UTS: ENGINEERING & INFORMATION TECHNOLOGY

SUBJECT NUMBER & NAME	NAME OF STUDENT(s) (PRINT CLEARLY)	STUDENT ID(s)
41113 Software Development Studio	*Luke Herron* *Sanghyeon Park* *Bhavya Khemlani* *Chi-Tai Nguyen* *Mitchell Lee* *SURNAME* *FIRST NAME*	24494116 13980154 24445076 14252593 13248202

STUDENT EMAIL	STUDENT CONTACT NUMBER
Luke.w.herrron@student.uts.edu.au **Sanghyeon.Park-4@student.uts.edu.au** **Bhavya.khemlani@student.uts.edu.au** **Chi-tai.nguyen@student.uts.edu.au** **Mitchell.Lee@student.uts.edu.au**	**0450-040-948** **0478-101-095** **0478 949 133** **0460 846 370** **0416-684-820**

NAME OF TUTOR	TUTORIAL GROUP	DUE DATE
Grace Billiris	**03**	**13/11/2023**

ASSESSMENT ITEM NUMBER & TITLE
Assessment task 3: Product Assessment

I acknowledge that if AI or another nonrecoverable source was used to generate materials for background research and self-study in producing this assignment, I have checked and verified the accuracy and integrity of the information used.
I confirm that I have read, understood and followed the guidelines for assignment submission and presentation on page 2 of this cover sheet.
I confirm that I have read, understood and followed the advice in the Subject Outline about assessment requirements.
I understand that if this assignment is submitted after the due date it may incur a penalty for lateness unless I have previously had an extension of time approved and have attached the written confirmation of this extension.

Declaration of originality: The work contained in this assignment, other than that specifically attributed to another source, is that of the author(s) and has not been previously submitted for assessment. I have rewritten any material provided by AI or other nonrecoverable sources and where appropriate acknowledged their contribution. I understand that, should this declaration be found to be false, disciplinary action could be taken and penalties imposed in accordance with University policy and rules. In the statement below, I have indicated the extent to which I have collaborated with others, whom I have named.

No content generated by AI technologies or other sources has been presented as my own work and I have rewritten any text provided by AI or other sources in my own words.

Statement of collaboration:

L.H.
S.P.
B.K.
H.A
M.L

Signature of student(s) ____ **Date** _____**13/11/2023**_____

Solution Architecture

CirQuS Cryogenic Wiring SDS Project

Version	4.0
Author(s)	Mitchell Lee, Bhavya Khemlani, Chi-Tai Nguyen, Sanghyeon Park, Luke Herron, Husam Ajaj

Table of Contents

1. Document Management

1.1 Revision History

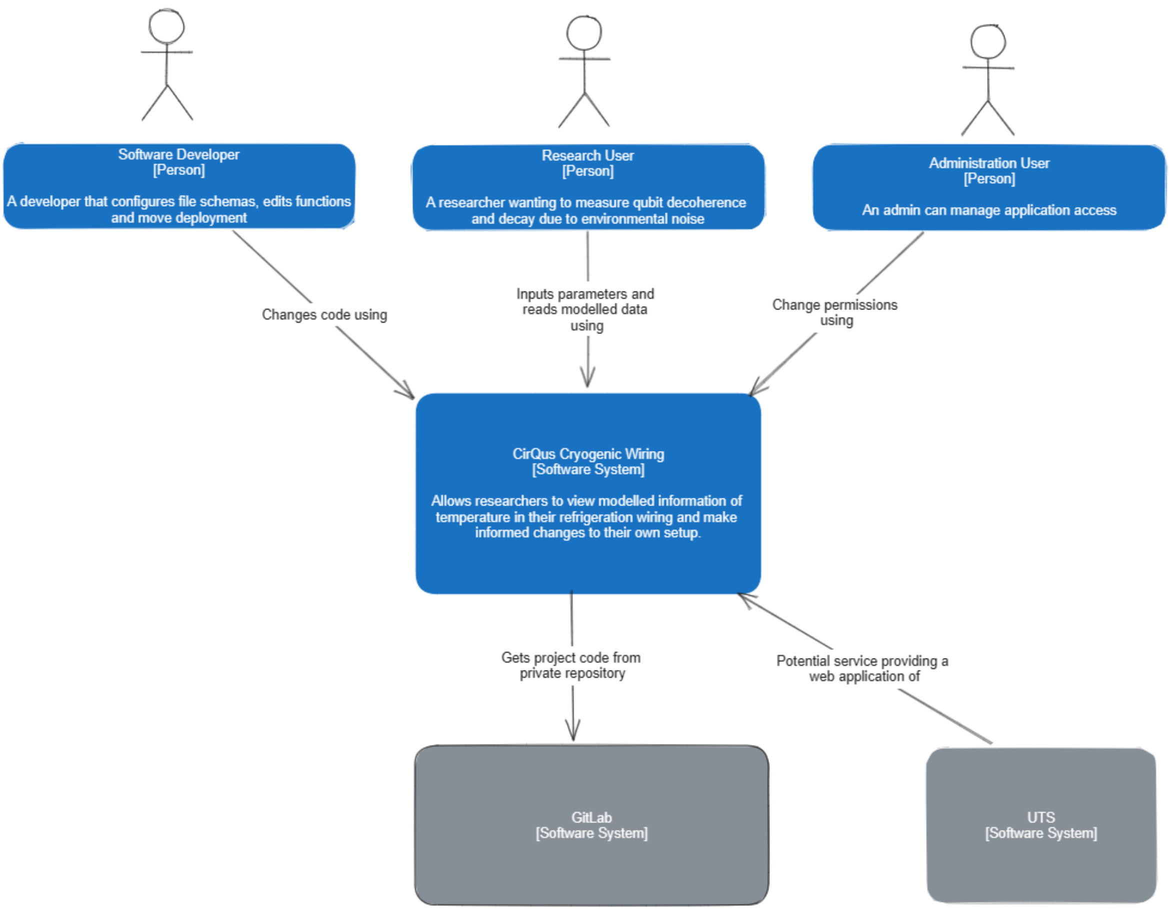
Revision History

Date	Version	Description of Change	Author
20/10/2023	1.0	INITIAL DRAFT	Mitchell
10/11/2023	2.0	Overall Solution, Interaction Architecture	Calvin, Bharav, Luke, Sanghyeon

11/11/2023	3.0	Final Draft	Calvin, Bharav, Luke, Sanghyeon
13/11/2023	4.0	Final Report	Luke

2. Solution Architecture

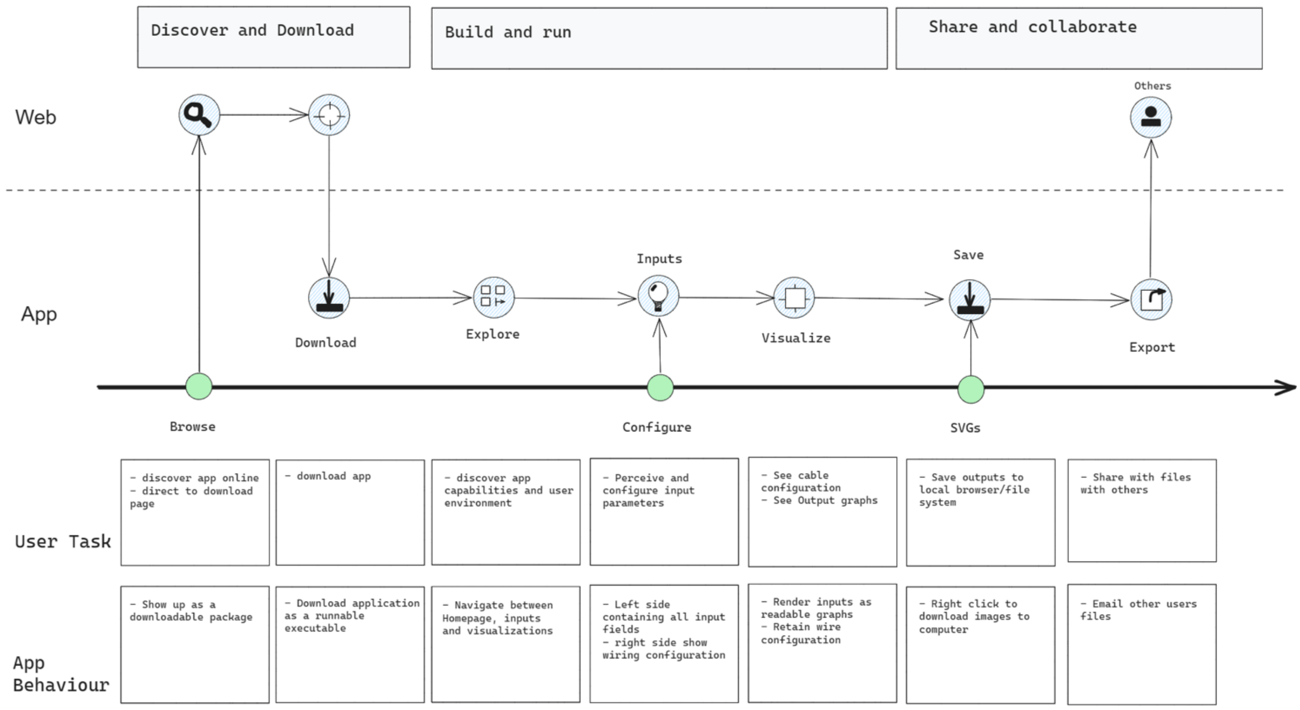
The diagram below outlines, at a high-level, the Cryogenic Wiring GUI software solution. The visual representation demonstrates the various interactions the system will have with its environment both in a stakeholder and end-user capacity and from an integration and wider software viewpoint. In doing so, the system will attempt to allow researchers to view modelled information of temperature in their refrigeration wiring and make informed changes to their own setup.



3. Interaction Architecture

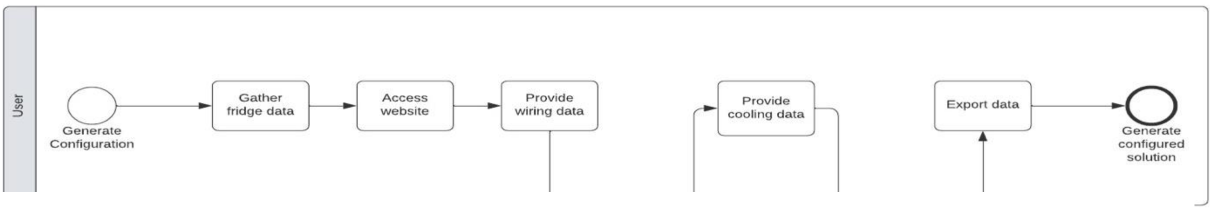
The interaction architecture diagram that follows reflects the key user interface components (UXD/CXD) and how users and the system interact with one another. Each are also provided a description to detail the steps and iterations of the system's application.

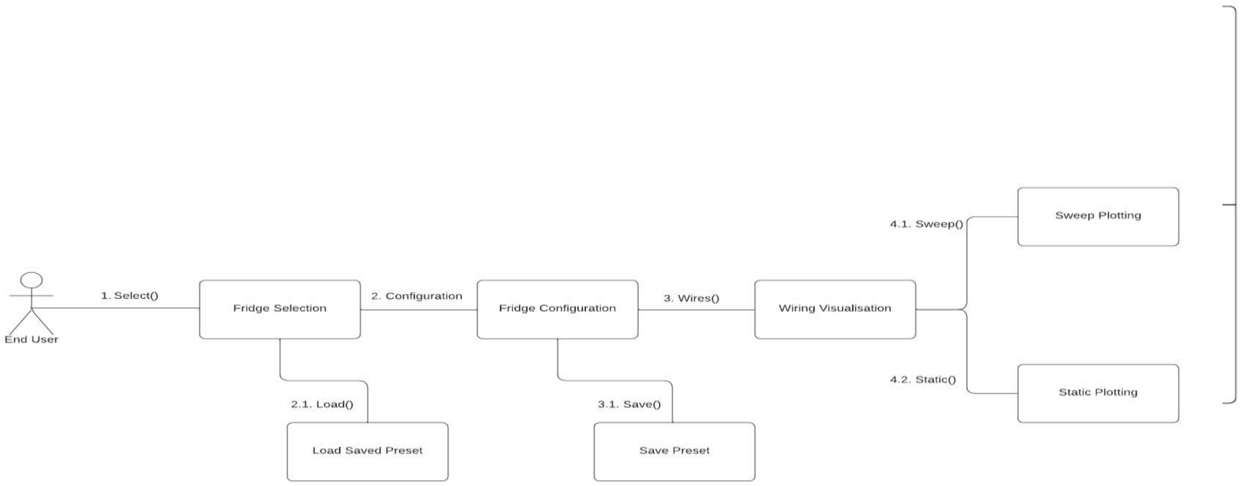
Solution Architecture



4. Business Architecture

The CirQuS organisation's business architecture is detailed in the following BPMN and communication diagrams that serve to capture the functional processes to be carried out. The application is designed to function as stand-alone in the monitoring of fridge data. It is designed for individual researchers to be able to input their own fridge measurements and generate wiring configurations and heat load models in order to monitor and develop their own experiments, and as such can fit into any cryostat research process. The key elements of this would be the user and the system itself, as the business process functions as a two-part mechanism between the user and the data processing application.

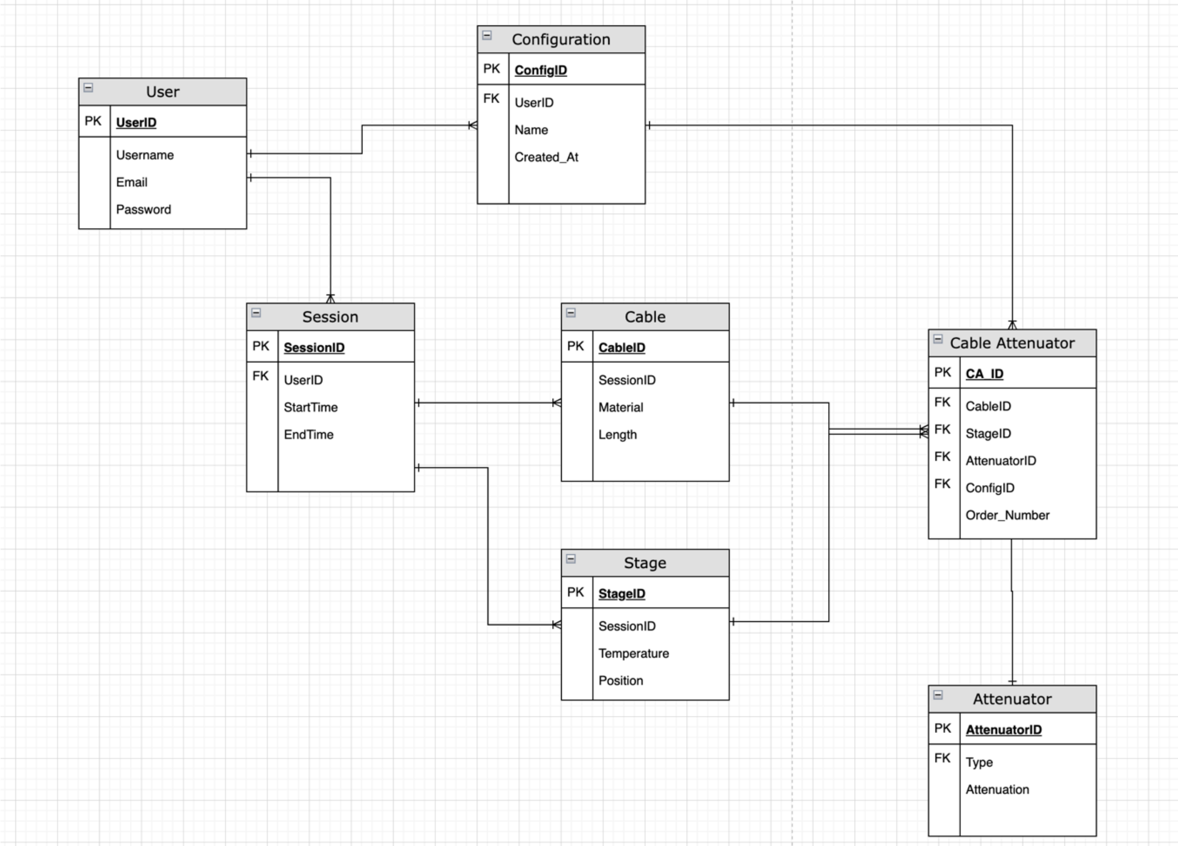




5. Information Architecture

The CirQuS UTS project aims to simplify the complex task of cryogenic wiring by offering a user-friendly software tool. To make this tool effective and easy to use, we've designed a clear and flexible Information Architecture. This setup manages different types of data, like user details and wiring configurations, and explains how they're related to each other. We use an Entity-Relationship Diagram (ERD) to visualise these connections and a Data Dictionary to describe what each type of data means. This approach makes the system easier to understand, manage, and extend, aligning with our goal to make advanced cryogenic science more accessible to a wider audience.

5.1 Entity Relationship Diagram



1. User: Represents users who can log into the system. Each user has a unique identifier (`UserID`), along with `Username`, `Email`, and a hashed `Password`.

2. Session: Contains session-related information for users. The session is identified by a unique `SessionID` and linked to a user by `UserID`. It also has `StartTime` and `EndTime` fields to keep track of the session duration.

3. Configuration: Represents the specific wiring configurations created by users. Each configuration is identified by a unique `ConfigID` and linked to a user by `UserID`. Other details like the `Name` and `Created_At` timestamp are also stored.

4. Cable: Stores information about the cables used in configurations. Each cable is identified by a unique `CableID` and optionally linked to a session (`SessionID`). Attributes include the `Material` and `Length` of the cable.

5. Stage: Contains details about the stages within the cryostat. Each stage is identified by a unique `StageID` and optionally linked to a session (`SessionID`). The `Temperature` and `Position` of each stage are stored.

6. Attenuator: Stores information about attenuators. Each attenuator is identified by a unique `AttenuatorID`. Attributes include `Type` and `Attenuation`.

7. Cable_Attenuators: This is a complex entity that represents the relationships between cables, attenuators, and stages within a configuration. It has a unique identifier (`CA_ID`) and references to `CableID`, `StageID`, `AttenuatorID`, and `ConfigID`. An additional field, `Order_Number`, helps in maintaining the structure.

The design aims for flexibility, enabling users to create and manage configurations while also allowing for the sharing and private session-based usage of cables and stages. It should also support complex configurations involving multiple cables, stages, and optional attenuators.

5.2 Data Dictionary

User

Field Name	Format	Length	Description
UserID	Integer	32-bit	Unique identifier for each user.
Username	String	50 characters	Username chosen by the user.
Email	String	255 characters	Email address of the user.
Password	String	255 characters	Hashed password for user authentication.

Session

Field Name	Format	Length	Description
SessionID	Integer	32-bit	Unique identifier for each session.
UserID	Integer	32-bit	Foreign key that links to User.
StartTime	Timestamp	N/A	Start time of the session.
EndTime	Timestamp	N/A	End time of the session.

Configuration

Field Name	Format	Length	Description
ConfigID	Integer	32-bit	Unique identifier for each configuration.
UserID	Integer	32-bit	Foreign key that links to User.
Name	String	255 characters	Name or title of the configuration.
Created_At	Timestamp	N/A	The date and time when the

			configuration was created.

Cable

Field Name	Format	Length	Description
CableID	Integer	32-bit	Unique identifier for each cable.
SessionID	Integer	32-bit	Foreign key that links to Session for private cables.
Material	String	50 characters	Material of the cable.
Length	Float	N/A	Length of the cable in meters.

Stage

Field Name	Format	Length	Description
StageID	Integer	32-bit	Unique identifier for each stage.
SessionID	Integer	32-bit	Foreign key that links to Session for private stages.
Temperature	Float	N/A	Temperature of the stage in millikelvin.
Position	Integer	32-bit	Position of the stage within the configuration.

Attenuator

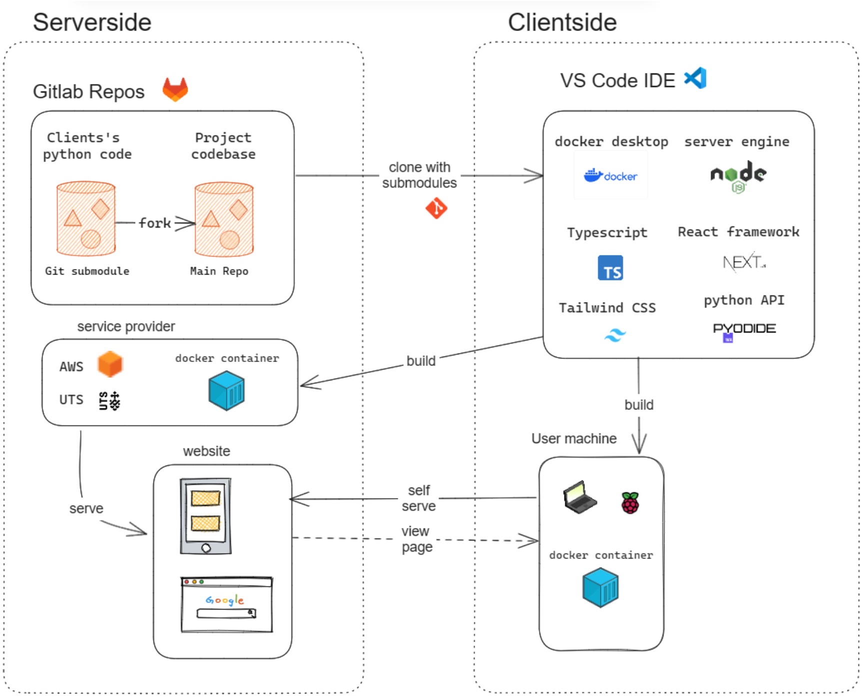
Field Name	Format	Length	Description
AttenuatorID	Integer	32-bit	Unique identifier for each attenuator.
Type	String	50 characters	Type of the attenuator.
Attenuation	Float	N/A	Attenuation value.

Cable Attenuator

Field Name	Format	Length	Description
CA_ID	Integer	32-bit	Unique identifier for each cable-attenuator relationship.
CableID	Integer	32-bit	Foreign key that links to Cable.
StageID	Integer	32-bit	Foreign key that links to Stage.
AttenuatorID	Integer	32-bit	Foreign key that links to Attenuator

			(optional).
ConfigID	Integer	32-bit	Foreign key that links to Configuration.
Order_Number	Integer	32-bit	Order of the cable within the stage for structuring.

6. Application Architecture



The system Architecture is a framework of server-side and client-side components. As the application should be publishable a package and used by the researcher community for personal use. This allows the user the freedom to choose between having themselves or UTS run and host the application for those that are averse to coding or modding. This results in a simple installation process and avoids user chagrin from manual configuration. Or as an alternative for interested parties, they can clone the repository and configure their own application if they wish.

7. Security Architecture

We'll implement a sophisticated logging and anomaly detection module within the Model layer. This module will generate comprehensive audit logs, capturing both user interactions and system events. It will employ machine learning algorithms to identify abnormal patterns and potential security threats in real-time.

To ensure compliance with privacy regulations, the Model layer will incorporate a Privacy-By-Design Compliance Enforcer. This component will automatically anonymize and minimize sensitive data during storage and retrieval. It will also provide audit trails for demonstrating compliance.

The View layer will feature an Immersive Cryogenic Simulation User Interface (ICUI). This cutting-edge interface will provide a highly intuitive and immersive user experience, complete with 2D visualizations of cryogenic wiring configurations and real-time simulations.

The Controller layer will manage secure communication between the ICUI and the backend. It will employ the latest SSL/TLS protocols and implement strict API token authentication to ensure data integrity and confidentiality.

Our Controller will feature a Dynamic Code Security Inspector. This intelligent component will perform continuous code analysis, automatically detecting and mitigating security vulnerabilities in real-time. It will also integrate with threat intelligence feeds to proactively defend against emerging threats.

To maintain an airtight security posture, the Controller will include a Real-World Penetration Testing Orchestrator. This component will schedule regular penetration tests conducted by expert ethical hackers to identify and remediate vulnerabilities.

In the event of a security breach, the Controller layer will deploy an AI-Enhanced Incident Response Orchestrator. This orchestrator will coordinate incident response efforts, automatically isolating affected components, and initiating rapid incident containment and recovery procedures.

By implementing these cutting-edge components within the MVC architecture, our security framework will ensure the highest level of protection for sensitive cryogenic data, deliver an immersive user experience, and proactively defend against evolving cyber threats while maintaining regulatory compliance.

8. Appendices

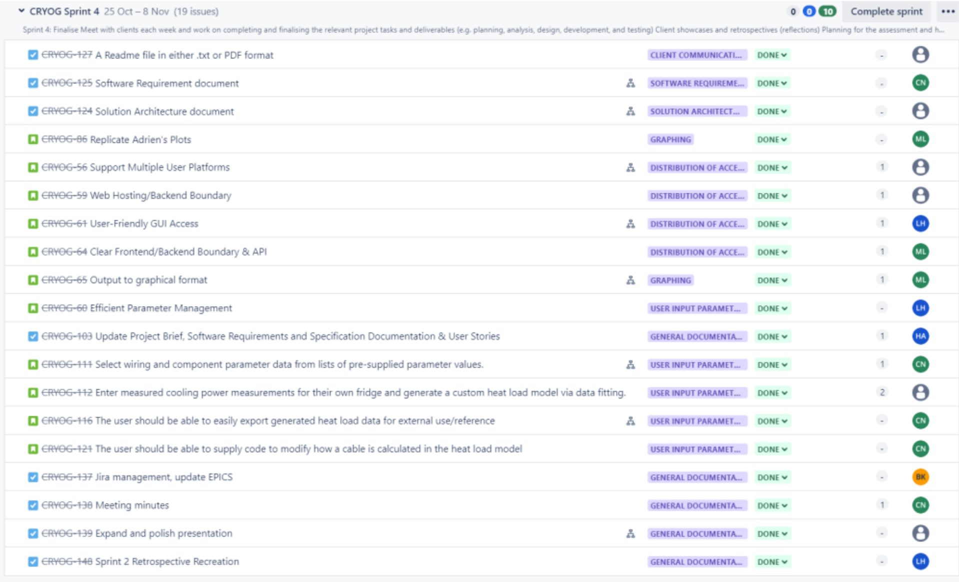
8.1 User Story/ Requirements Backlog (JIRA)

~~CRYOG-127~~ A Readme file in either .txt or PDF format	CLIENT COMMUNICATI...	DONE	-	
~~CRYOG-125~~ Software Requirement document	SOFTWARE REQUIREME...	DONE	-	CN
~~CRYOG-124~~ Solution Architecture document	SOLUTION ARCHITECT...	DONE	-	
~~CRYOG-86~~ Replicate Adrien's Plots	GRAPHING	DONE	-	ML
~~CRYOG-56~~ Support Multiple User Platforms	DISTRIBUTION OF ACCE...	DONE	1	
~~CRYOG-59~~ Web Hosting/Backend Boundary	DISTRIBUTION OF ACCE...	DONE	1	
~~CRYOG-61~~ User-Friendly GUI Access	DISTRIBUTION OF ACCE...	DONE	1	LH
~~CRYOG-64~~ Clear Frontend/Backend Boundary & API	DISTRIBUTION OF ACCE...	DONE	1	ML
~~CRYOG-65~~ Output to graphical format	GRAPHING	DONE	1	ML
~~CRYOG-60~~ Efficient Parameter Management	USER INPUT PARAMET...	DONE	-	LH
~~CRYOG-103~~ Update Project Brief, Software Requirements and Specification Documentation & User Stories	GENERAL DOCUMENTA...	DONE	1	HA
~~CRYOG-111~~ Select wiring and component parameter data from lists of pre-supplied parameter values.	USER INPUT PARAMET...	DONE	1	CN
~~CRYOG-112~~ Enter measured cooling power measurements for their own fridge and generate a custom heat load model via data fitting.	USER INPUT PARAMET...	DONE	2	
~~CRYOG-116~~ The user should be able to easily export generated heat load data for external use/reference	USER INPUT PARAMET...	DONE	-	CN
~~CRYOG-121~~ The user should be able to supply code to modify how a cable is calculated in the heat load model	USER INPUT PARAMET...	DONE	-	CN
~~CRYOG-137~~ Jira management, update EPICS	GENERAL DOCUMENTA...	DONE	-	BK
~~CRYOG-138~~ Meeting minutes	GENERAL DOCUMENTA...	DONE	1	CN
~~CRYOG-139~~ Expand and polish presentation	GENERAL DOCUMENTA...	DONE	-	
~~CRYOG-148~~ Sprint 2 Retrospective Recreation	GENERAL DOCUMENTA...	DONE	-	LH

8.1.1 Completed User Stories

CRYOG-57 Parameter Sweeping	USER INPUT PARAMET...	IN PROGRESS	2	CN
CRYOG-108 Upload wiring and component data in a standardised form	USER INPUT PARAMET...	IN PROGRESS	1	S
CRYOG-113 Individual sessions can be stored on the database and retrieved for later re-evaluation and updates	DISTRIBUTION OF ACCE...	IN PROGRESS	-	CN
CRYOG-119 New cable materials should be available to supply and save for use in later fridge configurations	USER INPUT PARAMET...	IN PROGRESS	-	CN
CRYOG-120 New cable presets should be available to determine and save for use in later fridge configurations	USER INPUT PARAMET...	IN PROGRESS	-	

8.1.2 Backlog from 11/11/23



CRYOG Sprint 4 25 Oct – 8 Nov (19 issues) 0 0 10 Complete sprint

Sprint 4: Finalise Meet with clients each week and work on completing and finalising the relevant project tasks and deliverables (e.g. planning, analysis, design, development, and testing) Client showcases and retrospectives (reflections) Planning for the assessment and h...

~~CRYOG-127~~ A Readme file in either .txt or PDF format	CLIENT COMMUNICATI...	DONE	-	
~~CRYOG-125~~ Software Requirement document	SOFTWARE REQUIREME...	DONE	-	CN
~~CRYOG-124~~ Solution Architecture document	SOLUTION ARCHITECT...	DONE	-	
~~CRYOG-86~~ Replicate Adrien's Plots	GRAPHING	DONE	-	ML
~~CRYOG-56~~ Support Multiple User Platforms	DISTRIBUTION OF ACCE...	DONE	1	
~~CRYOG-59~~ Web Hosting/Backend Boundary	DISTRIBUTION OF ACCE...	DONE	1	
~~CRYOG-61~~ User-Friendly GUI Access	DISTRIBUTION OF ACCE...	DONE	1	LH
~~CRYOG-64~~ Clear Frontend/Backend Boundary & API	DISTRIBUTION OF ACCE...	DONE	1	ML
~~CRYOG-65~~ Output to graphical format	GRAPHING	DONE	1	ML
~~CRYOG-60~~ Efficient Parameter Management	USER INPUT PARAMET...	DONE	-	LH
~~CRYOG-103~~ Update Project Brief, Software Requirements and Specification Documentation & User Stories	GENERAL DOCUMENTA...	DONE	1	HA
~~CRYOG-111~~ Select wiring and component parameter data from lists of pre-supplied parameter values.	USER INPUT PARAMET...	DONE	1	CN
~~CRYOG-112~~ Enter measured cooling power measurements for their own fridge and generate a custom heat load model via data fitting.	USER INPUT PARAMET...	DONE	2	
~~CRYOG-116~~ The user should be able to easily export generated heat load data for external use/reference	USER INPUT PARAMET...	DONE	-	CN
~~CRYOG-121~~ The user should be able to supply code to modify how a cable is calculated in the heat load model	USER INPUT PARAMET...	DONE	-	CN
~~CRYOG-137~~ Jira management, update EPICS	GENERAL DOCUMENTA...	DONE	-	BK
~~CRYOG-138~~ Meeting minutes	GENERAL DOCUMENTA...	DONE	1	CN
~~CRYOG-139~~ Expand and polish presentation	GENERAL DOCUMENTA...	DONE	-	
~~CRYOG-148~~ Sprint 2 Retrospective Recreation	GENERAL DOCUMENTA...	DONE	-	LH