Pinch Analysis as a Modular Tool for the Analysis and Optimization of Industrial Chemical Processes

Ethan David Francis MacLeod [1, a]

[1]Hamilton, New Zealand

[a]ethandavidfrancis@gmail.com

Abstract
The industrial process heat sector in New Zealand is a significant contributor to the country's greenhouse gas emissions, primarily due to its reliance on non-sustainable energy sources. As part of the nation's efforts to achieve net-zero emissions by 2050, optimizing energy use in this sector has become a critical focus for reducing carbon output and addressing the challenges posed by aging infrastructure. One effective method for optimizing energy efficiency in industrial processes is Pinch Analysis, a widely used technique for identifying opportunities to minimize energy consumption and maximize heat recovery. Existing software tools that perform Pinch Analysis, though useful for reducing energy usage, are outdated and suffer from significant usability and design issues, highlighting the need for improved software solutions. This research journal demonstrates the processes used to evaluate and improve on a new Pinch Analysis module being developed, from wire-frame testing to usability testing. The findings from these evaluations are presented, along with the final system that benefitted from this testing.

Keywords: Usability, Evaluation, Pinch Analysis

1. Introduction

The New Zealand government is committed to achieving net-zero Greenhouse Gas (GHG) emissions by the year 2050 as a part of its broader climate strategy [1]. This initiative is driven by the need to address climate change and take steps towards a more sustainable future for New Zealand and the world. To achieve this goal, specific sectors have been identified to reduce GHG emissions, one of these sectors is the energy sector, and more specifically, the industrial process heat sector.

Carbon emissions in the industrial process heat sector are a major contributor to New Zealand's net GHG emissions. In the context of industrial sectors, process heat refers to the heat required for warming and heating, usually in the form of steam, hot water, or hot gases. The primary issue with process heat generation is its reliance on fossil fuels - roughly 55% [2] of the total process heat in New Zealand is generated using fossil fuels and unsustainable energy sources - mostly gas and coal. This accounts for 28% [3] of the total GHG emissions from the energy sector annually. In 2016, the sector produced 31.3 million tonnes of emissions [3], highlighting the significant opportunity for reducing emissions in this area. The New Zealand Ministry for the Environment released a set of policies in 2023 outlining the requirements for air discharge when generating industrial heat [4], demonstrating the importance they place on reduction in this sector.

Some of the challenges faced when discussing how to reduce GHG emissions in the industrial heat sector stem from the dated nature of New Zealand's industrial infrastructure. Most of this infrastructure was built multiple decades ago, which presents an issue in its compatibility with modern, more efficient technologies. Additionally, the cost of replacing this old infrastructure poses a significant monetary barrier for most companies, this including the cost of the equipment as well at the lost profits associated with downtime. Addressing this particular issue is difficult, and so alternative options need to be explored. One viable approach is to optimize existing systems to reduce energy consumption and, consequently, GHG emissions.

The current tools that provide Pinch Analysis capability are generally outdated and lack many of the usability and software practices that are required for a modern software system. The motivation behind this project is to evaluate and improve on one of these systems to ensure usability and user-centered design is incorporated. This will serve to make these systems more easy to use and more accessible to the process engineering industry.

2. Experimental procedure

To align with an iterative, agile design style the first step required was wire-framing. In contrast to the other two phases for the front-end work, the wire-framing takes a very short amount of time but comes with the caveat of a low fidelity model without any functionality to examine. This made it perfect for weekly design cycles, and composed the bulk of all design work for this project. Most of the testing phase was also conducted on these wire-frames, as specific requirements and details could quickly be drafted for approval and feedback.

The format of wire-frames for this project was hand-drawn rough sketches of the system. For components and sections with more detail, these would be extracted from the drawing and another design iteration would be conducted on just the singular component.

The specific wire-framing workflow starts with a rough drawing of the user interface for what is currently being worked on. This can be the whole page, specific components, or groups of components. The most common example of this would be a design for a web-page, with separate wire-frames for components that are too intricate to accurately represent with a single wire-frame. Next, a series of tests would be conducted on the wire-frames, which is further explored later. From the results of this testing, one of three things occurred:

- Component Changes - If test participants indicated that the broad drawing was acceptable, however certain parts need more work, these parts were extracted into their own wire-frame and the workflow was restarted with these as the focus point.

- Layout Changes - If the general layout of the wire-frame was lacking, the work-flow re-entered another design cycle aimed at exploring ideas from the participants or otherwise overhaul the design as a whole.

- Passed Testing - if no further valuable feedback was gained from the testing, the design was considered ready for prototyping, and prototyping was then undertaken.

Some examples of wire-frames that were undertaken for this project are included below as Fig. 1 From these examples, the low quality nature of the wire-frames can be seen. These wire-frames provide just enough detail for testing to be conducted on them, without sacrificing any needless time to aesthetics.

These wire-framing design cycles were continued up until the conclusion of this project and provided invaluable insight into the direction of the User Interface. Undertaking this wire-framing process allowed for rapid prototyping and design iterations, ensuring that feedback from users and stakeholders could be quickly incorporated. This iterative approach also helped identify potential usability issues early, which expedited development and ensured that the final product was both usable and met the project's objectives. Overall, wire-framing was a crucial tool in refining the interface to meet both functional and aesthetic requirements.

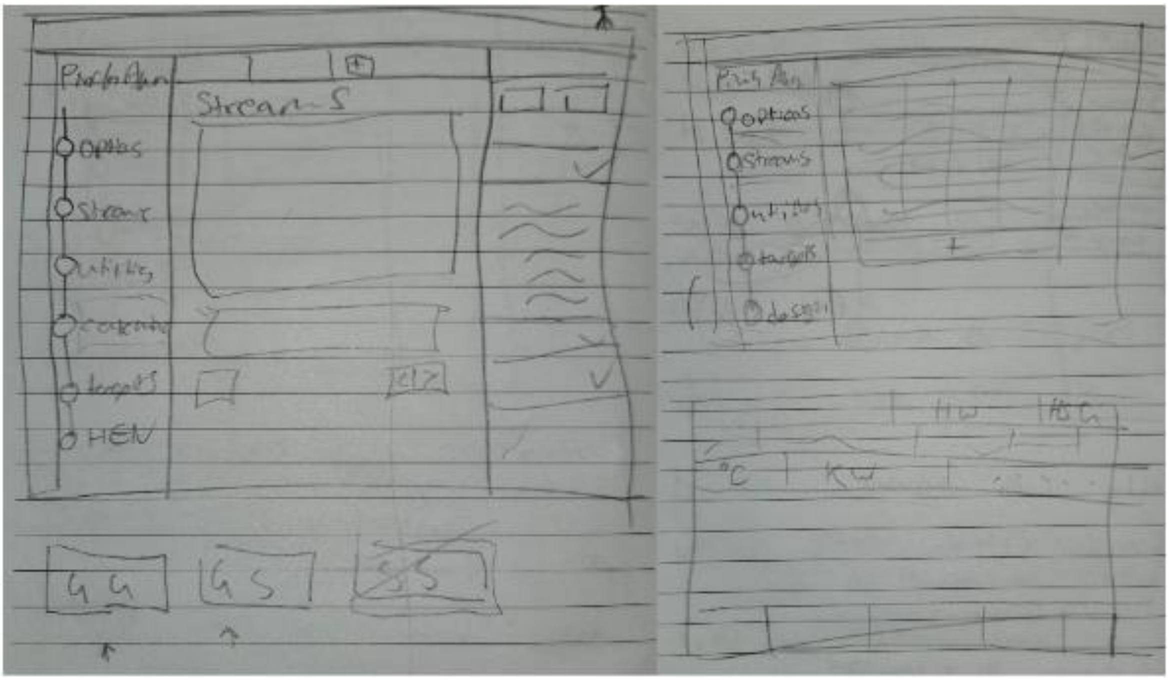

Figure 1 - Wireframing Example

When designing the user interface for a complex system such as this one, there is a need to be able to test the functionality as well as the general design - this is done through prototyping. This allows user to indicate problems beyond visual issues, such as the system workflow and interactions between individual components. The prototyping for this project was done through the online tool Figma. Figma is a web-based collaborative tool for building complex prototypes and includes features that allow for complex functionality to be added to designs. The reasons Figma was chosen for the prototyping for this project are outlined as follows:

- Existing Designs - The prototyping that was completed previously for the Ahuora Digital Twin Platform was already done through Figma. As the final goal for this project was integration with the Ahuora Platform, many of the existing components and layouts were required for the design of the Pinch Analysis module. By reusing common components, prototyping time is reduced drastically, and there is a consistent basis across the prototypes that allows for smooth visual integration between the two. In addition to this, as future design and prototyping work is completed on the Ahuora Platform, these changes can be immediately reflected in the Pinch Analysis module design through Figma's shared components system - ensuring there is no redundant work being done.

- Cloud Serving - Another function that was found to be immensely helpful during this project was the online nature of Figma. Being able to show stakeholders real-time updates to the prototyping was invaluable for communication and transparency. During testing, Figma was occasionally used to conduct testing online over video, removing the requirements for in-person testing that would have slowed the design cycle.

- UI Library Integration - A point that will be discussed later on is the component library Shadcn The primary component library used for front-end development was Shadcn, and Figma integrates well with Shadcn through the Shadcn UI Toolkit - a Figma library that allows developers to use Shadcn components and Tailwindcss colours inside of Figma. This allowed for a direct 1-1 model of the final system to be composed as a prototype.

An important prototyping paradigm that was followed was leveraging Figma's component functionality where possible. This involves separating out individual components to design and change, then compiling all these components into a single design. The reason behind this was that it allows future changes to be done more easily, and removes many of the problems of making global changes to the prototype. Anything designed in Figma can be turned into a component, then instances of that component can be used in other sections or designs. Due to the fact that this was done previously in the larger Ahuora prototyping, the concurrent design changes that were done were immediately reflected on the designs that used them, saving a lot of time when matching the OpenPinch system with the greater platform. Fig. 2 demonstrates how this was done for this project, where all the individual components are separated out to be used to build other components and designs. This system also matched well with how the wire-framing and testing was conducted.

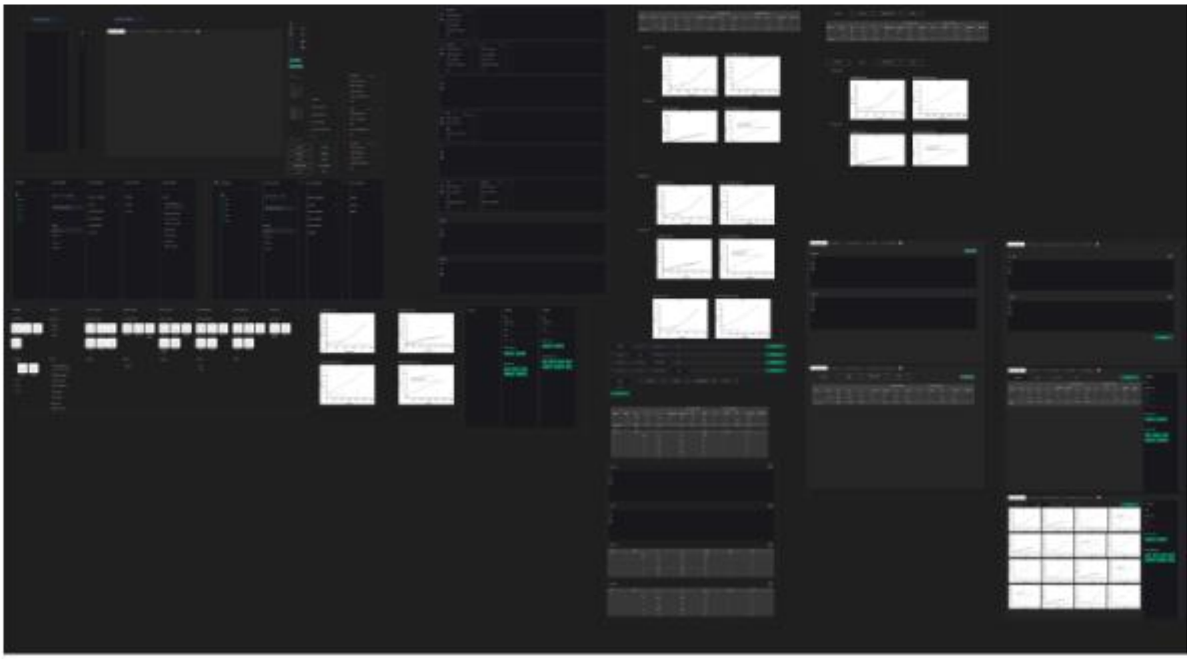

Figure 2 - Figma Components

At the end of this project, three major prototyping iterations have been completed, the third of which reflects the final implementation as seen on the front-end. Beyond these three major iterations, many smaller component level changes were done to polish and further the designs. This prototyping phase was invaluable to the development of the front-end, as changes to the front-end development would have been costly in development time, and the prototyping allowed the system to be matched to already tested and verified designs in a much faster and more efficient manner. Another key point that helped a lot when migrating the prototype designs to the actual system was that there was no need to check the technical feasibility of the system. Using the Shadcn toolkit for Figma allowed all the designs to be immediately implemented without many struggles developing the individual components.

One notable change that would be made if prototyping was undertaken again, is more time would be spent on wire-framing rather than prototyping. Despite this section being successful overall, there was a fair amount of wasted time identifying issues with the prototype that could have been caught earlier with more intensive wire-frame testing and design.

In an effort to progress through the design work and validate designs, a series of tests were conducted for each unique section. This was in an effort to allow valuable user feedback to be incorporated into the designs from an early stage.

The first step for design testing was to apply for and receive an ethics approval from the Waikato University division of Health, Engineering, Computing, and Science (HECS). This step was crucial, and needed to be done before any of the testing was conducted to maintain impartial results and ethical conduct with participants. In this application was a preliminary outline of the methodology that would be used for testing, alongside a participation form. This form was given to every applicant and outlines the steps that have been taken to ensure anonymity, the responsibilities of the conductor of the testing, and the rights of the interviewee.

An excerpt from the main body of the form is as follows:

Excerpt from Participant Information Sheet

Project Title
Designing the user interface of pinch analysis system to integrate into an existing application.

Purpose
This session of testing a digital prototype is conducted as a partial fulfillment of the requirements of an ENGEN582 capstone project. Low fidelity designs are required to be evaluated and refined as a part of the capstone project.

What is this research project about?
The overall aim of this research project is to use UI/UX testing techniques to evaluate the effectiveness of a digital prototype. This will involve exploring design and receiving feedback from participants to refine the final product.

What will you have to do and how long will it take?
The student will be providing you with a digital prototype and an environment where you will be able to interact with it. You will be asked to complete a set of instructions using the digital prototype after giving consent prior to the instructions, the student will be recording notes and observing how you navigate the system. After each minor task of the interview, you may be asked to give feedback and identify pain points that you encountered. After the interview is finished, you will be asked again to provide your consent to use the findings of the evaluation in a formal report. This process should take no more than 30 minutes. No personal or confidential information will be asked for or used during the interview.

What will happen to the information collected?
The information collected will be used by the student for their research and development project in ENGEN582-24X. This project involves presentations and a capstone portfolio. Only the researcher and supervisor will be privy to the notes, documents, and the paper written. Afterwards, this information will be destroyed, and erased. The researcher will keep transcriptions of the recordings and a copy of the paper but will treat them with the strictest confidentiality. No participants will be named in the publications and every effort will be made to disguise their identity.

Declaration to participants
If you take part in the study, you have the right to:
- Refuse to answer any particular question, and to withdraw from the study before 10th October 2024 when the report is due to be submitted.
- Ask any further questions about the study that occurred to you during your participation.
- Be given access to a summary of findings from the study when it is concluded.

Figure 3 - Participant Information Sheet

This project was approved by HECS shortly after its submission without any major changes required to the proposal and methodology. Once this approval came through, testing began immediately, and all participants were given the participant information sheet to read and sign. In total, 5 individuals were tested, the background of these participants varied, with some coming from a chemical engineering background while others were purely software engineers and designers. This variety allowed for a broad range of feedback. In general it was found that design and software engineers provided more in-depth feedback on website specific areas, such as identifying where the layout or components were not usable, while chemical engineers focused more on functionality and the systems relationship with products and software they had used previously. Both of these categories of feedback were valuable, and including these groups positively affected the end product

The wire-frame testing followed a less rigid format compared to the other testing conducted in this project. Participants were asked to review the designs and identify pain points or areas needing improvement with minimal prompts or specific workflows. This approach was particularly useful for the early wire-frames, which were created to explore the system's initial concepts. By using this flexible format, a wider range of ideas emerged — ideas that would have been harder to implement as the designs became more refined due to requiring more refactoring of prototypes and user interfaces. This allowed for quick experimentation and testing of different concepts, which were later polished through more testing.

For prototype testing, interviewees were given a loose workflow to follow. This workflow was structured in a way for testers to be able to see every part of the system being tested - this could be working their way through the whole system, or simply interacting with every part of a specific component. The prompts and workflow were left vague on purpose, as a key goal for this testing was to check that the system was intuitive and usable. If many users got stuck on a prompt or took too long to navigate to a certain area, this was noted down as a bottleneck to productivity and the next design cycle would address the concerns. This also serves to indicate exact changes that should be made - for example if users always try to click a certain element to navigate, then that element should have the functionality to navigate. An example of this would be something like "add a stream" then the users would be left to figure out how to navigate to that screen and input the data.

In total, 7 weeks of wire-frame and prototyping testing iterations were conducted across the duration of this project. In the beginning, this designing and testing was done during the back-end development to gauge practical design from the work being done, then later the design and testing was more focused addressing the front-end testing feedback.

3. Discussion

The testing as noted by Section 2 produced many valuable insights that were used throughout the project. As is the nature of any iterative process, a single result from this testing cannot be compiled, rather many documents like that of Fig. 4 and Fig. 5 were documented to inform development.

Iteration 1 - 6/08/2024

Filter inputs option

Easier to find calculate button

Add extra unit row as a header

full header names - or short names with a hover function for full name

Separate analysis and stream data tabs

Title should be Targets not summary

Better system for adding / removing filters. Either grey out non options, or delete with the option to re-add them somewhere. Greyed out would be the best option

There is some value in the utilities view

Global temperature / heat / other parameters

Side panels should be collapsible (options / filters panels)

Dropdown for graph type - selecting only one graph type is better than selecting multiple

Possible dashboard with custom outputs - specific graph types and targets

Add deselect all button

Add select all button

Auto scaling graphs

Hover over graph to see specific points

Scaled graph for problem tables

Ignore HSDT for now

Add site graphs as well as zone graphs

Figure 4 - Testing Feedback 1

The design phase played a significant role in shaping the front-end development and the final structure of the system. The initial vision for the project underwent a large variety of transformations during the wire-framing and testing stages, highlighting the immense influence that design testing and iterative practices can have on a project. Notably, the integration of design patterns and iterative workflows, developed during the literature review phase streamlined the development cycle and heavily influenced the overall design approach and project direction.

A key challenge during the design and testing process was the need to analyse and prioritize feedback effectively. Testing often resulted in conflicting opinions and general comments, which required careful analysis and additional rounds of testing to validate. Some feedback, such as suggestions on colour schemes or component layouts, had to be disregarded entirely, as they did not align with the established branding and design consistency set by the Ahuora Digital Twin Platform.

A notable subset of feedback that was present in the prototype portion of testing that was less common in others, is that participants were more open to suggesting novel features to the system. During wire-frame testing most participants would comment on the general layout of the UI and component positions, and during usability testing the focus was placed more on the workflow of the system rather

Iteration 1 Testing - 30/07/2024

- Make the dropdown on streams card clear that it is to select a process
- Toggle-able table view option for streams / utilities. Similar in execution to the current OpenPinch Excel Sheet
- Shorter names for variables - This is a Figma limitation due to lack of subscript / superscript
- Card View
- In general, saving the setup the user has decided. This will include which modules are collapsed / open, as well as what is a table view vs card view
- Icons for options are faint - need to add higher contrast
- Back to modify streams from outputs / being able to add / modify streams from the output page
- Header names on summary output are not clear. Keep with short names using subscript, with a longer name / description on hover on the header.
- Add units to header - for each header.
- For GCC's in particular, make them taller than they are wide, as vertical information is more important
- Plotly has a good system for zooming / managing graph outputs.
- Possibly be able to control x / y coordinate scale
- For more than two graph outputs in a process:
    - Two columns view
    - Space is expanded vertically downwards, with the box itself being expanded
- Collapsible process graph outputs - as well as draggable order of output

Figure 5- Testing Feedback 2

than missing features. One such example of this was the suggestion to add loading from CSV files that came through during prototype testing.

An area that would be explored if this project were to be undertaken again is heuristics testing. This was an area that was noted in the literature review conducted and would have improved the final design substantially - particularly near the end of development. Heuristics testing is a usability evaluation method in which a set of design principles, or "heuristics," are used to identify usability issues within a user interface. Typically performed by usability experts and designers, the evaluation is focused on how well the user interface adheres to these predefined principles. Incorporating the results from this testing would have brought an alternative perspective to the design work, and could have identified even more issues. Due to time and personnel constraints, heuristics testing wasn't viable for this project, as the usability experts weren't available during the time to perform this evaluation.

The results from these usability testing on the front-end helped inform the prototype, and in turn the front-end development. Many points of feedback that weren't identified during the prototype testing phases were encountered during this stage of the project, the most notable of which were concerning integration with the larger system.

4. Conclusion

In conclusion, conducting research into the usability of the Pinch Analysis system allowed for a natural progression of the design in response to user preferences. This process facilitated a significant improvement to both the user experience and the general aesthetics of the system. The feedback gathered from usability testing provided valuable insights, which in turn refined the key components and interface, making both more intuative. The results gained have had a significant impact on the development of the Pinch Analysis system.

Acknowledgements

I would like to extend my gratitude to Stephen Burroughs for their invaluable assistance in proofreading the content inside this document; taken from the larger dissertation. Their keen eye for detail and constructive feedback greatly improved the clarity and overall quality of my work. I truly appreciate their time, effort, and support throughout this process.

References

[1] Ministry for the Environment. New Zealand's projected greenhouse gas emissions to 2050. Dec. 2023. url: https : / / environment . govt . nz / facts - and - science / climate - change/new-zealands-projected-greenhouse-gas-emissions-to-2050/ (visited on 09/18/2024).

[2] MBIE. Decarbonising process heat. Dec. 2022. url: https : / / www . mbie . govt . nz / building-and-energy/energy-and-natural-resources/low-emissions-economy/ decarbonising-process-heat (visited on 09/18/2024).

[3] PHiNZ. Process Heat – Overview. Aug. 2018. url: https : / / www . mbie . govt . nz / dmsdocument / 152 - process - heat - current - state - fact - sheet - pdf (visited on 09/18/2024).

[4] Ministry for the Environment. National Policy Statement for Greenhouse Gas Emissions from Industrial Process Heat 2023. June 2023. url: https://environment.govt.nz/ publications / national - policy - statement - for - greenhouse - gas - emissions - from - industrial - process - heat - 2023 / # : ~ : text = The % 20National % 20Policy % 20Statement % 20for , of % 20heat % 20for % 20industrial % 20processes. (visited on 09/18/2024).