Assignment Coversheet

UNIVERSITY OF TECHNOLOGY SYDNEY

UTS: ENGINEERING & INFORMATION TECHNOLOGY

SUBJECT NUMBER & NAME	NAME OF STUDENT(s) (PRINT CLEARLY)	STUDENT ID(s)
41114 Software Development Studio		

STUDENT EMAIL	STUDENT CONTACT NUMBER

NAME OF TUTOR	TUTORIAL GROUP	DUE DATE
	Fridge Green	21/05/2023

ASSESSMENT ITEM NUMBER & TITLE

Assessment Task 2: Process assessment

☒ 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 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.

Statement of collaboration:

Signature of student(s) _____ **Date 19/05/2023**

TABLE OF CONTENTS

1 END TO END SPRINT PROCESS AND UNDERSTANDING OF SCRUM

Over the course of this semester, I have been working in a SCRUM team to develop a web based monitoring system for dilution refrigerators for CirQuS group. Agile is a way of working that promotes iterative project development to help with the delivery of small working increments of the overall solution. It places value in "individuals and interactions over processes and tools", "working software over comprehensive documentation", "customer collaboration over contract negotiation", and "responding to change over following a plan", as mentioned in the agile manifesto [1]. There are many different frameworks associated with the agile methodology, one of which is SCRUM. SCRUM is a framework for solving problems that supports the continuous improvement of the product, team and working environment. It works on the principal of a timeboxed effort to deliver a working solution at the end of that period. It has specified roles (product owner, scrum master, development team) for a cross functional team, events (sprint planning, daily SCRUM, sprint retrospective, sprint review), artifacts (backlog, sprint board), and rules which underpin the framework's successful usage. For my project, my role was within the development team, and I worked as a Business Analyst and UI/UX designer alongside Member A and Member B (see 3.1). The development team consisted of both the developers and the QA personnel, and the team leads fulfilled the role of both scrum master and product owner.

A number of different tools helped to enable the team to work in an agile manner, they include Jira for product backlog and task management, Confluence for meeting minutes, documentation, diagrams, and team rules, GitLab for version control of code, Microsoft Teams for meeting with the client, Facebook Messenger for instant messaging within the team, Drawio for diagrams, Mural for sprint retrospectives, and Figma for collaborative user interface design. These tools were used throughout the project to assist with project management, and support a quick feedback loop, both within the team and with the client.

In SCRUM a timeboxed effort, commonly referred to as a sprint, may range from 1-4 weeks but it is commonly 2 weeks in duration, such as the case for our project in Software Development Studio. Over the sprint the team commits to delivering a working component of the solution and self-organises tasks and efforts to meet that goal. This begins with sprint planning. Sprint planning is conducted at the beginning or just before a sprint starts and is a SCRUM ceremony where the team comes together to agree upon the goal and tasks for the upcoming sprint, it is also an opportunity to prioritise tasks towards achieving the goal and assign work to team members for completion. For my project, this was conducted by the team leads who outlined the goal for the sprint before consulting the team to confirm or clarify which tasks should be completed, and if there are task dependencies, when each task should be completed. This was completed on a whiteboard for ease of planning before translating the results onto Jira (see 3.2). For sprint 0 we started with creating the user stories and writing their acceptance criteria before creating the tasks to fulfil those stories. The tasks would be created so that a relationship existed between the task and the user story so we could see how the tasks help to fulfil those stories and which tasks complete which stories. This additionally helps in understanding when a user story is completed (see 3.3).

Once sprint planning was completed, the sprint would begin and the teamwork get to work on their assigned tasks, moving them across the different headings on the Jira board to reflect the current status (see 3.4). An important aspect in the movement of tasks under different headings (to do, in progress, in review and done) was understanding what each heading meant and in particular the process for reviewing and marking a task as done. We agreed upon a review process and a best practice for working with the code base to ensure that the team had a common understanding on how we would use Jira for task management (see 3.5). Another ceremony that is a part of the SCRUM framework is a daily SCRUM. It is an opportunity for the team to share their day to day progress and bring up any issues that are preventing them from progressing with their tasks, it is a way to look at the progress towards the sprint goal, plan work for the next day and understand how the team is progressing with regard to the completion of the product backlog. For my project we conducted a weekly SCRUM during the weekly SDS lesson time. The team would go around and share progress updates and call out any issues they faced or discoveries they have made which can help others as well. This was done at the beginning of the lesson before SCRUM of SCRUMs and help to inform the content shared in SCRUM of SCRUMs. SCRUM of SCRUMs is an extended agile methodology which allows multiple teams who are working on complex problems to connect and collaborate by providing insight into their team progress. The team leads would host both the weekly SCRUM and present the overall team progress at the SCRUM of SCRUMs. Additionally, both ceremonies are expected to be quite short in duration, and should run for about 15 minutes at max.

Another aspect of the weekly SCRUMs was a weekly meeting between the team and the client. This served as a sprint review or mid-sprint review as it was an opportunity for the team to share their progress with the client, CirQuS group and for the client to provide feedback or seek further clarification from the team. Our team held this meeting over Microsoft Teams and the team lead would host the meeting, redirecting the conversation to the relevant team member for different aspects. For these weekly client meetings, I often keep meeting minutes which helped to track the conversation between the team and the client and capture any action items from the client (see 3.6). Additionally, I showcased the user interface design to the client on different occasions and captured their live feedback by annotating the interface with comments (see 3.7).

At the end of each sprint there are 2 SCRUM ceremonies which are held, they are sprint retrospective and sprint review. The sprint retrospective is an opportunity for the team to reflect on their work over the past sprint and evaluate what went well, what didn't go well and what could be improved. It is feedback capturing event which enables the team to continually improve. For my project the team leads facilitated this through the usage of Mural, an online whiteboarding tool, where the team would place sticky notes in four quadrants which asked different questions. Once completed the team leads would go through the captured comments and the team would discuss any points of contention and how best to improve shortcomings. The team leads would then share these results on Confluence (see 3.8). After a sprint retrospective we have a sprint review. This is an event where the team showcases their results for the sprint to the client and obtains feedback from the client about the items that were done over the past 2 sprints. For my team we conducted this every second Wednesday during the class time, via Teams, and would demonstrate to the client the different aspects of the solution which were completed before asking for feedback and comments. We would aim to capture this feedback and comments in meeting minutes on Confluence. Any additional follow up between the team and the client could be handled by the team leads via email with CirQuS group or via the Teams group that we shared with CirQuS group (see 3.9). Any communication between the team and the client that was not immediately visible to the whole team would be shared either immediately through Messenger or if less urgent then at the following weekly SCRUM. The Messenger amongst the team also provided a space for team leads to share weekly SCRUM debriefs (see 3.10).

After the sprint is completed, we would begin a new sprint, starting once more from the planning stage. This process continues until the project is delivered or completed.

2 PERSONAL ROLE, LEARNINGS, AND PROCESS IMPROVEMENTS

As previously mentioned, my role within the team for this project was a Business Analyst (BA) and UI/UX designer, as such I contributed predominantly to the requirements elicitation and modelling, user interface design and occasionally kept meeting minutes (though I shared this role with Me
The team had a total of 3 BA and UI/UX designers, we operated as a sub-team within the greater project team, and I led this section.

Sprint 0 was the beginning of the project. During this sprint the team focused a lot on building a foundation and setting up the tools to support our agile development. For sprint 0 I worked on an initial user interface design and theme. I also helped to keep meeting minutes and created the user stories on Jira, adding in the acceptance criteria for each. Key learnings during this sprint including developing an understanding of the roles and responsibilities of a BA (a new role I had not previously fulfilled), learning to work in a large development team of 9 (one of the largest I have worked at uni), and building up a knowledge base of user interface design principles for web based applications (as I previously came from a mobile app design background). I chose to keep the meeting minutes in a format that would enable a reader to easily follow 'threads' of conversations and understand what was discussed and by whom (see 3.6). Regarding the user stories and acceptance criteria in Jira, I tried to make sure that the acceptance criteria were specific and measurable but also relevant to the user story and followed the 'given when then' structure. Furthermore, as the team had more than one UI designer, it was important that the tool used for design was going to support collaborative design, as such I chose Figma. I created a first iteration with a logo, colour palette and font as well, and shared the link with the team. This sprint was not tracked on Jira but sprint artefacts were recorded on Confluence (see 3.12).

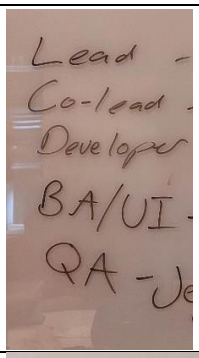
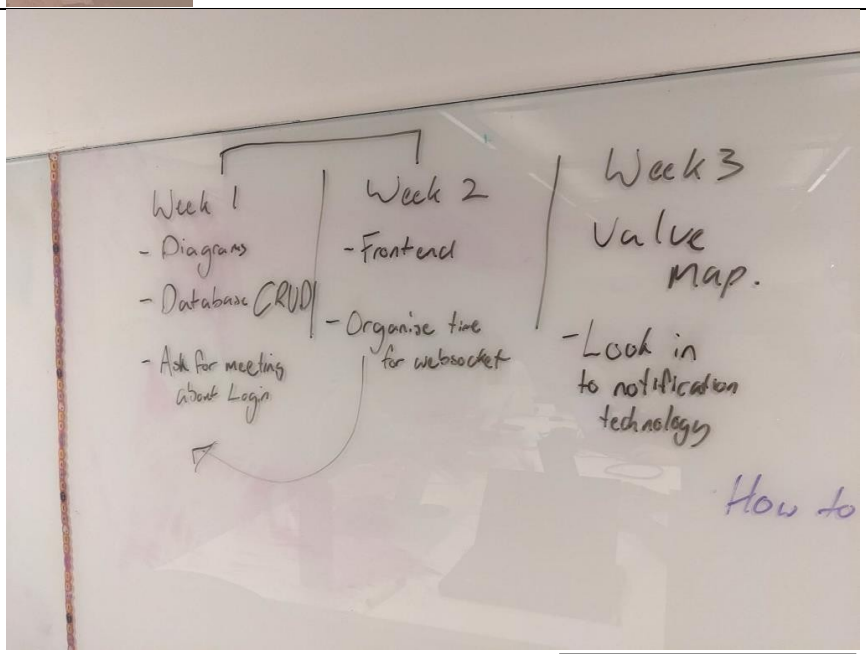
For sprint 1 the goal was to continue refining the design of the user interface and design out more of the relevant pages and features. I was assigned 5 tasks which were associated with all 12 user stories as it focused on creating application wide deliverables. The tasks included completing the first iteration of the user interface design, creating a user flow diagram, designing the system architecture, finalising the acceptance criteria, and defining a branching structure and GitLab practices. I worked with Member A and Member B on the user interface design, and we managed the completion of the design by creating sub-tasks for each associated page or feature like login and creating a new graph (see 3.11). I worked on defining a branching structure and GitLab practices with the developers to ensure it was understood and made sense, I was tasked with this aspect as I had more development experience than my team but would not be formally developing in any capacity as I wanted to challenge myself and try a new role. The finalisation of the acceptance criteria was done independently by me and checked by the team. As for the 2 diagrams, the user flow diagram, and the system architecture, I worked on these independently using Drawio before sharing the results with the team and consulting them for feedback and review. Over this sprint I was able to complete the acceptance criteria and the define the branching structure, however the initial UI, user flow diagram and system architecture were carried over into the next sprint (see 3.13).

For sprint 2 the goal was to adapt the initial user interface designed in sprints 0 and 1 to a new design format and align the user interface designs with the client's current logo and colour palette. I was assigned 13 tasks which were connected to all 12 user stories. The tasks included the 3 tasks carried over from sprint 1 as well as 9 new tasks focused on the redesign and rebranding of the initial UI to better suit the client's needs and colour palette, and 1 task associated with a use case diagram for the log out user flow. Completing the carry over tasks from sprint 1 were my top priority so I focused on completing the initial UI, user flow and system architecture before moving onto enhancing the UI for the second iteration. For the second iteration of the UI, I worked on this solo and sought feedback from the team and the client. A large portion of the second iteration of the UI focused on re-branding or re-designing features of the application such as the create a graph, account settings, add new fridge, notifications, system settings, edit graph, historical graphs, and user settings. The application design was uplifted and re-branded as the client provided us with their logo and colour palette, and I did more research into best practices as well as principles like the Gestalt principles, to enchance the UI/UX. The use case diagram for the log out flow was initially created by Member A and M ber B, but it was discovered that the system should connect to UTS's single sign on. This left Member A and Member B unsure how to progress, thus the task was reassigned to myself. During sprint 2 I was able to complete all of the tasks assigned to me and no tasks needed to be rolled over to the next sprint (see 3.14).

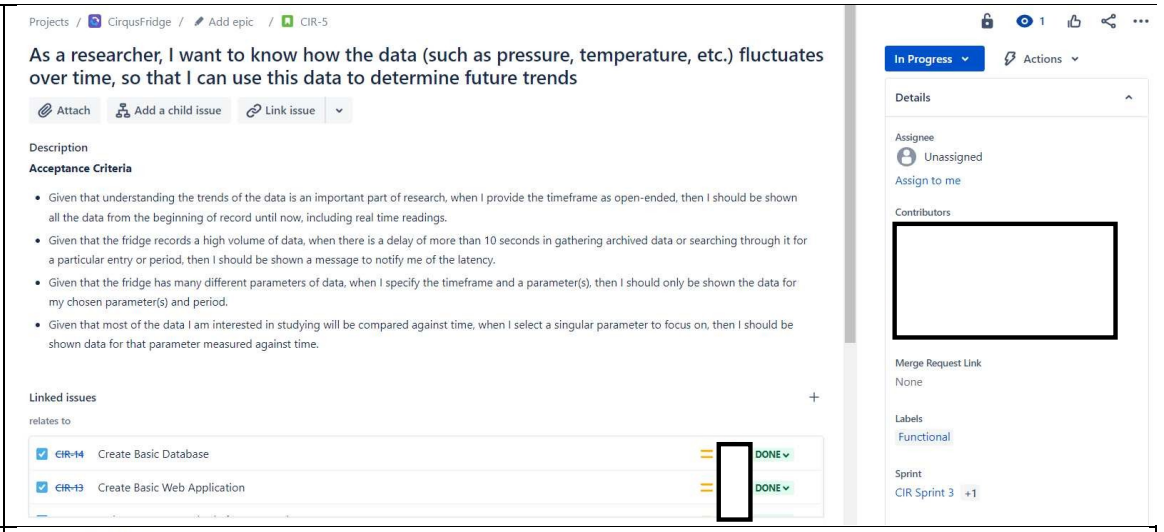
For sprint 3 the goal was to finalise the user interface designs and features to ensure they capture the client's needs and adhere to user interface design patterns and guides like the Gestalt principals and Material Design Guide. Additionally, as the client requested for the application to be cross-device functional, there was also a need to design sample pages which outlined how the application would work and appear on different sized devices. I was assigned 8 tasks, and these were associated with 12 user stories. The tasks included designing some screens for mobile capability as it was a requirement from the client and working on the user guide which would outline to the user how to use the app and what the elements of the UI do. During this sprint I completed 2 tasks which were assigned to me, and the rest were rolled over into sprint 4 (see 3.15).

Overall, I have been able to learn a great deal about working in an agile team and developed a better understanding on the roles and responsibilities of a BA and UI/UX designer. This project has allowed me to challenge myself and learn more about as BA role, which is a role I previously had not undertaken and something quite outside my comfort zone. It highlighted the importance of requirements elicitation and modelling, as well as my shortcomings in that area. As for UI/UX design, this project gave me the creative liberty to explore my design skills and provided me the chance to learn more about it for different platforms as I previously only had experience with mobile applications. I developed a better understanding of the Material Design Guide, a popular web design framework developed by Google, and learnt to design my pages to best fit those guidelines. I learnt the importance of word choice, and placement of user interface elements for optimising the user experience. The final takeaway from this project was I got to learnt how to navigate working in a large team as I previously had not worked in a university project team with 9 people, and learning to navigate and manage the team dynamic and expectations was something quite new. And as the team was so large, sub teams inevitably formed, and this allowed me to refine my leadership skills as I led the BA and UI/UX designers. This highlighted the importance of communication, leading by example and being proactive. In summary, this subject has been informative and helpful as it taught me more about the agile and SCRUM practices and the value of it, as well as provided an opportunity to practice them, it also allowed me to try out a new role and improve my skills more broadly.

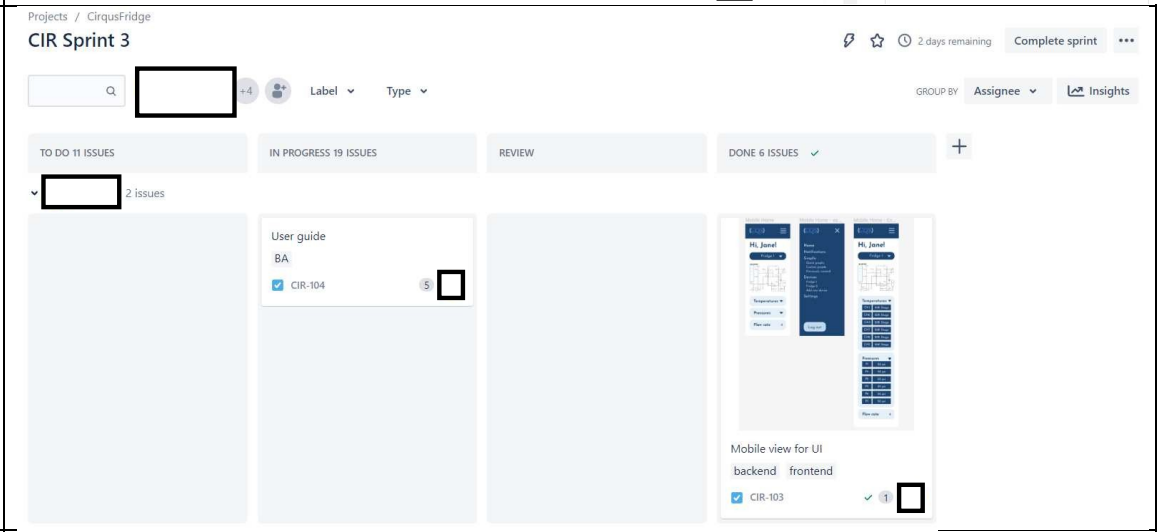
3 APPENDIX

Item		Evidence
3.1	TEAM ROLES	
3.2	SPRINT PLANNING EXAMPLE	 CIR-60 Implement CRUD for Graphs = DONE Done CIR-57 Create MVC Diagram (Version 1) = DONE Done CIR-56 Security Architecture = DONE Done CIR-55 Business Architecture = DONE Done CIR-54 Interaction architecture = DONE Done

3.3 USER STORY WITH ACCEPTANCE CRITERIA AND TASKS EXAMPLE



3.4 JIRA BOARD EXAMPLE



3.5 GIT PRACTICES AND JIRA REVIEW PROCESS

Git Practices and Branching Structure

Created by
Last updated: Apr 07, 2023 • 1 min read

Develop is our 'master' branch. You typically wouldn't commit code directly to it but contribute through Merge Requests.

We should have at least 1 person review the merge request before it is merged. Comments and improvements should be implemented or captured in a new Jira ticket before the merge request is merged with the develop branch. You can assign the merge request to yourself and request someone to review it.

Branch names should be the ticket number so **CIR-<number>**.

Jira Review Process

Created by
Apr 26, 2023 • 1 min read

1. Move task to review column
2. Re-assign task to person who you want to review the work
3. Send them a message to notify them that you want them to review it
4. The reviewer can review and check the task. Once happy they can re-assign back to the original assignee and move to done. If not good then move to TODO and re-assign to original assignee

3.6 MEETING MINUTES EXAMPLE

March 15th

Created by [redacted]
Last updated: Mar 22, 2023 • 7 min read

Summary

- Students should decide where the data from the fridge should be stored as part of the project development. Currently the log files are created daily and sampled every minute but it would be worth being able to view the changes in the data in 5 minute intervals.
- The need for a password to access the site will depended on how the site is implemented. As a baseline, the site should not be Google searchable but quick and easy to access. This may require interacting with IT to determine how we can leverage existing security practices (like single sign on) to make the site secure. CirQuS group will help to set up line of communication between students and IT.
- The fridges are connected to the internet and can be remote accessed via Remote Desktop but this has limitations. It would be ideal if the web page can allow the CirQuS team to issue basic commands to the fridge in the event that something has occurred but no CirQuS team member is on-site to resolve it immediately. This need not necessarily be implemented but the solution should be architected in a way such that this feature is possible. Likewise the ability to reboot the monitoring tool itself is not necessary but nice to have as a non-critical feature of the solution.
- In terms of producing graphs and charts, scatter plots and line charts will be most common. CirQuS group would like the ability to display historical data and specify the time frame the data should be displayed from.
- Some of the parameters that may be relevant to the CirQuS team but are not available in the data logs include the temperature, gauge pressure, power applied to the fridge stages, flow rate, snapshot of what the front panel of the fridge looks like, turbo speed and power.
- The ability for the solution to be platform agnostic and adaptable to different devices is important and should be considered as part of the functional requirements. The same is true for the solution modularity.
- Documentation will be developed progressively throughout the project.

Action items

- CirQuS to share data logs
- CirQuS to connect students with IT contact to discuss security of solution and how it may affect implementation.
- Students to prepare basic architecture of solution.
- Students to prepare project scope document for sign off between CirQuS and students.

Minutes

- CirQuS needs to sign off on the scope that the student team has decided for this semester. The scope factors in what is in scope for the project and what is out of scope.
  - [CirQuS] 1st and 3rd dot points are the same?
  - [Students] Sort of but the first one is defining the need for notification and the second is what data
- [Students] Clarification of data and graphing. Should there be templates like temp vs time?
  - [Cirqus] Most graphs will be against time. For customisability it can be like time range, x axis range or y axis range, or what we see on what plot.
  - [Students] Like how far back?
    - [Cirqus] Yes
  - [Cirqus] Maybe also plotting multiple parameters against time with 2 y-axis for the two parameters against time
  - [Cirqus] Maybe when having 3 parameters or more we can have multiple graphs instead of plotting it onto the same chart.

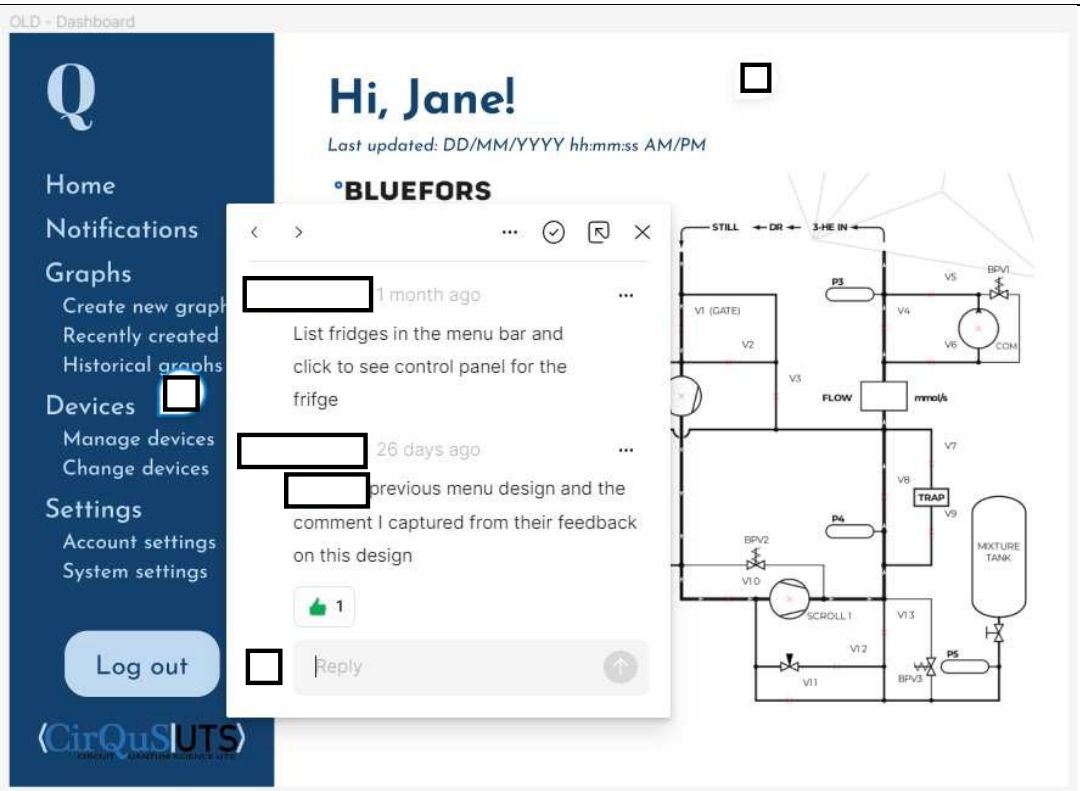
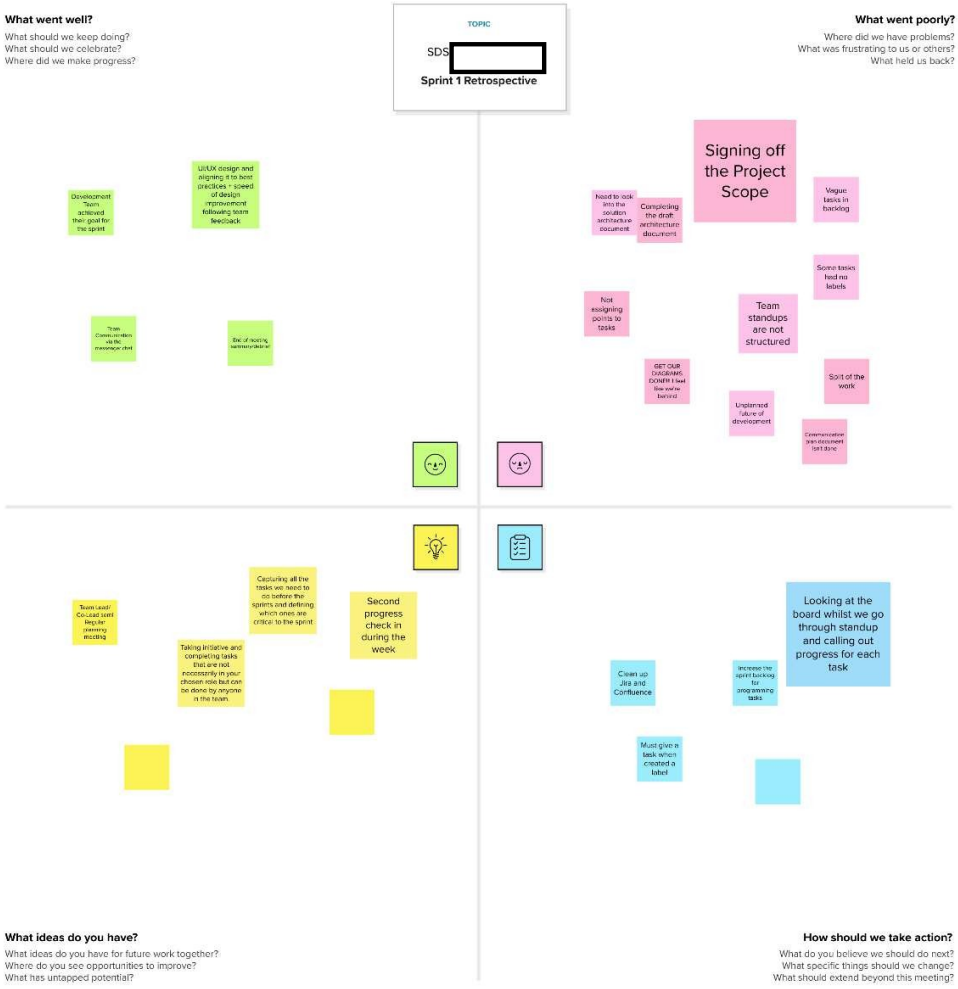
- [Cirqus] We also want to have the ability to change between log and linear values
- [Cirqus] We are working on lab access for next week to have the meeting there.
- [Cirqus] There are things that you can do in Blufors that is cool like scrolling along one axis or another instead of having to scroll a lot. This would be a nice feature to have but it is not critical to have.
- [Students] We also have identified out of scope things as well but we plan to design for these items just not developing or implementing it. An example is to have other devices connect to the monitoring software.
  - [Cirqus] Is there a reason for that?
  - [Students] We are concerned about the time but we can make it so that the solution is set up so it is easy to implement.
  - [Cirqus] We shouldn't put this out of scope but rather at a lower priority. To implement it we would need to allow the server to ping the blufors api.
  - **Agreement**: The ability to connect to other devices will be made of lower priority rather than out of scope.
- [Students] We have the sample data now, which is great but we would love to have documentation on how to read the logs and what it is about.
  - [Cirqus] We will give you a list of things that we want and the exact keys are defined in a table in the manual. We will also provide units for each of the data category.
- [Students] We have some access issues with GitLab could you help
  - [Cirqus] Yep! There are some weird access issues outside of the uni which you may need help with.
  - [Students] Yep we have discovered them so we should be able to work around it now.
- [Students] We want to outline our architecture for you but we plan to remove majority of the processing from the fridge computer and manage majority of it on another device (raspberry pi) to avoid overloading the fridge computer resources.
  - [Cirqus] We had a meeting with the uni regarding hosting but we will keep you updated on that as we learn more.
- [Cirqus] We will give you a list of everything we want to plot + units for each data item in the logs. Do you plan to copy the log to the raspberry pi or move to a database?
  - [Students] We plan to migrate to the raspberry pi and then move to a database for ease of use.
  - [Cirqus] Any particular database?
  - [Students] Not yet decided but any preferences?
  - [Cirqus] No preferences 🙂
- [Cirqus] Assume the meeting will be via teams if we can't arrange an in person meeting.

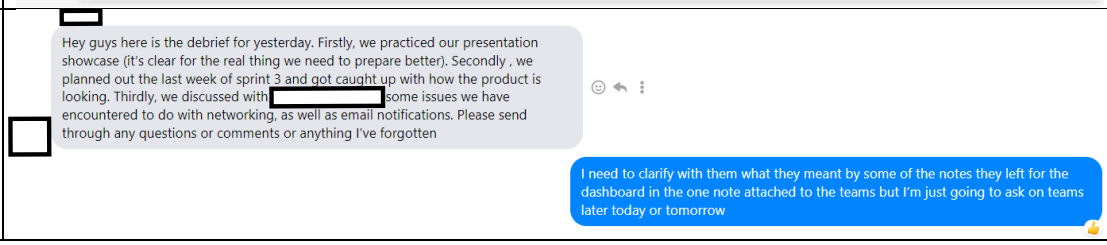
Action Points:

- Everyone ensures they can push and pull from the GitLab
- Ensure everyone has the tools (Javascript, VSCode, React, Express, Python, etc.)
- [redacted] and [redacted] follow up about SSO Login
- Set up web application - [redacted]
- Set up database - [redacted]
- Create fridge objects/chart objects that store locally -
- Taking data from the database (and generating graphs with it) -
- Continue working on the UI/UX
- Create tasks for user stories and other set up tasks
- Define branching structure + branch names
- Workflow
  - 2 reviewers before merge

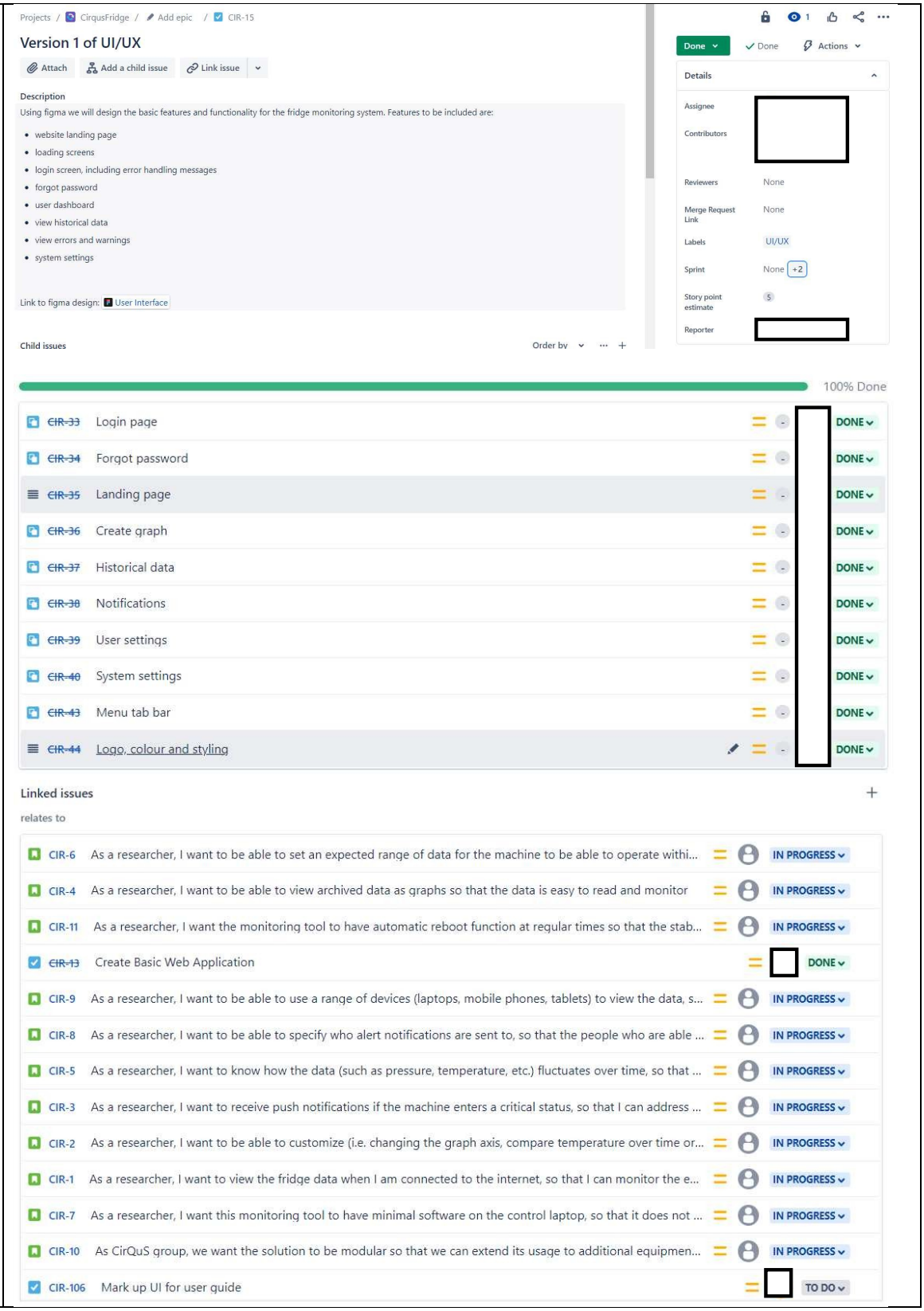
Development:

- VSCode has been decided as the IDE
- Will use Javascript, React, Python, Express
- Use Recharts JavaScript library to generate graphs
- Data should be stored in a database (off the laptop that controls the fridge) to allow for historic data to be displayed
- Modular
  - Fridge object type
  - Chart object type
    - Variables of X, Y, Scale, History
    - Notation type - do they have a preference? or should we allow them to swap between scientific and every-day?
    - Colour x, colour y
    - These variables should be stored locally, so graphs can be generated dynamically each time the page is loaded
- [redacted] to follow up at SSO Login with [redacted] and IT
  - Developers don't worry about login for now, to be implemented later depending on outcome of meetings
- Notification settings
  - Will they set these in the database? or from the website itself?
    - This will depend upon the security of the webpage, and who is able to access it
- Database - Postgres

3.7	USER INTERFACE WITH COMMENTS FROM CLIENT	
3.8	SPRINT RETROSPECTIVE EXAMPLE	

3.9	CIRQUS TEAMS QUERIES	
3.10	WEEKLY DEBRIEF	

3.11 UI VERSION 1 SUB TASKS



3.12	SPRINT 0 ARTEFACTS	3.12.1 Meeting Minutes

March 8th

Created by [redacted]
Last updated: Mar 22, 2023 • 8 min read

Summary

- The solution that the students work on should be focused on monitoring the fridge so CirQuS can be notified when fridge systems are behaving outside their expected range. The solution should not aim to control the fridge system as software already exists for this need.
- Some things that students should monitor include the valves, pressure and temperature of the fridge but there may be other pieces of information that are valuable to the situation
- The system that the students develop should meet the following criteria:
  - It should be web based.
  - It should have an easy interface that is accessible on different devices and from different locations e.g. at home on a phone

- The solution should consider the safety of the equipment and provide notification to appropriate personnel when operating outside its safe range.

- There are existing working practices set up by CirQuS that students should utilize.
  - CirQuS use teams as a centralised place for communication and collaboration and have OneNote integrated into the teams group to allow for brainstorming and meeting minutes.
  - CirQuS has GitLab for version control of code.
  - CirQuS utilises the Sharepoint associated with the teams channel to store other documentation.

Outcomes

- For CirQuS
  - Share data logs from the fridge
  - Share information API
  - Share installation information on Bluefors software
  - How to split the project
    - Same project?
    - Split it into a new feature?
    - Split it into mini components for the 2 teams?
  - Provide access to
    - GitLab
    - Teams
    - Sharepoint
  - Connect students to IT so we can discuss how to implement the web hosting part of the solution

- Organise lab and equipment tour to provide students with context on the equipment their solution should interact with

- For students
  - Set up a time for regular meetings.
  - Design solution as if we are working on the same project.

Full Meeting Notes

- CirQuS was not aware of the two teams but they need to clarify if there is an additional feature or some solution to split the work across the two teams
- CirQuS plans to organise the lab and showcase the equipment that the project concerns.
- Fridge gets to 10-20 millikelvin.
- Controls for the experiment/fridge don't live directly on the fridge but at another location. It interacts with the fridge using proprietary software. The software comes with the fridge.
  - The software tracks information as a function of time such as temperature, pressure etc.
  - Fridge is controlled via gas flows, turbo pumps.
  - Fridge valves and turbos can be controlled by fridge software.
- Need: monitoring software for the fridge not to control the fridge.
  - Software should not control the fridge, for safety.
  - There are situations where things can go wrong so monitoring would be helpful with this. It can happen during bad ideas such as crashing the turbo. It typically doesn't happen during normal operations.
    - E.g. the turbo is spinning and then you try to slow it down without following proper procedures can cause destruction.
  - Something that can happen during operation is when there are superconducting magnets in the fridge. When the fridge is operating, they work on cryogenic gas being pumped around through to the fridge. Once the fridge is cold then everything inside the fridge is liquid like helium would be liquid.
    - E.g., suppose the fridge is cooled down and there is a superconducting magnet inside but if there is something wrong with the fridge and the magnet goes above its superconducting temperature its resistance increases and it can carry a lot of current which can create a lot of heat and vaporise a lot of cryogenic gas which can lead to an exposure.
  - Monitor – valves, pressure, temperature etc.
    - To be able to look at the situation.
  - Easy interface – At home and able to login into a web page from another device. Ideally the web page is hosted on a UTS web address. The web page can show state and some configurability to see what is happening with different aspects.
  - Safety checks – Parameter ranges for different values. Need to provide operating conditions such as sending a warning when the value lives outside the expected parameter ranges. This can be in the format of text messages or emails to the relevant people.
- A potential second stage process for the project – currently setting up measurements in the lab and it would be nice to have the fridge software in a modular manner to be able to apply it to different contexts.
  - E.g. for an experience that is taking data we may want to see what is happening with the data rather than checking in on the fridge.

- Web part may be done agnostic to what it's attaching to. Then the other part can be a driver and allow the portal to plugin to the same backend. So, we can have 2 things go to the same backend.
    - Modularise the frontend from the backend for that piece of equipment. So it can allow for expansion into other equipment, though a driver will be required.
    - The solution should be replicable for each fridge so each fridge can have its own portal. Currently there are 2 fridges but there may be more in future.
- CirQuS has established data policies for data management and code management.
    - They have an existing GitLab repo for UTS research. It's hosted by UTS hosted. It is called CirQuS_FridgeMonitoring.
    - All code written for the group goes into this repository and with version control.
        - There may be a second repo if the project is split in that manner or it may just use the same once.
    - CirQuS has created a team for the fridge project. It's called CirQuS Cryogenic Wiring.
    - CirQuS uses OneNote in the teams to record things such as brainstorming and meeting minutes.
        - The sharepoint directory attached to the team has a documents folder.
    - They prefer centralised storage of assets such as code and documentation.
        - Student teams prefer to split documentation for context bounding.
        - CirQuS agrees that it can be done.
        - Agreement: Project repository and other tools will be paused until CirQuS can confirm how the project will be split across the team.
- Do you have a specification for the Bluefors API?
    - [Cirqus] There is information on it but it's not publicly accessible.
    - [Students] Do we have access to the software for implementation?
        - [Cirqus] A copy can be provided. It can technically be installed on any computer.
        - [Cirqus] Suggestion: write a list of things that cirqus can provide and that students need access to.
        - [Cirqus] In terms of fridge software access it needs to be done in a controlled manner.
        - [Students] How do we test it?
            - [Cirqus] We currently don't have a way to test it, but we have cool downs coming up that people can observe. The bluefors software has controls for cooldown but this access won't be provided to students due to the cost of it and the risk. But it is a good idea.
        - [Cirqus] We try the API and see how much we can do once students install the software on their machines. But it will go looking for the equipment.
            - [Students] How is it connected?
            - [Cirqus] It is connected to the electronics via USB then to the internet.
            - [Cirqus] The fridge has a computer that is connected to it all the time. It does all the data software and all the computation. Students will interact with the software running on that computer, not the fridge itself. There are ways to achieve this, but it depends on how close/far you are. When a cooldown is going the computer will capture and store the information.
            - [Students] We need to know the formatting for that data.
            - [Cirqus] We can provide sample data. But one way to achieve this is to just interact with the log files and not the API itself. But that's just a suggestion which has advantages and disadvantages, and the students should figure this out. We should ask bluefors if there is a sandbox for their fridge control software. And bluefors may be interested in the solution that the students build. But bluefors can also help with testing.

- [Students] Python is the backend?
- [Cirqus] That is the suggestion, but other options are available if they can be justified but the understanding is that the end of semester will result in code being handed over to cirqus and currently other systems and software utilise python which ensure that we can maintain it.
- [Students] But most will be in JS for the web component and quite separated from the computer connected to the fridge?
- [Cirqus] It's not so much that it's running on the computer that's the issue but its whether or not we can have people interacting with it. But for web interface JS is likely the most sensible option but there are no hard opinions for it from Cirqus. There could be a way to do most of the work on the computer and part of it done elsewhere but this is more of a design consideration. It is not an assumption to be made but part of the solution design.
- Agreement: Design proposals from the students and then decide which one is more optimal or perhaps how to go forward. Cirqus will also work out how to set up a delineation between the two teams.
  - [Students] But perhaps there are two mini projects that we can do?
  - [Cirqus] We definitely don't think splitting it in terms of gas and pressure management is a good idea but working out a way to split the work will need to be thought of internally for Cirqus. To be done by the end of the week. Students and CirQus need to agree on how to split the work and move forward with the project. But the default is to have two options of the same thing, and this is not the initial idea, but this is a safe option.
  - [Students and Cirqus] we assume that it's the same project.

- [Students] Where to host it, perhaps at something like cirqus.uts.edu.au?
  - [Cirqus] We don't yet have something set up but we may need to interact with IT to discover the best way forward. Perhaps this is something students can do
  - [Students] Yes!
- [Students] How to send data from the fridge to the computer and have that interaction?
  - [Cirqus] Those computers have software that implemented the safety for those fridges. Ideally, we should not add additional software and computational complexity to the computer and utilize its resources. So, things should be kept separate from the monitoring computer.
- Do we work on the same or separate features?
  - Agreement: Work as we are working on the same or complementary features and work out later how the workload would be split. It encourages collaborative teamwork and also compatibility of the software components.
- [Cirqus] Perhaps separate meetings per group might be more manageable?
  - [Students] Yes, it would be a nice idea, but scheduling is to be discussed.
- [Students] Consistent meeting time?
  - [Cirqus] Yes!
  - [Students] Does now work?
  - [Cirqus] Yes but the time needs to be managed properly. But this can be a later problem once we finalise how we split the work across the 2 fridge teams.

Outstanding Questions to be Answered

- How do you connect to the system?
- TCP/IP (Most likely) or other (serial?)
- Is the device accessible over the public internet
- If not, we will need to set up a VPN as much of the development will take place off-campus
- Do you have a test setup we can talk to or only a working (production) system
- Is this accessible now ? If not, when?
- Do you have any self-developed software which talks to this API (This would be a great starting point)
- Do you have a rough design (sketch) of what you would like to see on the web portal
- Or at least a list with how you would like everything to look
- Does it need some kind of user control (passwords etc.)
- It asks for predictions, should it be predictive of future trends based on current data to predict problems?
- What do you mean by customisable graphs (as in apply filters? Or select data to graph?)
- What is a data archive? What is the structure? How is it being stored? How will we access it?
- How is the real-time data being captured presently? How is it being accessed? Where does the data go?
- What kind of push notification? Would an email alert or text message be enough?
- What do you define as a "critical status"
- In the context of real-time viewing how often is real time e.g. every second, minute?
- What are the core features you would expect as an operational prototype?

March 15th

Summary

- Students should decide where the data from the fridge should be stored as part of the project development. Currently the log files are created daily and sampled every minute but it would be worth being able to view the changes in the data in 5 minute intervals.
- The need for a password to access the site will depended on how the site is implemented. As a baseline, the site should not be Google searchable but quick and easy to access. This may require interacting with IT to determine how we can leverage existing security practices (like single sign on) to make the site secure. CirQuS group will help to set up line of communication between students and IT.
- The fridges are connected to the internet and can be remote accessed via Remote Desktop but this has limitations. It would be ideal if the web page can allow the CirQuS team to issue basic commands to the fridge in the event that something has occurred but no CirQuS team member is on-site to resolve it immediately. This need not necessarily be implemented but the solution should be architected in a way such that this feature is possible. Likewise the ability to reboot the monitoring tool itself is not necessary but nice to have as a non-critical feature of the solution.

- In terms of producing graphs and charts, scatter plots and line charts will be most common. CirQuS group would like the ability to display historical data and specify the time frame the data should be displayed from.
- Some of the parameters that may be relevant to the CirQuS team but are not available in the data logs include the temperature, gauge pressure, power applied to the fridge stages, flow rate, snapshot of what the front panel of the fridge looks like, turbo speed and power.
- The ability for the solution to be platform agnostic and adaptable to different devices is important and should be considered as part of the functional requirements. The same is true for the solution modularity.
- Documentation will be developed progressively throughout the project.

Action items

- CirQuS to share data logs
- CirQuS to connect students with IT contact to discuss security of solution and how it may affect implementation.
- Students to prepare basic architecture of solution.
- Students to prepare project scope document for sign off between CirQuS and students.

Full minutes

- [Students] We have user stories for the different features that should be completed but a question that we have is where is the data being stored?
  - [Cirqus] Working out where the data is being stored is part of the project scope but currently it is being stored on the local fridge laptop. The backup of the data is the responsibility of the university. There are specifics when it comes to backup the data. The data should not be constantly being sent back and forth between the point of computation or analysis and the fridge computer. We want to avoid using laptop power to have constant communication of data between source and destination.
  - [Cirqus] Log files are created on a daily basis. A possible solution is to transfer the log file at the end of each day to its long term storage place but the data for the last 24 hours may be accessible over the network. This can limit the amount of data that is readily required.
  - [Cirqus] The data is sampled every minute.
  - [Students] Should we update the data at the same rate for the graphs?
    - [Cirqus] As quick as possible might be nice but it strictly being every minute is not required. At least updating the web page every 5 minutes would be nice to ensure its kept up to date. 25 minutes as a delay time between the data and the web page is too much.
    - [Cirqus] A potential idea is to get the data every 5 minutes as a standard and then change it to 1 minute cycles if it passes through a particular flag or something for an incident.
    - [Cirqus] Alternatively you can read the last snapshot of data to see what has the data has changed rather than getting the whole data set each time. This means you can get the data from the fridge computer when required.

- [Students] Can we have a sample of the data logs?
  - [Cirqus] Yes its being uploaded to sharepoint at the moment
  - [Students] How sensitive is the data? Does it need to be protected in transit? Does the website need to be protected by password?
    - [Cirqus] The password to the website will depend on implementation. For example if it is behind the UTS sign in then it doesn't need additional security protection. We don't want the website to be Google searchable. Depending on the implementation of the front-end we can discuss the level of security necessary. Ideally we want to just be able to open the web app and see the data as we need like during a meeting.
    - [Students] Single sign on requires us to liase with UTS but who would be contact?
    - [Cirqus] It depends but we can put it behind another serivce already associated with single sign on. I need to communicate with someone from Eresearch regarding this project as a proof of concept so we can set up a pathway to deploy the website. This is something which I [redacted] will do. Once this has been set up then I will connect them with one of the people from the student team. <name> is the contact.
- [Students] We are going to communicate with the fridge laptop in some way and this requires us to connect to it but how does this work?
  - [Cirqus] The laptops are internet connected
  - [Students] Is there a proxy? Is it externally accessible?
  - [Cirqus] Remote desktop to the laptop is possible but there are limits. E.g. Teamviewer is not an option for controlling the laptop. But as an interim solution we can have another computer in the lab to communicate with the fridge laptop.
  - [Students] That is similar to the solution we proposed using a raspberry pi.
  - [Cirqus] We don't have a raspberry pi but in the lab we have 3 rooms. The first room is the control room, the next room is the fridge room and the last room is the backroom where the fridge control units are. In the control room we have computer which will display what is going on in the fridges. The fridge computers are on the UTS network and are network accessible. The API for the fridge control software has been shared with you, and we have used this in the pass to implement controls on the fridge. The API allows you to send commands through a web socket or a HTTP address to the fridge software, and the fridge can implement that request whilst running the rest of its commands. It's possible to interact with the fridge without interrupting its operation.
  - [Students] We want to be hands off as possible from the fridge as long as we can regularly access the fridge data.
  - [Cirqus] Thats understandable but no one on our team who is close to campus so we would want to issue limited commands to the fridge through the interface in the event that something happens and we need to make changes. We want a real time status on the fridge like what valves are open, so it would be good to have an option to have the ability to control the fridge. For instance if something is happening then we want to control the fridge but this shouldn't be available all the time.

    - [Students] Is this an optional or additional thing to be completed after monitoring?
    - [Cirqus] If the design is architected in a way that makes that possible and allows us to add control options to it in the future. It should be a design consideration even if it is not completed. We want a product that has useful functionality but it doesn't necessarily need to do everything. We should tick things off in an order that enables a functional product at each stage, the product should have opportunities for expansion.
- [Students] How far back in the data should trends consider?
    - [Cirqus] We want the ability to look back at all the data to see the trend holistically but realistically we wouldn't be doing that regularly. We want to be able to pick the timeframe for the data to be presented on the charts.
- [Students] We have put scaling as a non-functional item and a secondary focus after we have implemented the functional thing.
    - [Cirqus] Platform independence or agnostic is important but we want this to be a functional item just to avoid discomfort of viewing it on a phone for MVP.
- [Students] Is modularity a functional thing as well?
    - [Cirqus] Yes
- [Students] We had an idea for a reboot function for the monitoring tool if it stops working. We have listed it as non-functional.
    - [Cirqus] This is not a bad idea but I agree its non-functional.
- [Students] Documentation is the final item that we plan to do at the end of the development or close to.
    - [Cirqus] True but building documentation along the way would make it easier for you so perhaps maybe we should do it along the way. But yes having documentation is important and valuable. If there are elements of the solution that are not written in python then it would be worth while to ensure that the code is well written and self-documenting but also modularising the code. So separating the front end from the back end and separating the responsibilitites but also separating the python from non-python. This is just to help with future maintenance and expansion.
    - [Students] We will likely structure it like this as well.
- [Students] So onto the architecture. We should have a python backend to get the data and do the analysis and then use javascript to write the front-end for the web app. But we need a little more information to dig into the semantics which is good since we have now that information available and can prepare that for next meeting.
    - [Cirqus] Good to hear!
- [Students] Are there parameters that are not in the data logs that you would want?
    - [Cirqus] Yes there are a lot of other parameters. Some of those include, temperature, gauge pressures, power applied to the fridge stages, status type items like flow rate, snapshot of what the front panel of the fridge looks like (this is binary like is this valve open or close), turbo speeds and power,
    - [Students] What types of graph do you want for this? Line charts? Pie charts?
    - [Cirqus] Scatterplots with lines but other types of visualisation could also be nice for specific contexts but scatterplots are a good start?

3.12.2 Retrospective

Sprint 0 Retrospective

Date: 22/3/23

Worked Well	To Improve	Other Comments
• Communicating as a group	• We can all be more vocal/don't be shy	• We should define class goals each lesson and check we complete them AND define actions for next class
• Collaborating and discussing ideas	• Didn't feel like there was much for us to actually do	• How much of our project can be open source
• Prepared for meetings well	• A bit unorganized at the start	
• Everyone was active to share ideas and problem solve	• Finalise a specific tech stack	
• Communicating as a group	• Project scope	
• User stories	• Recording and allocating work, could use Jira board more	
	• Taking initiative even if it's not your expertise	

Action Points:

- Run ice-breaker activities for foster a better group dynamic
  - Start with Two Truths and One Lie this week
- As a group, review the project scope document, add out-of-scope, and review example confluence layout to create a cleaner confluence which can be handed over to CirQuS at the end of the semester
- Decide on a tech stack by the end of today to enable allocation of tasks
- Work on the specific action points listed above today, and spend time developing actions for next week.
  - Going forward, develop action points for the following week
- Spend time allocating tasks/dividing the project up for people to start working on

Other Points:

- Talk about using open-source software for the project have been shelved, as they don't appear to be in the spirit of the subject

3.13	SPRINT 1 ARTEFACTS	3.13.1 Meeting Minutes March 22nd • Summary • Minutes • Action Points: • Development: **Summary** • Cirqus to sign off on project scope that was defined by students. • Most f the charts that will be generated will plot some parameter against time. Students should consider working in the ability to compare multiple parameters at once and having more than y-axis to show the values for the parameters. • Students should also consider the ability to generate multiple graphs based on multiple parameters being selected at the same time. i.e. select 3 parameters and it creates 3 graphs • The graphs should factor in the ability to convert between log and linear axis values • The ability for the monitoring software to connect to other devices will be factored into the design and the scope of the project but made of lower priority as it is not mission critical. • Cirqus will specify which parameters from the data logs are relevant and important, they will also specify the units of the data. Information regarding how to read the data logs is captured in the manual

- The conversation surrounding hosting and security of the monitoring software is in discussion between IT and CirQuS, and studens will be updated of any news.
- CirQuS are working on getting the students into the lab for next week meeting but if no confirmation then assume it is online.
- There is no preference for what database but one is likely required.

Minutes

- CirQuS needs to sign off on the scope that the student team has decided for this semester. The scope factors in what is in scope for the project and what is out of scope.
    - [CirQuS] 1st and 3rd dot points are the same?
    - [Students] Sort of but the first one is defining the need for notification and the second is what data
- [Students] Clarification of data and graphing. Should there be templates like temp vs time?
    - [Cirqus] Most graphs will be against time. For customisability it can be like time range, x axis range or y axis range, or what we see on what plot.
    - [Students] Like how far back?
        - [Cirqus] Yes
    - [Cirqus] Maybe also plotting multiple parameters against time with 2 y-axis for the two parameters against time
    - [Cirqus] Maybe when having 3 parameters or more we can have multiple graphs instead of plotting it onto the same chart.
    - [Cirqus] We also want to have the ability to change between log and linear values
- [Cirqus] We are working on lab access for next week to have the meeting there.
- [Cirqus] There are things that you can do in Blufors that is cool like scrolling along one axis or another instead of having to scroll a lot. This would be a nice feature to have but it is not critical to have.

- [Students] We also have identified out of scope things as well but we plan to design for these items just not developing or implementing it. An example is to have other devices connect to the monitoring software.
  - [Cirqus] Is there a reason for that?
  - [Students] We are concerned about the time but we can make it so that the solution is set up so it is easy to implement.
  - [Cirqus] We shouldn't put this out of scope but rather at a lower priority. To implement it we would need to allow the server to ping the blufors api.
  - **Agreement**: The ability to connect to other devices will be made of lower priority rather than out of scope.
- [Students] We have the sample data now, which is great but we would love to have documentation on how to read the logs and what it is about.
  - [Cirqus] We will give you a list of things that we want and the exact keys are defined in a table in the manual. We will also provide units for each of the data category.
- [Students] We have some access issues with GitLab could you help
  - [Cirqus] Yep! There are some weird access issues outside of the uni which you may need help with.
  - [Students] Yep we have discovered them so we should be able to work around it now.
- [Students] We want to outline our architecture for you but we plan to remove majority of the processing from the fridge computer and manage majority of it on another device (raspberry pi) to avoid overloading the fridge computer resources.
  - [Cirqus] We had a meeting with the uni regarding hosting but we will keep you updated on that as we learn more.
- [Cirqus] We will give you a list of everything we want to plot + units for each data item in the logs. Do you plan to copy the log to the raspberry pi or move to a database?
  - [Students] We plan to migrate to the raspberry pi and then move to a database for ease of use.
  - [Cirqus] Any particular database?
  - [Students] Not yet decided but any preferences?
  - [Cirqus] No preferences 🙂
- [Cirqus] Assume the meeting will be via teams if we can't arrange an in person meeting.

Action Points:

- Everyone ensures they can push and pull from the GitLab
- Ensure everyone has the tools (Javascript, VSCode, React, Express, Python, etc.)
- Liam and Braden follow up about SSO Login
- Set up web application -
- Set up database -
- Create fridge objects/chart objects that store locally -
- Taking data from the database (and generating graphs with it) -
- Continue working on the UI/UX
- Create tasks for user stories and other set up tasks
- Define branching structure + branch names
- Workflow
  - 2 reviewers before merge

Development:

- VSCode has been decided as the IDE
- Will use Javascript, React, Python, Express
- Use Recharts JavaScript library to generate graphs
- Data should be stored in a database (off the laptop that controls the fridge) to allow for historic data to be displayed
- Modular
  - Fridge object type
  - Chart object type
    - Variables of X, Y, Scale, History
    - Notation type - do they have a preference? or should we allow them to swap between scientific and every-day?
    - Colour x, colour y
    - These variables should be stored locally, so graphs can be generated dynamically each time the page is loaded
- [redacted] to follow up at SSO Login with [redacted] and IT
  - Developers don't worry about login for now, to be implemented later depending on outcome of meetings
- Notification settings
  - Will they set these in the database? or from the website itself?
    - This will depend upon the security of the webpage, and who is able to access it
- Database - Postgres

March 29th

[redacted]

Checklist items

- Everyone ensures they can push and pull from the GitLab
- Ensure everyone has the tools (Javascript, VSCode, React, Express, Python, etc.)
- [redacted] follow up about SSO Login
- Set up web application - [redacted]
- Set up database - [redacted]
- Create fridge objects/chart objects that store locally -
- Taking data from the database (and generating graphs with it) -
- Continue working on the UI/UX
- Create tasks for user stories and other set up tasks
- Define branching structure + branch names
- Workflow
  - 2 reviewers before merge

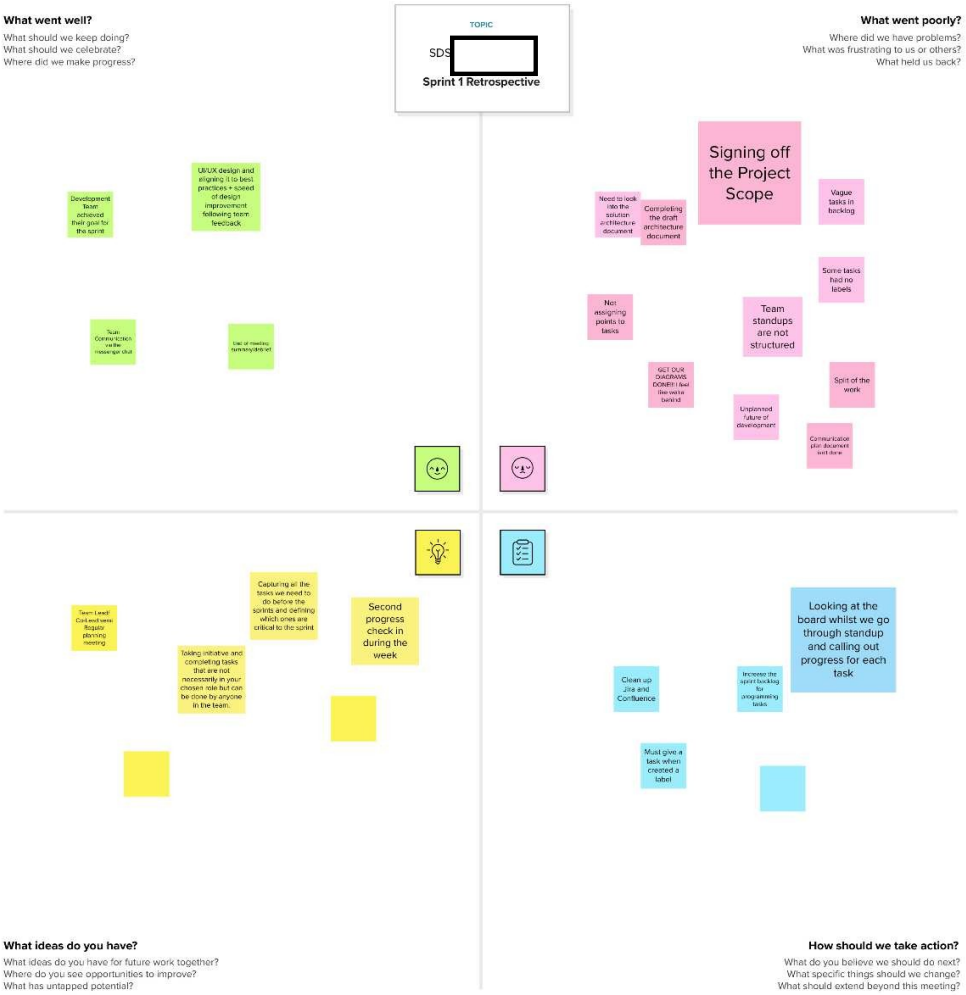
Notes

- Should add in fridge control panel to the web interface, and mark up which valves are open etc.
- Presentation from e-research
  - Lab network is high risk if compromised, client network (where we will put our dashboard) will be low

- f5 VPN is used to encrypt and decrypt traffic between lab computer and client device. It is also used for authentication.
- potential solution - limited access through aws
  - lab network - high risk
    - lab computer
  - aws via direct connect - medium risk
    - web container
    - s3 for objects
    - stats can be done through grafana
    - load balancer - managed end point traffic and can have policies to preevnt sql inject csfx etc.
      - removed need for vpn
    - VPC to host it
  - client network with web page - low risk
- UTS IT architecture principles
  - broad
  - not super up to date
  - data management
    - transport
    - risk
  - risk management
  - access management
  - future focused, not over architecting, minimal porting
  - can use service connect to raise tickets for the task and then connect with eresearch or it support
    - eresearch supports research projects
    - mention nathan in the support ticket

- Agreement: Make controlling the fridge a stretch goal rather than removing it from the scope completely.
- [Students] Monitoring through Grafana?
  - [Cirqus] Potentially but not sure
  - [Students] We need to do more research on this
- [Students] Other charts like bar graphs would be out of scope
  - [Cirqus] Yep, we were concerned that it meant that we were cutting out graph features and controls from the solution.
  - [Students] Yep we need to clarify this
  - [Cirqus] There are many different types of scatterplots and they can include plotting different parameters, different scales, etc.
  - [Students] Yep understood
  - [Cirqus] We won't need something like a box and whisker plot
  - [Students] Yep we will clear up the scope points so they are more detailed and thorough
- [Students] Colour scheme and logo?
  - [Cirqus] We do have some logo and colour scheme but open to inspiration. We follow UTS and specific colours like QSI. Maybe we can use the lab colours?
  - [Cirqus] New logo is not in the bandwidth at the moment but can be a side priority.

3.13.2 Sprint Retrospective



3.13.3 Completed Tasks

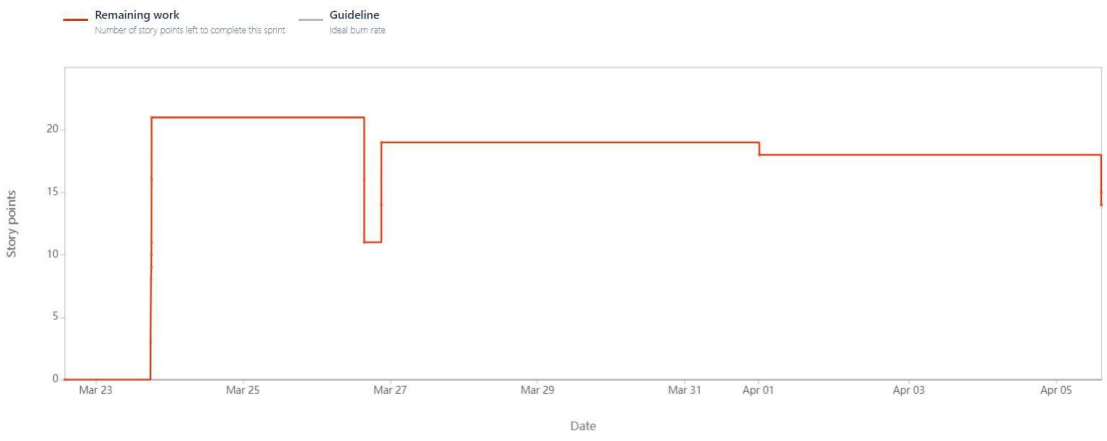
CIR-16	Write acceptance criteria for user stories	Task	DONE	3
CIR-18	Define branching structure and GitLab practices	Task	DONE	1

3.13.4 Incomplete Tasks

CIR-15	Version 1 of UI/UX	Task	DONE	5
CIR-30	Create application logic flowchart/user flow diagram	Task	DONE	3
CIR-31	System architecture diagram	Task	DONE	5

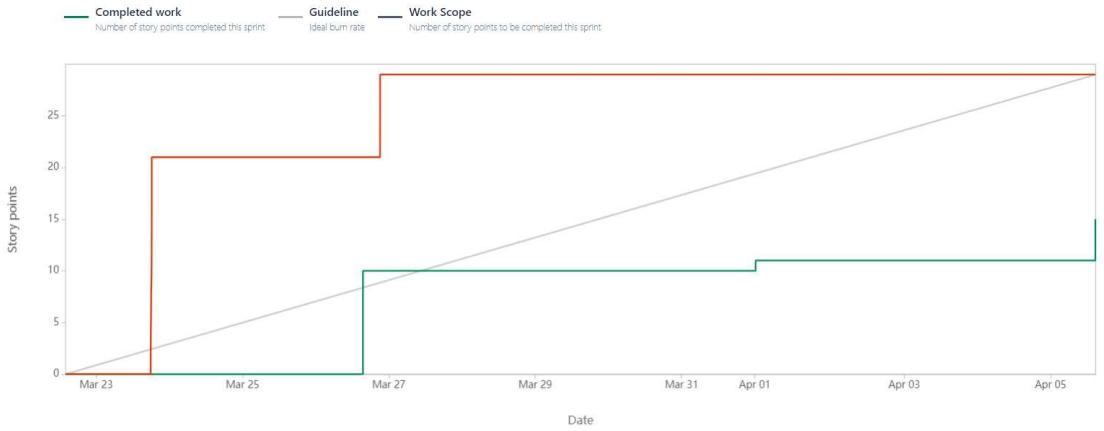
3.13.5 Burndown report

Date - March 22nd, 2023 - April 5th, 2023



3.13.6 Burnup report

Date - March 22nd, 2023 - April 5th, 2023



3.14 SPRINT 2 ARTEFACTS

3.14.1 Meeting Minutes

April 12th

(Raw Meeting Minutes)

- Can we organize a time to meet with the
  - Sure, I will send him and email. (Can you CC me in?) Sure, yes

Do you want the historic data from before we implement the solution? Or can the data start from when we first start recording with the pro


shows off UI

- Do you have an inverted colour option that increases contrast logo that stands out better against the background
  - I don't think we have one. I think it should be fine now. I do have some other comments about the ui
  - Login in (once you do log in) – the dashboard should at a glace have the simple info to have a clear picture of what is going on. Its for both fridges 5 temps, 6 pressures and flow rate – split the page in two and shows both fridges (I will provide a one note about this
  - Under devices (in the sidebar) have Fridge 1 and Fridge 2 where you click to see the control panels for each fridge, should also include all that info from the dashboard for this specific fridge, as well as other info (I will provide the one note)
    - You don't have to go to select the specific device and then go back to dashboard to see that info,
  - Graphs – take up as much horizontal space as possible to stretch them out. Vertical space doesn't matter as much, as long as it looks nice. And the way to create graphs – there aren't that many choices you need to create a graph. The only choice should be the metrics you want to graph. They will be groups (5 temp, 6 pressure). Have a toggle for these fields (check boxes?). And for time range, have today an one week ago as the time range options . On a created graph, if you hover over the x axis and scroll in and out we should be able to zoom in and out of the time. – That is the way we change the x and y limits (clicking and scrolling) – I understand that functionality may not be easy to implement. In the case you are unable to implement this, you would specify the time the way you currently have (Specify a start and end time). Do you have a place where we set the upper and lower limit for the y data?
    - Do you want us to plot all of temp vs time for all valves on one graph?
      - You should choose which temps you want to graph (eg. two of them) and you can see these two over time.
  - All of these comments pertain to the GUI – I am happy with the style of it
    - Yeah, that's fine


shows off development.

- There isn't a lot to look at, but we wanted to show you we are developing stuff
  - I assume the graph is just a placeholder. All my previous comments still stand.

Would we be able to spend some time with you guys and a couple of our devs to try and look at the API and WebSocket stuff?

- Sure. Ping us a couple of days before, or even the day before, and

[redacted] Joins

- We just wanted to finalize the project scope so we can sign off
- We saw your email and have made some changes to reflect that, so have a look.
  - How does the functionality get added to your code? If you have inactive buttons that kinda answers that question. There were two questions, how is the control handled and how would future implementation would be added to your code , and if you have the skeleton structure that kinda answers that.
- If you're happy with that could you sign it off?
  - Do you need a literal signature? Or can I just initial it
- Initials should be fine

Signed document

I just wanted to check something – there is no issue suingthe Q as an icon? I just just the Q for the icon in the UI design where there was not enough room for the full logo. The full logo is still present, but I used the Q where that would not fit

- Q is possibly. There are two things that could be more optimal. I have a shortened version of the logo, a compact version - let me share that in the teams.
- CirQus Logo very small has been sent to you. We have adobe illustrator versions of it as well, that range in smallness. I don't have a png of them, but if you adobe you can view it
- Another option would be to us the name of the lab. Our organization is called CirQuS
- Millikelvin Quantum science lab
- [redacted] do you have a preference?
  - Not really? MQS might look nice, but we do have that Logo
- Was that what you were looking for? A shortened version of the logo

Do you have an icon?

- What is an icon

Kind of a symbol of the group. Think of Google chrome

- It was always my plan to come up with an actual logo logo, like a badge. I just haven't done it yet. I don't think Q is it. I think either having the shortened version of the logo, or the initials of the Lab. I think it's best to go with the shortened version of the logo for now.
- You aren't a graphic designer are you

I have some skill in that area, but I am just worried about the time of the project

- That is ok, it definitely was not in the original scope of the project.

April 19th

(Raw meeting minutes)

Teams?

- We wont always have team to notification on. Email would be priority. Teams would be good if that's possible. We understand there may be some charges with text so that is ok we you don't include it

Can we visit next week?

- Sure. You can also visit some other time too

How hosting this is going to work once we are finish and want to deploy?

- We can host it either on the computer in the lab.

We weren't sure if you want it hosted in the lab or if the uni IT people would host somewhere.

- If it's within the UTS network it should be able to talk to each other. We could test it, but

Don't want to VPN to it

- No. For now we might have to VPN, because getting permissions for wider access would take time. Bureaucracy, might take some time.

Jenni shows off ui

We have updated the dashboard now to swap between fridges at a glance. Devices are listed out on the side for quick access. A quick access option with fridge, parameter, title creates graph and lets you switch between location and what you are displaying. Previously created lets you edit previously created one. Operating limits let you set the range where operating is healthy

- Operating limits. When it's cold it's going to be cold and when it's hot it's gonna get hot. Can we turn off the limits when we are heating up and cooling down.
- In the one-note we shard there were a bunch of fields we want checked. We want operating limits set for all those fields. Not just temp, pressure and flow, but all the fields listed in the one-note.

You had a note on at a glace

- The temp, pressure and flowrate, do they take you somewhere else? Are they buttons?

No they just show you the current temp, etc.

- There are multiple values for temp for the different stages. There are also multiple pressures. On the flow map, you could display the pressure information on there for where the pressure is taken. The temp doesn't have a place in the map, but you could find a place above it or something to show it

Question about the valve map. Do you want us to use our clean created one? Or use the picture that you provided us (Blue force).

- I don't really mind, whatever is easiest for you guys.

And we also changed the logo to the compact CirQus logo

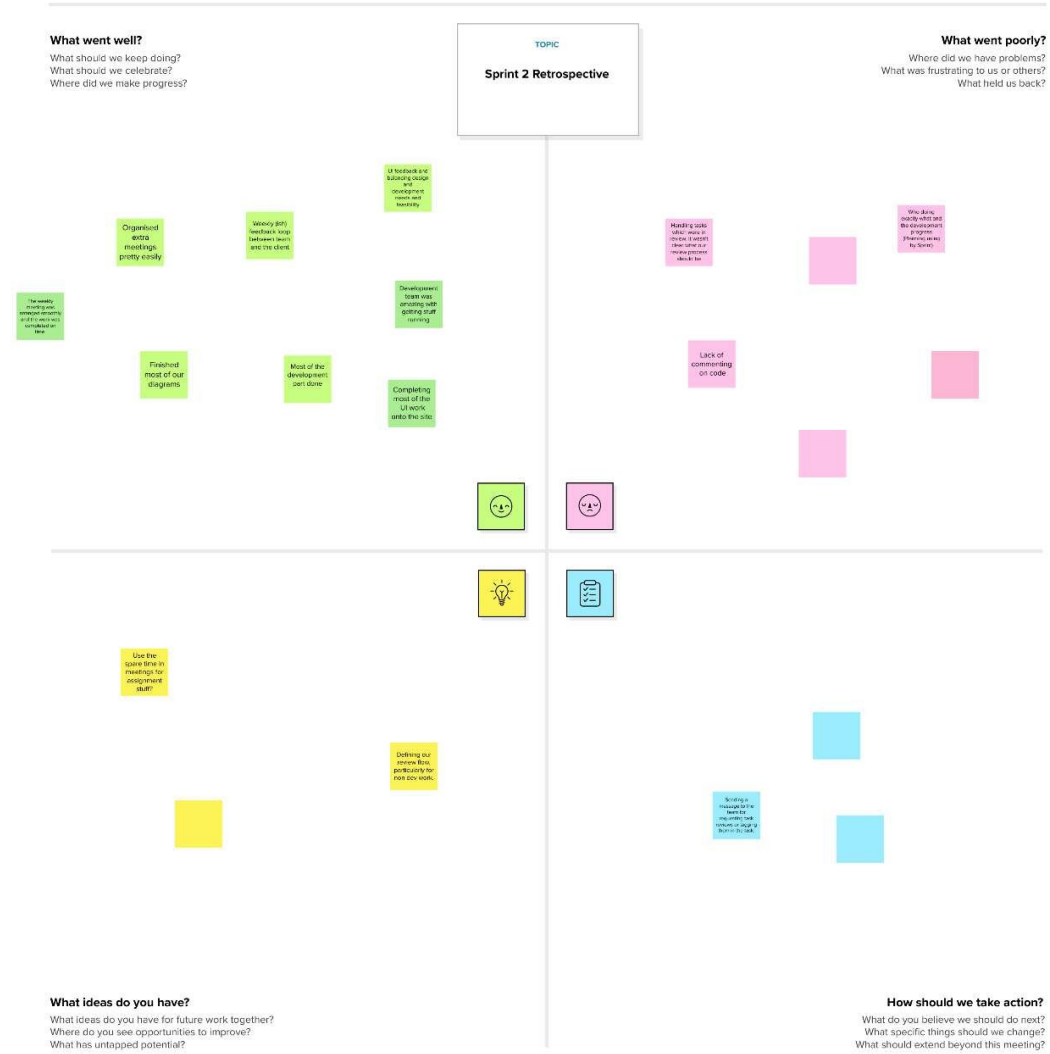
Angus shows dev

This is showing the graph with data coming in every second. OF course yours would be every minute, but this is just to show real time data. But also you can see the scroll and zooming and scalling on the graph, as well as the axises.

- This looks really good. I'm happy with that, no comments

Ping us on teams for that meeting in the lab. If we don't hear from you we will assume next Wednesday in place of our regular meeting.

3.14.2 Retrospective



What challenges have we faced? How did we deal with them?

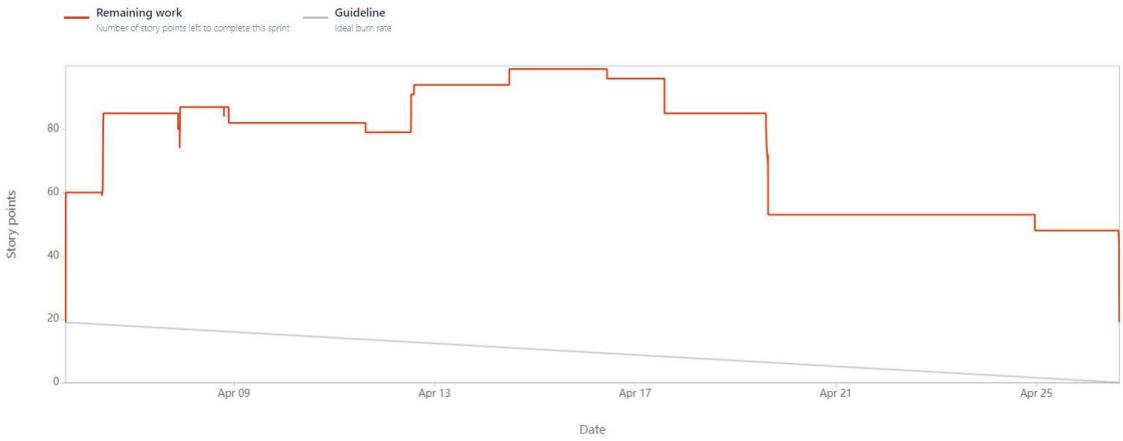
Balancing design principles with development requirements and technical feasibility of implementing said designs.	Researching more into best practices as well as the methods of implementation for the particular language. Additionally getting feedback from the development team about designs.
Was unsure of the proper implementation of conditional rendering to avoid repeating code.	Asked for assistance and learned about how to properly toggle the state
Back and forth with Nathan over what's in scope	came to a compromise solution

3.14.3 Completed Tasks

CIR-47	UI/UX re-brand	Task	DONE	8
CIR-48	Notification page re-brand	Task	DONE	3
CIR-49	Account settings re-brand	Task	DONE	3
CIR-50	System settings page re-brand	Task	DONE	0
CIR-51	Create graph error pop up	Task	DONE	3
CIR-52	Edit graph and edit graph error pop up	Task	DONE	3
CIR-53	Historical graphs page re-brand	Task	DONE	5
CIR-15	Version 1 of UI/UX	Task	DONE	5
CIR-31	System architecture diagram	Task	DONE	5
CIR-30	Create application logic flowchart/user flow diagram	Task	DONE	3
CIR-64	Add new fridge page	Task	DONE	5
CIR-65	Detailed fridge device page	Task	DONE	5
CIR-69	Use case diagram for logout	Task	DONE	3

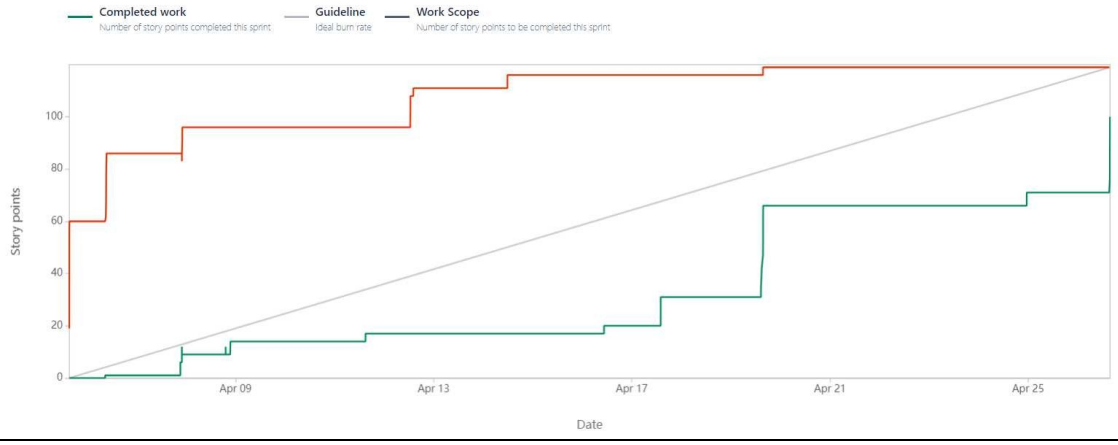
3.14.4 Burndown report

Date - April 5th, 2023 - April 26th, 2023



3.14.5 Burnup report

Date - April 5th, 2023 - April 26th, 2023



3.15	SPRINT 3 ARTEFACTS	3.15.1 Meeting Minutes May 3rd  • There is a firewall that is blocking deployment of the solution. So we currently plan to write out documentation to future proof the solution in the event that we can't get IT to support us in deploying it. Does that sound good? ◦ [redacted] Sounds good but can be designed it as if it were to connect? ◦ [redacted] – Yep that's the plan. ◦ [redacted] Yep I assume that it isn't all that different in implementation. ◦ [redacted] – Yep so the implementation and product wouldn't change but we just wanted to make sure that its understood that we will be documenting how to connect to the fridge as if there was no firewall issue • [redacted] – For the email notifications, if it exceeds a certain number then there is a payment associated with it. ◦ [redacted] – I didn't know that, what service is that? ◦ [redacted] – EmailJS ◦ [redacted] – Do you know any work arounds Giorge? ◦ [redacted] – Nup ◦ [redacted] – What does EmailJS provide specifically? ◦ [redacted] – It just sends the email securely. ◦ [redacted] – Are there any ways to send notifications without paying for a service? Like using MS Teams

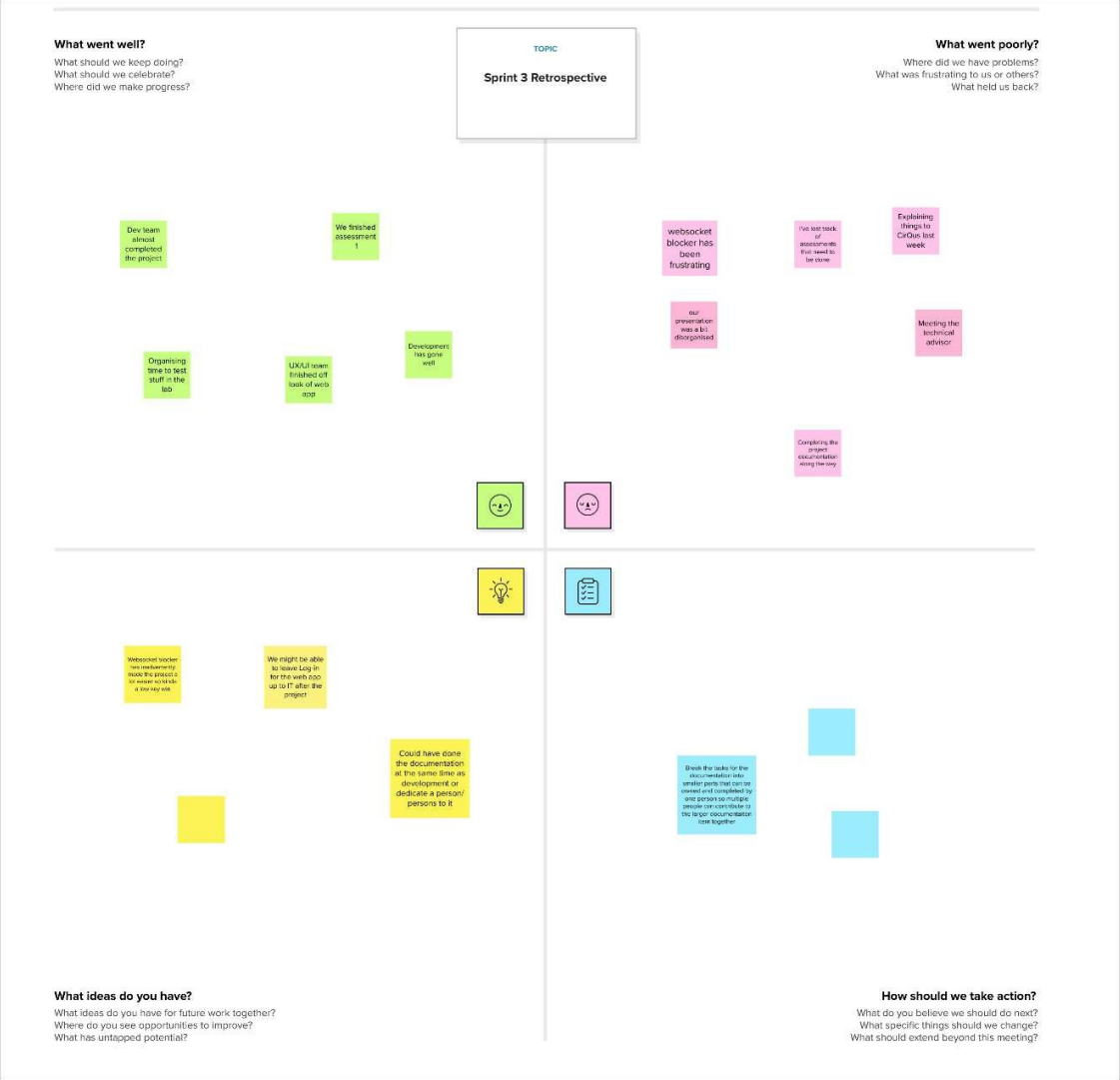
- [redacted] – It should be free yes
- [redacted] – What are you basing that on? Between now and next week, could you find a service that is [redacted] understand how it works? We will also let [redacted] know and see what he thinks
- [redacted] this is just something we wanted to raise with you.
- [redacted] – Just to clarify the service is technically free if you send less than 200 per month.
- [redacted] – We hope that the machines don't break but we would exceed it over 200 per month.
- [redacted] – Do we know how much it costs?
- [redacted] – $9 for 2000 per month. We will also share the link to it
- [redacted] We also will investigate teams notifications

- [redacted] – About hosting the site, we understood that it should be internet accessible but this requires a domain to be set up on one of the machines in the lab. So we wanted to understand how possible it is to set up a domain and if that's possible.
  - [redacted] – Yeah we need to check with Nathan
  - [redacted] It might also be connected to IT
  - [redacted] – I agree so
  - [redacted] – I also thought about hosting it on the windows machine but it would need open ports to the internet to do that. And then we could change some things on the machine so that it would be accessible but if we can't open the ports then it would be difficult to connect to things
  - [redacted] – We will relay that to Nathan
- [redacted] – Single sign on, we got that email about the integration of the SSO but we can't implement that until the web app is hosted and has a domain as it needs a domain to redirect to. So currently we cant integrate that until the very end of the project. It looks promising though.
  - [redacted] We can get a domain ASAP to help things along
    - [redacted] – Also important we can open ports on the machines so we can connect to it.
    - [redacted] – I have emailed people on the firewall stuff for the lab computers and am waiting on a response.

Action Items

- CirQuS to follow up with IT about the computer firewall
- Team to explore different email notification options and investigate MS Teams as an option.

3.15.2 Retrospective



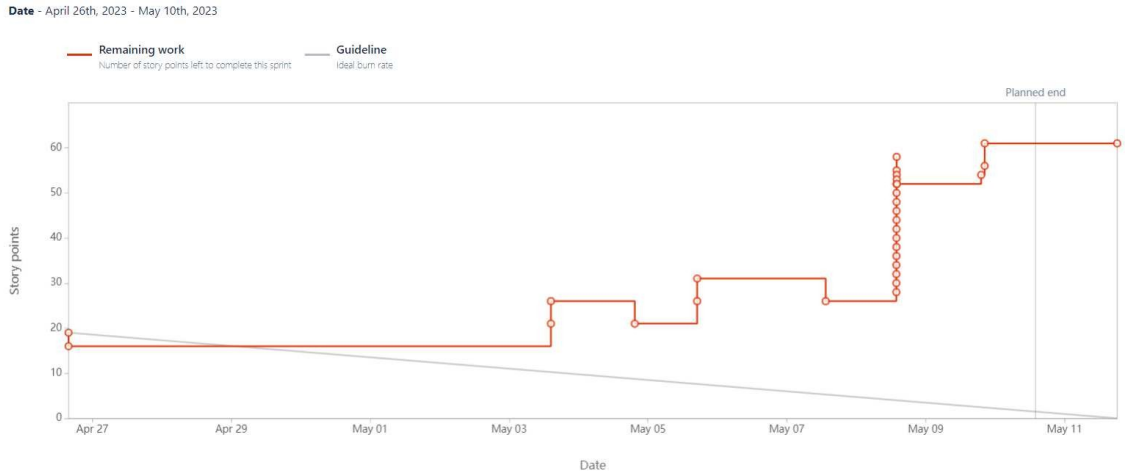
3.15.3 Completed Tasks

CIR-103	Mobile view for UI	Task	DONE	1
CIR-104	User guide headings and tickets	Task	DONE	1

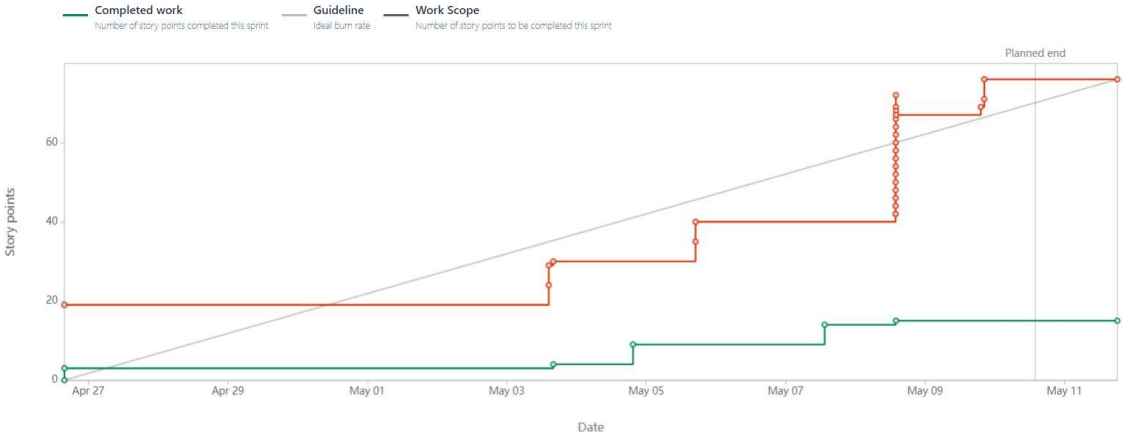
3.15.4 Incomplete Tasks

CIR-109	User guide: Introduction	Task	IN PROGRESS	2
CIR-111	User guide: Acronyms used and meanings	Task	IN PROGRESS	1
CIR-113	User guide: Overview of the system	Task	IN PROGRESS	2
CIR-114	User guide: Dashboard page	Task	IN PROGRESS	2
CIR-122	User guide: Error messages	Task	IN PROGRESS	8
CIR-125	User guide: Menu Bar	Task	IN PROGRESS	2

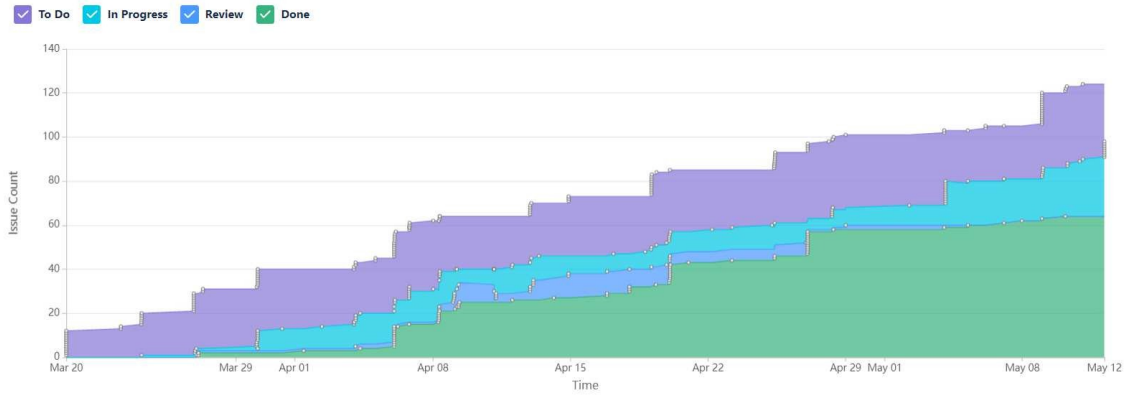
3.15.5 Burndown report



3.15.6 Burnup report



3.16 CUMULATIVE FLOW



4 REFERENCES

[1] K. Beck et al. (). Manifesto for Agile Software Development. Available: https://agilemanifesto.org/.