can capture a new blank from the instrument screen	
Priority	2	
Test	Step	
	1	Click "Select blank"
	2	Click "Capture new blank"
	3	Verify a reminder window pops up
	4	Click "Capture Blank" button
	5	Verify that a sample was produced
	6	Click "Save Sample"
	7	Initiate a Sample Capture with the same blank sample inserted into the machine
	8	Verify that absorption does not deviate from 0 by more than 0.005.

Number	UCT - 004b	
Name	Use Old Blank Test	
Related to	UC - 004, UC - 002, UC - 005	
Description	Tests that a user can load an old blank from a file	
Priority	2	
Test	Step	
	1	Capture a Blank with a Blank Sample
	2	Save Blank to a file
	3	Capture a second Blank with a Non-Blank Sample
	4	Save Non-Blank as Session Blank
	5	Insert Blank Sample into the machine
	6	Take sample of Blank using Session Blank
	7	Verify that absorption deviates from 0 by at least 0.005 at least once during capture
	8	Click "Select blank"
	9	Click "Select old blank"
	10	Select original Blank Sample File produced
	11	Take Sample of Blank using Old Blank file
	12	Verify that absorption does not deviate from 0 by more than 0.005

Number	UCT - 004c	
Name	Capture New Blank Default Test	
Related to	UC - 004, UC - 002, UC - 005	
Description	Tests that when user captures a new blank from the instrument screen it does not override the default session blank	
Priority	2	
Test	Step	
	1	Capture a Blank with a Blank Sample
	2	Save Blank as Session Blank
	3	Click "Select blank"
	4	Click "Capture new blank"
	5	Capture Blank with Non-Blank Sample
	6	Replace sample with the original Blank Sample
	7	Capture a Sample
	8	Verify that absorption deviates from 0 by at least 0.005 at least once during capture
	9	Log out
	10	Log back in
	11	Capture a Sample without touching any settings
	12	Verify that absorption does not deviate from 0 by more than 0.005

Number	UCT - 004d	
Name	Use Old Blank Default Test	
Related to	UC - 004, UC - 002, UC - 005	
Description	Tests that when user loads an old blank from a file it does not override the default session blank	
Priority	2	
Test	Step	
	1	Capture a Blank with a Non-Blank Sample
	2	Save Non-Blank to a file
	3	Capture a Blank with a Blank Sample
	4	Save Blank as Session Blank
	5	Click "Select blank"
	6	Click "Select old blank"
	7	Select file for the Non-Blank Sample
	8	Capture a Sample with the Blank Sample
	9	Verify that absorption deviates from 0 by at least 0.005 at least once during capture
	10	Log out
	11	Log back in
	12	Capture a Sample without touching any settings
	13	Verify that absorption does not deviate from 0 by more than 0.005

Number	UCT - 004e	
Name	Use Session Blank Test	
Related to	UC - 004, UC - 002, UC - 005	
Description	Tests that a user can utilize the blank that was captured for that session after switching to a different blank.	
Priority	2	
Test	Step	
	1	Capture a Blank with a Non-Blank Sample
	2	Save Non-Blank to a file
	3	Capture a Blank with a Blank Sample
	4	Save Blank as Session Blank
	5	Click "Select blank"
	6	Click "Capture blank"
	7	Capture Blank with a Non-Blank Sample
	8	Click "Select blank"
	9	Click "Active Blank"
	10	Capture a Sample with a Blank Sample
	11	Verify that absorption does not deviate from 0 by more than 0.005.
	12	Click "Select blank"
	13	Click "Select old blank"
	14	Click "Select blank"
	15	Click "Active Blank"
	16	Capture a Sample with a Blank Sample
	17	Verify that absorption does not deviate from 0 by more than 0.005.

Number	UCT - 005	
Name	Capture Sample Test	
Related to	UC - 005	
Description	Test if a user can collect a sample	
Priority	5	
Test	Step	
	1	Initiate a Sample Capture
	2	Verify a reminder window pops up
	3	Click "Capture" button
	4	Verify that a sample was produced

Number	UCT - 006a	
Name	Data Visualization Test	
Related to	UC - 006, UC - 005	
Description	Test if the data visualization shows data that has been collected by the user	
Priority	4	
Test	Step	
	1	Login to the system
	2	Verify that the data visualization is empty
	3	Collect a Sample
	4	Verify that the data visualization show one sample
	5	Collect another sample
	6	Verify that the data visualization show 2 different samples
	7	Logout
	8	Verify that the data visualization is empty

Number	UCT - 006b	
Name	Data Visualization Bounds Test	
Related to	UC - 006, UC - 005	
Description	Test if the data visualization's set bounds option changes the range on the graph.	
Priority	3	
Test	Step	
	1	Take a sample
	2	Click set bounds
	3	Input values inside the normal bounds
	4	Verify that the data visualization's range has changed according to the inputted bounds

Number	UCT - 006c	
Name	Data Visualization Out of Bounds Test	
Related to	UC - 006, UC - 005	
Description	Test if the data visualization's set bounds option checks if the imputed bounds are outside the graphs max values.	
Priority	2	
Test	Step	
	1	Take a sample
	2	Click set bounds
	3	Input values inside the normal bounds
	4	Verify that the data visualization's range has changed according to the inputted bounds
	5	Input a lower bound below the minimum bound.
	6	Verify that the lower bound was set to the minimum
	7	Verify that the upper bound does not change
	8	Input values inside normal bounds again
	9	Input an upper bound above the maximum bound
	10	Verify that the upper bound was set to the maximum
	11	Verify that the lower bound does not change

Number	UCT - 006d	
Name	Data Visualization Minimum Width Test	
Related to	UC - 006, UC - 005	
Description	Test if the data visualization's set bounds option enforces a minimum width	
Priority	2	
Test	Step	
	1	Take a sample
	2	Click set bounds
	3	Input values for the upper and lower bounds as the same number
	4	Verify that the lower bound is set to the inputted value
	5	Verify that the upper bound is set to the lower bound plus the minimum width.

Number	UCT - 006e	
Name	Data Visualization Hide Sample Test	
Related to	UC - 006, UC - 005	
Description	Test if the data visualization's hide sample option toggles a samples visibility	
Priority	3	
Test	Step	
	1	Take 3 samples
	2	Verify that all 3 samples are visualized
	3	Click hide sample 2
	4	Verify that sample 2 is hidden
	5	Verify that sample 1 and 3 are visualized
	6	Click hide sample 3
	7	Verify that sample 2 and 3 are hidden
	8	Verify that sample 1 is still visualized
	9	Click show sample 2
	10	Verify that sample 1 and 2 are visualized
	11	Verify that sample 3 is still hidden
	12	Verify that sample 2's data remains unchanged from step 2

Number	UCT - 006f	
Name	Data Visualization Reset Test	
Related to	UC - 006, UC - 005	
Description	Test if the data visualization's reset option resets all values to their defaults	
Priority	3	
Test	Step	
	1	Take 2 samples
	2	Click set bounds
	3	Input values inside the normal bounds
	4	Verify that the data visualization's range has changed according to the inputted bounds
	5	Click hide sample 1
	6	Verify that sample 2 is visualized
	7	Verify that sample 1 is not visualized
	8	Click reset
	9	Verify that sample 1 and 2 are visualized
	10	Verify that the upper and lower bounds are set to their defaults

Number	UCT - 007a	
Name	Save Sample Test	
Related to	UC - 007, UC - 005	
Description	Test if a user can save a sample they just took	
Priority	4	
Test	Step	
	1	Capture a sample
	2	Verify that a dialog box appears
	3	Click "Save Sample"
	4	Verify that the sample remains in memory

Number	UCT - 007b	
Name	Discard Sample Test	
Related to	UC - 007, UC - 005	
Description	Test if a user can discard a sample they just took	
Priority	3	
Test	Step	
	1	Capture a sample
	2	Verify that a dialog box appears
	3	Click "Discard Sample"
	4	Verify that the sample does not remain in memory

Number	UCT - 008a	
Name	Send Sample Test	
Related to	UC - 008, UC - 007, UC - 005	
Description	Tests if a user can send a sample's data to ICN	
Priority	5	
Test	Step	
	1	Capture a sample
	2	Click "Continue"
	3	Verify that
	4	Verify that a confirmation window appears
	5	Click "confirm" button
	6	Verify that the data was sent

Number	UCT - 008b	
Name	Send Sample after Reset Test	
Related to	UC - 008, UC - 007, UC - 005	
Description	Tests if a user can abort the sending process once started	
Priority	3	
Test	Step	
	1	Capture a sample
	2	Save the sample
	3	Click “Send data and Exit”
	4	Verify that a confirmation window appears
	5	Click “Go Back” button
	6	Verify that the data was not sent

Number	UCT - 008c	
Name	Failed Send Sample Test	
Related to	UC - 008, UC - 007, UC - 005	
Description	Tests that if a ICN is unavailable, an error is displayed	
Priority	1	
Test	Step	
	1	Block ICN’s IP Address
	2	Capture a sample
	3	Save the sample
	4	Click “Send data and Exit”
	5	Verify that a confirmation window appears
	6	Click “confirm” button
	7	Verify that an error appears

Number	UCT - 009a	
Name	Logout Test	
Related to	UC - 009	
Description	Test if a user can logout of the system (button click)	
Priority	3	
Test	Step	
	1	Click “Logout”
	2	Click “Log out now” button
	3	Verify that the user has been logged out of the software

Number	UCT - 009b	
Name	Logout Test	
Related to	UC - 009	
Description	Test if a user can logout of the system (automatic logout)	
Priority	3	
Test	Step	
	1	Click "Logout"
	2	Click nothing until timeout finishes
	3	Verify that the user has been logged out of the software

Number	UCT - 009	
Name	Quit Program Test	
Related to	UC - 009	
Description	Test if the program can be exited from within the program	
Priority	2	
Test	Step	
	1	Click "Exit" button
	2	Verify that a confirmation box appears
	3	Click "Exit Software" button
	4	Verify that software has been closed
	5	Verify that the temp file folder is empty of temp file that our program produces

3. Non-Functional Requirements

This section presents the non-functional requirements. These are goals which are not related to the functionality of the system, but are still important, such as speed requirements and data security. Each one is presented with a description and the tests required for it.

Number	NFR - 001	
Name	Runs on lab computers	
Description	The Software must run on the Windows 11 computer that is used by the lab	
Priority	5	
Tests	NFT - 001	
	1	Load software onto the lab machine
	2	Run all automated tests to verify that they pass on the lab machine

Number	NFR - 002	
Name	Signal security	
Description	The software shall only send a specified set of signals to the machine	
Priority	2	
Tests	NFT - 002	
	1	Initiate an instruction to create a package
	2	Have only necessary instructions in package
	3	Verify package information

Number	NFR - 003	
Name	Data format	
Description	After collecting the data from the machine the software must format the data into a JSON file using the format specified by ICN	
Priority	5	
Tests	NFT - 003	
	1	Capture a Sample and save it
	2	Verify that the output is a JSON File
	3	Verify that the output has the following 3 fields: label, data, type
	4	Verify that the label field is a string
	5	Verify that the type field is a string
	6	Verify that the data field is an array of arrays that each contain 2 floats

Number	NFR - 004	
Name	Operates all day	
Description	The Software must be able to operate over the course of one day of sample collection with minimal degradation of performance	
Priority	3	
Tests	NFT - 004	
	1	Launch the software once the labs open at the start of a day
	2	Run a control trial of 10 samples and measure the time it takes to complete
	3	Automatically trigger the machine to start a trial of 5 samples every 15 minutes
	4	After 16 hours of trials, verify that the software remained operational during the entire day
	5	Run a test trial of 10 samples and record the time it takes to complete
	6	Verify that the test trial time is no more than 1.1x the control trial time

Number	NFR - 005	
Name	Operates during continuous use	
Description	The Software must be able to operate over a period of continuous use	
Priority	4	
Tests	NFT - 005	
	1	Run a control trial of 10 samples and measure the time it takes to complete
	2	Run samples as often as the machine can safely operate
	3	After 4 hours of samples, verify that the software remained operational during the test
	4	Run a test trial of 10 samples and record the time it takes to complete
	5	Verify that the test trial time is no more than 1.1x the control trial time

Number	NFR - 006	
Name	User experience	
Description	The Software must be understandable to new users with no outside assistance	
Priority	3	
Tests	NFT - 006	
	1	Gather students with no prior experience with this software
	2	Instruct student to try to collect a sample from the machine
	3	Record times of each student
	4	Verify that all students collected a sample

Number	NFR - 007	
Name	Common visual language	
Description	Both pieces of software must use common design elements	
Priority	1	
Tests	NFT - 007	
	1	Verify common fonts, sizes, text, colors, and button positions/shapes across both softwares. Ensure accuracy to 99%

Number	NFR - 008	
Name	Inactivity timeout	
Description	The system must automatically log out any user after 30 minutes of inactivity	
Priority	2	
Tests	NFT - 008a	
	1	Log into the system
	2	Leave the screen inactive for 30 minutes
	3	Verify that the inactivity warning prompt appears
	4	Verify that the user has been logged out after 30 minutes
	5	Repeat steps 1-4 for each possible page a user can navigate to
	NFT - 008b	
	1	Log into the system
	2	Perform continuous activity for 30 minutes
	3	Verify that the user is still logged in

Number	NFR - 009	
Name	Clear data	
Description	Upon logging a user out, the system must clear all information and return to the default configuration	
Priority	2	
Tests	NFT - 009	
	1	Log into the system with a user
	2	Change sample settings
	3	Run a sample
	4	Log out of systems
	5	Log back in as the same user
	6	Verify that the sample settings have returned to their default values
	7	Verify that the previously collected sample isn't available

Number	NFR - 010	
Name	ICN outage	
Description	The system shall store the data locally and report if it receives no response from ICN within 5 seconds	
Priority	1	
Tests	NFT - 010a	
	1	Block ICN's IP address
	2	Attempt to log into ICN
	3	Verify that the user is notified if no response is received within 5 seconds
	4	Attempt to collect a sample and send it to ICN
	5	Verify that the user is notified if no response is received within 5 seconds
	6	Verify that the Data is saved locally
	NFT - 010b	
	1	Log into ICN
	2	Block ICN's IP address
	3	Attempt to collect a sample and send it to ICN
	4	Verify that the user is notified if no response is received within 5 seconds
	5	Verify that the Data is saved locally

Number	NFR - 011	
Name	Help Section	
Description	The system shall have a tutorial available explaining how to utilize the software on each screen except the login screen.	
Priority	2	
Tests	NFT - 013	
	1	Login to the software
	2	Verify that there is a button to bring up the tutorial in the instrument page
	3	Systematically go to every screen in the software
	4	Verify that there is a button to bring up the tutorial on each screen
Open Issues	Should any other screens be excluded? Should we list the screens that will have tutorials instead?	

4. User Interface

See the "User Interface Design Document" for Chemistry Instrumentation: Control, Data Acquisition and Offline Analysis for visualizations and information on the user interface.

5. Deliverables

This section describes what items will be delivered as part of the project, when they will be delivered, and how they will be delivered.

Deliverable Item	Description	Format	Delivery Method	Delivery Date
System Requirement Specification (SRS)	Defines all system, functional, and non-functional requirements agreed upon by the client and the development team	PDF	Git Repository & digital PDF submission	10/29/2025
System Design Document (SDD)	Describes the system's architecture, design components, and interfaces	PDF	Git Repository & digital PDF submission	11/17/2025
User Interface Design Document (UIDD)	Contains all user interface mockups, flow diagrams, and layout specifications	PDF	Git Repository	11/17/2025
User Manual	Guide explaining how students and lab managers can install, operate, and troubleshoot the software	PDF	Git Repository and PDF sent to client	TBD
Administrator Manual	Explains installation, configuration, and maintenance for lab administrators	PDF	Git Repository and PDF sent to client	TBD
Source Code Repository	Complete, well-documented source code implementing all approved requirements	Python/C++/Visual Basic files.	Git Repository	TBD
Executable Program	Fully compiled and tested version of the software	Windows executable (.exe)	Git repository release and USB copy	TBD
Biweekly status report	Regular project progress updates submitted to the instructor	PDF	Digital PDF submission	Every other Monday

6. Open Issues

Below is a table that describes current open issues in the project's development. Along with the issue number is the date it was issued, the priority in which the issue should be resolved, the team member the particular task was assigned to, and the date that the issue was resolved.

Disclaimer: It is very likely that new open issues will be added to this table in future versions of this document as the project progresses.

No.	Issue Date	Severity	Issue Title	Priority No.	Assigned to	Date Resolved	Status
1	10/27/2025	5	How do students interact with blank samples?	1	Eric Jestel	11/13/2025	Resolved
2	10/27/2025	4	Lab access permissions	4	Alex Ayer		In progress
3	10/27/2025	2	Computer internet connections	5	Alex Ayer		In progress
4	10/27/2025	3	Finalize languages used and frameworks	2	Chris L.	11/06/2025	Resolved
5	10/27/2025	2	Research UI/accessibility information	3	Bryce Roy		In progress
6	10/28/2025	3	What are 'good' settings for the program?	7	Conall G.	11/13/2025	Resolved
7	10/28/2025	3	Should the settings be modifiable by users?	8	Alex Ayer	11/13/2025	Resolved
8	10/29/2025	4	How do logins to remote server work?	6	Eric Jestel		In progress
9							
10							

Appendix A - Agreement Between Customer and Contractor

This document outlines the software requirements that are agreed upon by the customer and the development team for the Chemistry Instrumentation project. By signing and dating this document, both parties agree on the information outlined within the contents of this document.

In the event that any information within this document changes, the software team is liable to meet with the client and review any alterations. If both parties agree to the new information, new signatures and dates will be recorded in the table below indicated with a new version.

Mitchell Bruce		11/17/2025
Customer Name	Customer Signature	Date
Alexander Ayer		11/17/2025
Team Member	Team Member Signature	Date
Conall Gouveia		11/17/2025
Team Member	Team Member Signature	Date
Eric Jestel	/s/ Eric Jestel	11/17/2025
Team Member	Team Member Signature	Date
Chris Letourneau	/s/ Chris Letourneau	11/17/2025
Team Member	Team Member Signature	Date
Bryce Roy		11/17/2025
Team Member	Team Member Signature	Date

Appendix B – Team Review Sign-off

Below are the signatures of all the team members working on the chemistry instrumentation project. By signing this document, all team members agree to its contents and format. Along with signatures, any member can provide comments in the provided space to voice any concerns or minor disagreement that they may have with the information provided within this document.

Alexander Ayer		11/17/2025	N/A
Print Name	Signature	Date	Comments
/s/ Eric Jestel			
Eric Jestel		11/17/2025	N/A
Print Name	Signature	Date	Comments
/s/ Chris Letourneau			
Chris Letourneau		11/17/2025	N/A
Print Name	Signature	Date	Comments
/s/ Conall Gouveia			
Conall Gouveia		11/17/2025	N/A
Print Name	Signature	Date	Comments
/s/ Bryce Roy			
Bryce Roy		11/17/2025	N/A
Print Name	Signature	Date	Comments

Additional Comments:

Appendix C – Document Contributions

All team members contributed to the initial drafting of the functional requirements and writing of the nonfunctional requirements.

Eric Jestel: Coordinated tasks for SRS, reviewed 1.3, wrote 1.1, 1.4, 2, each section description, title page, and ensured consistent formatting and language throughout. Edited 1, 2, and 6.

Alex Ayer: Created deliverables table, assisted with functional and non-functional requirements, set up appointments with the client, created an open issues list, added appendix A & B, edited use cases & use case tests.

Bryce Roy: Set up SRS template document, worked on Use Case Tests/Functional Requirements Tests, provided clarification on use cases and associated tests.

Chris L: Non-Functional Requirements and Functional Requirement Tests

Conall Gouveia: Purpose of the Product, Review of Functional requirements & Deliverables