Visualising a complex data solution landscape

Approaching organisational issues from an Interaction Design perspective

Olof Ingelsten

ingelstenolof@gmail.com

Interaction Design

Bachelor

22.5HP

Spring 2023

Supervisor: David Cuartielles

# Abstract

# Acknowledgements

I would like to thank all those who helped me through the process of planning, designing, and writing this thesis. Firstly, I would like to thank Jonas Uddman for his encouragement and support and the entire UX team at Inter IKEA for their contributions and help with insights and testing. Secondly, I am grateful to my MAU supervisor, David Cuartielles, for always telling me what I need to hear and not what I necessarily want to hear. Lastly, a big thank you to all my friends and family who have supported me through all the highs and lows this semester.

# Table of Contents

Table of Contents

[1 Abstract 3](#_Toc134277452)

[2 Acknowledgements 3](#_Toc134277453)

[3 Table of Contents 3](#_Toc134277454)

[4 Introduction 5](#_Toc134277455)

[4.1 Context 5](#_Toc134277456)

[4.2 Aim 5](#_Toc134277457)

[4.3 Delimitations 5](#_Toc134277458)

[4.4 Structure of the thesis 5](#_Toc134277459)

[4.5 Problem statement 5](#_Toc134277460)

[4.5.1 Developing the research question through the process 5](#_Toc134277461)

[4.6 Related works 5](#_Toc134277462)

[4.6.1 Voyagers and Voyeurs: Supporting Asynchronous Collaborative Information Visualization 5](#_Toc134277463)

[5 Theory 7](#_Toc134277464)

[5.1 Terminology and definitions for interactive visualisations 7](#_Toc134277465)

[5.2 Designing user-centric visualisations 9](#_Toc134277466)

[5.3 Future implications and design challenges 10](#_Toc134277467)

[5.4 Summary 12](#_Toc134277468)

[5.4.1 Research question 12](#_Toc134277469)

[6 Design process and methods 13](#_Toc134277470)

[6.1 Planning 13](#_Toc134277471)

[6.1.1 User Centred Design 13](#_Toc134277472)

[6.1.2 Literature research 14](#_Toc134277473)

[6.1.3 The Double Diamond research method 15](#_Toc134277474)

[6.2 Early design stages 16](#_Toc134277475)

[6.2.1 Exploratory discussions 16](#_Toc134277476)

[6.2.2 Analysis process 17](#_Toc134277477)

[6.2.3 Results from interviews 18](#_Toc134277478)

[6.3 Sketching & Prototyping 21](#_Toc134277479)

[6.3.1 Prototyping methodology 21](#_Toc134277480)

[6.3.2 Three-point approach 22](#_Toc134277481)

[6.4 Usability testing 25](#_Toc134277482)

[6.4.1 Testing with users 26](#_Toc134277483)

[6.4.2 Iterating on the design 26](#_Toc134277484)

[7 Proof of Concept 31](#_Toc134277485)

[8 Discussion 31](#_Toc134277486)

[8.1 The prototype 31](#_Toc134277487)

[8.2 Design process 32](#_Toc134277488)

[8.2.1 Reflections on the interview process 32](#_Toc134277489)

[8.3 Working with an outer stakeholder 33](#_Toc134277490)

[8.4 Governance and handoff 34](#_Toc134277491)

[9 Conclusion 35](#_Toc134277492)

[10 References 35](#_Toc134277493)

# Introduction

## Context

* General overview of the field of visualisation and the and how it relates to Interaction Design
* Governance intro

## Delimitations

* The prototype
* Stakeholders

## Structure of the thesis

## Problem statement

### Developing the research question through the process

## Related works

### Voyagers and Voyeurs: Supporting Asynchronous Collaborative Information Visualization

(Heer et al., 2007) presents a design that implements aspects of asynchronous collaboration in a data visualisation program, *sense.us (*Figure 1*)*. The design aims to connect the cognitive benefits of visualisation with the benefits of interpersonal social interactions.

A computer screen shot of a graph

Description automatically generated with low confidence

Figure 1. sense.us prototype by (Heer et al., 2007)

To promote the social aspects of collaboration (Heer et al., 2007) implemented functions within the data visualisation software that allows users to comment, annotate, share insights, and discuss. From user tests, (Heer et al., 2007) could conclude that, through the implementation of social interactions and annotations, users engaged deeper in the data visualisations and helped each other make sense of and gain new insights otherwise were not presented through the visualisation. For future developments, (Heer et al., 2007) see a value in adding social aspects within visual interfaces empowering users to build upon and contribute rote knowledge.

(Heer et al., 2007) identifies three standard features in visualisation designs that implement multi-user bookmarking functions.

1. First, *application bookmarks* allow users to save a particular state or position within a visualisation for future reference. Application bookmarks can be shared between users or kept as personal references.
2. *Independent discussions* are places outside of the data interface which can point to specific parts within the interface. The data interface is one-way, meaning that external sources access the interface. Users within the interface cannot access the independent discussion.
3. Finally, (Heer et al., 2007) define *embedded discussions* as a functionality within the data interface that allows for streaming information accessed and viewed by any user.

The graphical annotations in the application used common tools and methods that their users already would be accustomed to. They drew inspiration from the annotation tools in Microsoft PowerPoint. The tools empowered users to add free-form drawings, lines, arrows, shapes, and text annotations.

(Heer et al., 2007) found that user-generated comments led to a greater shared understanding of the content in the visualisations as users now had the opportunity to ask and answer questions directly. Users also provided each other with contextual information not present in the data set. The annotations became an additional data point for the interface itself, adding to the searchable data.

Several users expanded upon the base functionalities in the software and used existing functions to build more advanced systems. One such system was the user implementation of narratives on the data sets. Through annotations and links to other data visualisations, users could expand upon the existing data set and contextualise the data through custom narratives.

# Theory

## The current state of Visualisation Design

The report by (Andrienko et al., 2020) aims to provide an overview from many different expert perspectives on the current goals and challenges in VIS. Each researcher was tasked to base their predictions and thoughts on two prompts, namely:

* *The top future research challenges in Big Data visualisation and analytics.*
* *The top emerging applications in the context of Big Data visualisation and analytics.*

A point shared by many of the researchers and the creators of the report for the future of Data Visualisations is the focus on their interactivity and accessibility.

Data analysis often leads to results that require expert interpretation. Thus, future Data interfaces “*should provide sustainable insights and insight recommendations*” (Andrienko et al., 2020, p. 3). Andrienko & Gennady (Andrienko et al., 2020) argue that the tools currently available have been science and research focused and are built in such a way that they are advanced at the cost of usability. From the perspective of Gennady & Andrienko, using visualisations should be continuously implemented throughout entire project processes, not only at the end, as is usually standard. Having a continuous dialogue with a visualisation system could effectively streamline current processes.

When designing visualisations in existing software, an issue arises when sufficiently complex data is used. Namely, the flexibility of use gets progressively harder, and insights gained rely heavily upon user interpretation. Kraska (Andrienko et al., 2020) identifies a design opportunity to design tools that do not restrict the user's approach to asking questions and allow for flexible changes in visualisation. The flexibility of such a system should then be quick in response regardless of the data size used. Like Andrienko & Gennady, Kraska urges that these systems' designs be accessible so that non-data scientists can interact with the visualisation interface. To make these data interfaces more accessible, designers can approach the interface's design from novel perspectives. These perspectives could be changes in the modality of interaction, as previously mentioned by Dimara & Perin (2020), as well as the design of the interface itself. Kraska urges a change in design thinking from the current focus on technical solutions done in the backend to the user-centric ideologies of HCI and IxD:

 “Design the user interactions first and then figure out the system [the backend], which can actually support them.” (Andrienko et al., 2020, p. 5)

Oulasvirta (Andrienko et al., 2020) provides a technology-agnostic perspective and urges incorporating user-centred design methods to design data visualisation and analytics. By understanding how users perceive their environments and how they decode data, we can design logically intuitive visualisations rather than computationally intuitive ones.

For the future of VIS, Oulasvirta sees value in building foundational research into the psychological aspects of human understanding rather than technological advancements. An opportunity for Interaction Designers to bridge fields and share methodology. Oulasvirta summarises their user-centred perspective with questions that challenge the main issues in VIS and asks the designer to challenge their design practice:

“**Why** should a particular visualisation be favoured over another one in some context? **Why** should one choose particular design parameters over other ones? What are the **limits** or a particular type of visualisation, what can it do and -more importantly- what can it **not** do.” (Andrienko et al., 2020, p. 7)

Fekete (Andrienko et al., 2020) says, “*To be effective, visualisation and visual analytics should be interactive, meaning that computing visual representations should happen in a few seconds, interacting on them should be responsive.” (Andrienko et al., 2020, p. 4)*. Fekete further argues for implementing *Progressive Data Analysis*. A similar method to how UX designers approach user flows on the internet. Large computational tasks are sectioned and divided into manageable chunks where the user can follow the process and deviate at any point instead of analysing in one step.

Additionally, Fisher (Andrienko et al., 2020) contextualises Fekete’s point by asking how to pose inquiries to data interfaces. Quicker responses from these tools would lead to a clearer dialogue between the person and the interface. Fisher further urges designers to create design systems that allow continuous interactions and are scoped in on specific visualisation tasks. By designing for specific use cases, designers will better understand users' wishes, use patterns, how they wish to interact, and the fidelity of the tools. Through this user-centric design ideology, Fisher believes that broader issues and questions in the field will become more manageable to answer and design for.

Therefore, for the future of tools in data visualisation, care should be taken to construct interaction flows that are *user-oriented* and intuitive to use while retaining their professional application. The opinions of the authors of (Andrienko et al., 2020) elude to aspects of design methodologies where Interaction Designers have opportunities to provide real value to the development of both fields.

## Terminology and Definitions for interactive visualisations

In the article “*What is Interaction for Data Visualization?*” (Dimara & Perin, 2020) look at the definition of the term *interaction* both from the perspective of the field of Data Visualisation (VIS) and from Human-Computer Interaction (HCI). They argue that several inconsistencies exist in how and when terminologies and methodologies are used regarding interactive data-driven visualisations and present a common language.

From the perspective of VIS, they define three concept approaches for interaction:

1. *System-centric* approaches disregard the specifics of the user, such as specific roles and how they interact with the artefact. Instead, this approach is technology centred and is concerned with program-specific operations.
2. *Task-centric* approaches involve the user to a lesser degree, focusing more on low-level tasks within the artefact (such as re-arranging, filtering, inputting data etc.).
3. Lastly, the *human-centric* approach is concerned with *user intent* and high-level interactions. For example, Dimara & Perin (2020) defines intent as a “high-level cognitive activity” that transcends the scope of interactions within a given software, such as note-taking and mental images of previous interactions.

Interaction within VIS is defined as a dialogue between the user and visualisation. However, they identify that interaction in VIS applications needs more flexibility for the user in the interaction flow. The VIS definition of a user is limited to a data analyst whose primary interaction tool is a computer with a mouse, limiting the design space in which new designs could be made.

Dimara & Perin’s (2020) view on the state of interaction within HCI is based on “*What is interaction*” by (Hornbæk & Oulasvirta, 2017), who defines “*seven concepts that characterise interaction in HCI*”, namely: Dialogue, Transmission, Control, Tool Use, Optimal Behaviour, Embodiment, and Experience.

Dimara & Perin (2020) identifies some key differences:

1. First, they see a differentiation between HCI and VIS in interpreting which *“entities”* take part in the moment of interaction. HCI detail the interplay between the human and the computer, whereas VIS interprets the data component within the application as a third entity in addition to the others.
2. HCI design is primarily focused on efficient user flows and designing intuitive interactions. In contrast, the flow in data visualisation applications tends to be slower and more complex as the user is always expected to be an expert.
3. *Intent* within VIS is more aligned with the user's state in the interaction flow and sees the entire flow as having a single goal in mind. On the other hand, (Hornbæk & Oulasvirta, 2017) identifies that user intent in HCI often starts before the actual flow of interaction and follows through to the end.
4. Finally, *flexibility* within each design differs between HCI and VIS. Where interactions in HCI are often low-level functions that act as parts of a whole and allow for flexibility within the defined scope, flexibility in VIS designs encompasses a larger scope and involves several aspects and parts of the interactive flow.

Dimara & Perin (2020) has defined the scope and definitions of HCI and VIS and found commonalities and differences in methodology, ideology, and terminology. With both fields’ views established, they define how interaction should be viewed in a visualisation setting.

“Interaction for visualisation is the interplay between a person and a data interface involving a data-related intent, at least one action from the person and an interface reaction that is perceived as such.”

(Dimara & Perin, 2020, p. 7)

Several sub-components define whether a visualisation interaction could be viewed as interactive to contextualise the interaction definition further.

1. **Interplay**. Interaction in a data visualisation setting should allow for a back-and-forth dialogue with the system. However, the authors chose to differentiate *interplay* from *dialogue* as they find it to define sequential interactions when that may not necessarily be the case in VIS interactions.
2. **Person**. Dimara & Perin talk about persons as *entities* that engage in interactions. A person refers to a human user but does not exclude non-human users.
3. **Data Interface**. The term *data interface* allows for agnostic interpretations that extend the design space scope to visualisations that, other than visual, are tactile, non-digital, auditory etc.
4. **Action**. An *action* could be any interaction performed by the person above, whether physical, mental, high-level, or low-level, that results in a reaction from the interface.
5. **Action-reaction**. Action-reaction refers to the dialogue between the person and the data interface. Specifically, it refers to the reactionary nature of natural dialogues between two people.
6. **Reaction perceived as such**. A necessary aspect for a reaction in a dialogue to be meaningful is that the recipient also *perceives* it as such. For example, suppose an interface reacts in a way where a clear cause and effect cannot be determined. In that case, the user becomes disconnected from the dialogue resulting in non-satisfying or non-effective interactions.
7. **Data-related intent**. As mentioned, the intent aspect is essential in HCI and VIS and is defined here as *data-related intent*. This aspect of the visualisation interaction implies that the person interacting has a specific intent before, after, or during the interaction related to the third entity in data visualisation interactions, namely, the data. The modality and temporality of the data-related intent are left open for interpretation.

## Designing user-centric visualisations

(Huron et al., 2014) present *Constructive Visualisation*, a perspective on designing simple, dynamic, and expressive visualisations. Constructive visualisations aim to empower users to, with simple building blocks, construct complex structures that embody their specific visualisation needs.

Huron et al. (2014) identifies three design challenges for making constructive visualisations (Figure 2):

1. **Keeping it simple**. Simple designs are something that users know intuitively and have known their entire lives. A key to designing for simplicity is identifying and leveraging actions intuitively connected to daily-and life-long activities.
2. **Enabling expressivity**. Expressivity in constructive visualisation would entail an interface that allows users to design non-destructively with reversible actions to adapt the interface to their needs.
3. **Incorporating dynamics**. Huron et al. (2014) explains that the most significant challenge is to design an interface in such a way that allows for simplicity and expressivity while still updating and adapting as the source of data changes. Currently, the most common way of designing dynamic visualisations is through coding, a non-common practice requiring in-depth knowledge to use practically.

A picture containing text, screenshot, font, number

Description automatically generated

Figure 2. Huron et al (2014) situate the design challenges with existing solutions.

Based on the design challenges and approaching visual interface design from a constructivist perspective, Huron et. al. (2014) propose Constructive Visualisation, *“/…/ the act of constructing a visualisation by assembling blocks, that have previously been assigned a data unit through a mapping.”* (Huron et al., 2014, p. 4). A constructive visualisation is comprised of components which are used in processes (Table 1):

|  |  |
| --- | --- |
| **Components** | **Description** |
| *Base token* | The token is the basic building block of which the data interface is compromised. The token can be physical or virtual and take any shape, colour, volume, texture etc., as well as interactive elements such as moving the token. Each token aims to represent a data element chosen through data mapping. |
| *Token grammar* | The token grammar of an interface defines different types of base tokens and how their attributes are mapped to the data. |
| *Environment* | The environment is the space that defines constraints of the data interface. The environment can be described through 2D-, 3D-graphics or through other spatial means. |
| *Assembly model* | The rules defined by the assembly model determine how tokens manifest in the visualisation. |
| **Processes** |  |
| *Environment initialisation*. | Define the visualisation environment and how the tokens are represented within. |
| *Mapping data to tokens*. | Defining token grammar and how it should be represented through each token. |
| *Assembling the tokens*. | The tokens should be presented in such a way to produce relevant value to the user. Defining the user’s intent of interaction. |
| *Evolution over time*. | How does the interface change and update over time? Is it manual, automatic, and how is it reflected through the interface? |

Table 1. Huron et al's definitions for Constructive Visualisations.

## Summary and Implications going forward

Through the analysis of the scholarly works, we have defined three key aspects of designing data interfaces:

1. We have identified that, from the perspective of VIS, there is a need for user-centred design approaches.
2. We have established a meta-discussion about the purpose and execution of what defines an interactive visualisation and how the user relates to it.
3. Lastly, we have an established model for explaining data interfaces from a user-centred perspective.

The implications of these learned aspects inform the design process and design prototype and will be developed throughout the text. With the context provided through the introduction and specified further by scholarly reports, the primary research question is presented as such:

How might we apply a user-centred design approach to the design of organisational visualisation needs?

Besides the primary research question, there are also a few sub-questions that I was interested in exploring as a part of this thesis:

What effect can an Interaction Design project have on a large organisation?

What are the implications of designing a user-centred data interface for Interaction Design and the organisation?

# Design process and methods

## Planning

In the following section, I will describe and outline the design process from start to finish, with each methodological choice described as they were applied. The structure of the text follows the design process chronologically, with literature research and user research leading into sketching and prototyping, which in turn ends in user testing and final prototype iterations.

### User Centred Design

In short, a User Centred Design (UCD) process is any design process that, in any way, includes the participation and inclusion of the intended end-user of the design (Abras et al., 2004). The main ideology of UCD is that by situating the user at the “centre” of the design process, designers can understand the user's needs on a deeper level, through which a design can be made that better aligns with user experiences and expectations. The benefits of UCD on design and research projects are numerous and often lead to more efficient design processes and project outcomes (Mao et al., 2005; Sharp et al., 2007).

*User Centred Design* was coined in the 1980s in a design laboratory led by Don Norman (Abras et al., 2004; Norman & Draper, 1986). From its conception to now, the adoption of UCD in research and design projects has increased considerably as the adoption of accessibility and useability laws has become more commonplace, as well as now common useability conferences (Mao et al., 2005; Marcus & Rosenzweig, 2020).

Since there are only a few users, I have approached the project from a *qualitative research* perspective. By focusing on a smaller set of users, I aim to understand the users’ individual needs, through which I can design a concept that would provide real value to each user.

Qualitative research addresses users on a deeper level. Therefore, it is possible to have a more nuanced discussion about the ethics of design and the roles of the designer (Brinkmann et al., 2014). (Fossey et al., 2002) explain further that when projects aim to do qualitative research, each part of the design process becomes influenced by qualitative measures. Qualitative research questions lead to qualitative methodologies, leading to qualitative results. Therefore, the implementation of qualitative measures directly connects to the quality of a project and is best described through transparent methodological choices (Fossey et al., 2002; Thorne, 2000).

A qualitative research method employed in this project is one with a rich ethnographic history, namely the interview. The core function of an interview is to (1) build a personal repour with the user, (2) gain a richer understanding of the user and the context in which they reside, (3) and explore the user in a dynamic setting (Knox & Burkard, 2009; Martin & Hanington, 2012; Wood, 1997). Interviews can be structured in a spectrum of fidelities and modalities. This thesis specifically employs the method of *semi-structured interviews*.

(Knox & Burkard, 2009, p. 2) defines the semi-structured interview as follows:

“/…/ using open-ended questions based on the study’s central focus /…/ to obtain specific information and enable comparison across cases; interviewers nevertheless remain open and flexible so that they may probe individual participants’ stories in more detail.”

(Wood, 1997) further contextualises Knox & Burkard’s (2009) definition through the lens of a user-centred software development process. Wood (1997) situates the researcher in three levels of knowledge to elicit different types of information from users. *Grand Tour* questioning focuses on broad contextual questions with little to no domain-specific terminologies to gain a base understanding of the user. *Case Focused* questioning focuses on specific interactions or flows in the user's workflow to elicit a deeper understanding of workflows and terminologies unknown to the researcher. Lastly, *Native Language* questioning requires understanding the user's field of work and focusing on domain-specific questions to elicit specialised answers.

### Literature research

Secondary research in the form of a literature review is commonplace in academic research. It provides a way for the researcher to review existing research and either synthesise new theories or situate research in relation to each other (Martin & Hanington, 2012).

The secondary research conducted in this thesis is mainly what (Ralph & Baltes, 2022) defines as an *Ad Hoc Review*. An Ad Hoc Review consists of a selection of papers chosen by the researcher to support, contextualise, develop, or otherwise situate the main topic of the research project. What differentiates the Ad Hoc Review from other literature review methodologies is its lack of systematised structure in how papers are picked. (Ralph & Baltes, 2022) calls the method for paper selection in Ad Hoc Reviews *purposive sampling*. The literature research was conducted through desk research using online databases such as *Google Scholar* and field-relevant databases such as the *ACM Digital Library* and *IEEE*. The process of filtering relevant research papers can be summarised as follows:

1. How many citations does the paper have?
2. When was the paper published?
3. How are the findings presented in the abstract?

### The Double Diamond research method

The double-diamond research process is a widely adopted method for describing and planning a design project. Design Council UK popularised the method through in-depth methodological research of 11 large design companies (Design Council, 2007; Gustafsson, 2019). In short, the method divides the design process into four phases, *Discover*, *Define*, *Develop*, and *Deliver*. A recurring theme of the method is a divergence and convergence of thought and design. The *discovery* phase diverges and opens the design space for exploration and research, which converges during the *define* phase into more well-defined design opportunities. Similarly, based on the design opportunities, the *develop* phase diverges into different prototyping iterations, which conclude through converging to a final design in the *deliver* phase (Design Council, 2019).

Since its initial conception, the Design Council has iterated upon the core model of the double diamond and has presented a refined model (Design Council, 2021). The revised Double Diamond approaches the design process from a more holistic perspective and includes a larger focus on the surrounding context and ethical implications of the design process. The core “diamonds” remain like earlier iterations, although renamed, with the new additions surrounding (Figure 1):

1. *Orientation and Vision Setting*. How will you approach the design? Why is this project important, and who will gain from its conceptualisation? What are the implications of the design, and how is it contextualised in a larger setting?
2. *Leadership and Storytelling*. The design process is collaborative, and a transparent design process builds a community. Through reflection and personal growth, you, as a leader, can support and build this community.
3. *Connections and Relationships*. Understanding the user and trusting them to be their own expert allows for a more inclusive design process.
4. *Continuing the Journey*. How will the design continue living as you finish the design process and move on? Everything we create exists in a larger ecosystem. What are the future implications of its existence? Reflect and learn from successes and mistakes.

Chart, radar chart

Description automatically generated

Figure 3. The revised Double Diamond (Design Council, 2021)

## Early design stages

Meetings were booked with each UX team designer during the project's first official week for the initial exploratory interviews. This included UX Designers, UX Design Leads, and the Technical Manager. The meetings were recorded through audio and were analysed and summarised afterwards in an affinity diagram using the digital whiteboard tool *FigJam*.

Before meeting each team member individually for the interview, I invited all the stakeholders to an introductory in-person meeting at the IKEA offices. It was an opportunity to get acquainted with the entire team, explain who I was, my intentions with the project, the project's purpose and expectations, and meet their comments regarding the thesis. A benefit of explaining this to everyone before the individual meetings was that everyone had a common baseline understanding of the project and its stakes. This allowed me to be direct and efficient without re-explaining the project at every meeting.

### Exploratory discussions

The individual meetings with each team member consisted of one-on-one sessions spanning 45-60 minutes in person or digitally (Four physical meetings, two digital). In addition, each interview was recorded through audio for later reference with consent from each interviewee. By recording each interview, I had the opportunity to focus more on the discussions with more fluid conversation rather than taking notes while talking, which can be difficult while conducting research alone.

The interview format was mainly *informal,* without any predetermined questions or questionnaires. It resembled an open discussion based on an initial prompt given to each interviewee, “*What type of value would a system solution overview bring to your work and design process?*” I wanted to learn the designers’ perspectives and values regarding how they are affected by the current digital landscape and how a mapping would impact them.

Beyond learning the *Why* of each user, I also wanted to know the *What* regarding the mapping. When and where would they see themselves utilising a mapping tool, in what context, and why? What type of data would be relevant for them to see? Through this, I could situate and contextualise the project within the greater scope of the digital landscape at Inter IKEA.

### Analysis process

Affinity diagramming is a method of data analysis that focuses on externalising insights and helps designers to structure information by categorising among common themes (Martin & Hanington, 2012). Insights and observations from user interviews are written down on individual post-it notes and scattered in no specific order. The post-its’ are then clustered by commonalities and themes. These themes are not determined beforehand but arise through the process. The themes can then be analysed and used to generate design opportunities and design issues that can be iterated upon (Beyer & Holtzblatt, 1998).

Chart

Description automatically generated

Figure 4. Finished layout after applying the affinity diagram method. Red post-its consist of raw data, and green post-its are generated insights.

After taking notes, analysing, and condensing the raw data to manageable insights through the affinity diagram (Figure 2), seven themes could be identified: *Data & Input*, *Purpose of the Overview*, *Finding Information*, *Issues & Problems*, *Governance & Ethics*, *Previous Mappings*, and *Practicalities*. Some themes were large enough to be divided further into sub-themes which will be expanded upon later. All insights gained through this process do not necessarily fit within this project's scope but provide a richer understanding of the context in which the project is situated and will be presented as such. The design opportunities and insights gained from this exercise and learnings from the literature inform the later sketching, designing, and prototyping stages.

### Results from interviews

|  |  |  |  |
| --- | --- | --- | --- |
| **Context & Connectivity** | **User Information** | **Solution Specifics** | **Interactions** |
| How do multiple solutions work together in a process, is it linear, circular, or iterative? | Who are the primary users of the solution, what roles use it, and who is the solution owner? | What is the primary function and purpose of the solution? | Several users used the analogy of “zooming” in and out when discussing interacting with the mapping. |
| How does data flow between solutions, and what type of data is it? | What is the primary function of a specific role within the solution, and what is its expected outcome? | What types of workflows and problems does the solution solve? | The data displayed should not be from a technical perspective, e.g., usage statistics or business goals and costs. |
| What is the context of the solution, and where does it feed data? | What does the user's workflow look like within the solution process from start to finish? | What types of data are handled within the solution? | The system should allow for flexibility in interaction and input to enforce a greater sense of ownership in interaction. |
| What tools or processes share data? | What other tools do the users use within the process? |  | Animations could augment and enforce certain overview aspects, e.g., data flow between solutions. |

Table 2. Data & Input.

The insights gained from *Data & Input* are technical and inform the design space accordingly. Therefore, they are framed as questions rather than straightforward observations. *Data & Input* can be divided into four sub-headings (Table 2).

|  |  |
| --- | --- |
| **The larger context** | **The personal context** |
| Communication in the UX Team needs to be more cohesive regarding the digital landscape. | Gain a deeper understanding of the users. |
| Most UX Designers need a clearer picture of the digital landscape. | Get an understanding of the digital landscape from multiple perspectives. |
| A deeper understanding of the digital landscape would aid in the design of new solutions. | Find related solutions to build more effective workflows and draw new connections. |
| Multiple differing perspectives between UX Designers and end users lead to a need for more consensus in discussions. “*There should always be a shared consensus between us designers and our users, and if it doesn’t happen all the time, it* ***should*** *happen all the time!*”. | Concretising projects and determining project directions. |
|  | The overview can be used as a personal “*quality assurance*” to situate oneself within the larger scope of the digital solution landscape. |

Table 3. Larger and personal context.

A recurring theme amongst the designers was the need for a common understanding of the contexts in which they worked. Through a better understanding of the larger context, some users hypothesised that projects could be defined better and have more focused design processes than they currently have. Beyond interpersonal communication and understanding, several users wished to have a better understanding for their own sake when approaching new projects (Table 3).

|  |
| --- |
| **Finding information** |
| Information access depends on the personal network of each designer. |
| Multiple intranet portals with bad search engines and UX lead to them being unused as sources of information. |
| Most designers contact other people through e-mails, Microsoft Teams chats, and book meetings to get the information they are after. |

Table 4. Finding information.

When asked how the designers find information about specific solutions in their work, they all said they prefer to contact and discuss it with someone. In these cases, the reason for preferring direct contact is that the existing digital sources of information regarding each solution are spread out over multiple databases, each displaying different types of data differently. The discrepancy in information availability and accessibility has led to a culture where people prefer personal methods of gaining information, namely booking meetings, sending messages, and e-mails. However, a limiting factor in this emergent system is that one's access to information directly correlates with the size of one's network, leading to situations where junior co-workers are unintentionally restricted from relevant data compared to older co-workers with a more extensive network (Table 4).

|  |
| --- |
| **Issues & Problems** |
| There is a lack of communication and shared data between teams. |
| The lack of consensus and landscape contextualising often leads to overlapping projects. |
| Those who discover overlaps are usually senior designers with a more extensive understanding of the digital landscape through work experience. |
| Much time is spent on projects researching problems already solved by other solutions. |
| Seeing a greater context outside your project space is difficult. “*We tend to work a lot in silos*”. |

Table 5. Issues & problems.

There is a need for consistent access to solution information within the organisation. Beyond difficult-to-find information, there is a need for better communication between teams. Project specifics are not directly available within the UX Team, requiring designers to actively seek out and learn about others' projects at their leisure. This often leads to situations where multiple teams work with significant overlap (Table 5).

|  |
| --- |
| **Governance** |
| Who *owns* the mapping? |
| How could you connect UX Designers in different projects? |
| How would you present the map to an outer stakeholder? |
| How to present multiple perspectives for a common understanding? |
| Empower junior designers with their learning and independence. |

Table 6. Governance.

The future development of a complete mapping application would entail engaging developers and designers outside of the UX Team, which further stresses the question of *governance*. When several teams design, develop and maintain the software, who is the owner (Table 6)?

|  |
| --- |
| **Previous Mappings** |
| Too specialised and constrained to specific projects. |
| Focused on singular perspectives. |
| Created for the UX Team but not by the UX Team. |
| After the project ended, there was no sense of ownership over the mapping. |
| Manual input was too time demanding and led to the mappings not being maintained. |

Table 7. Previous mappings.

Previous mappings were too specific and designed from a single perspective in the project, only providing insights in a narrow scope. Much time was spent on research and design that was discarded afterwards. Another issue was that the design of these mappings often was not created by the UX Team but resulted from an external consultant leading to a situation with no clear mapping owner. Lastly, the largest factor in why no mapping has been maintained is that the input method for each has been primarily through manual inputs requiring many work hours spent to keep the mapping up to date (Table 7).

|  |
| --- |
| **Practicalities** |
| Users would mainly use the mapping at the start of projects. |
| Values the insights gained as a point of departure. |
| Prefers to use tools already in their toolbox. |

Table 8. Practicalities.

As a part of the informal interview format, some discussions led to topics regarding practicalities surrounding the project and subsequent prototypes (Table 8).

## Sketching & Prototyping

### Prototyping methodology

Prototyping is a core methodology for any Interaction Design project and can be done in many ways to yield differing results (Houde & Hill, 1997). Prototyping, at its core, is a way for designers to compartmentalise and explore specific aspects of a design. Constructing prototypes follows the designer throughout the process and changes according to fidelity and purpose. Due to how functional prototyping is to the entire design process, a broad spectrum of methodologies exists to apply (Martin & Hanington, 2012).

As will be developed later, this thesis revolves around three main methods of prototyping and sketching (Table 9). However, there is a methodological distinction between sketching and prototyping (Buxton, 2007). Buxton (2007) explains that the main difference between sketching and prototyping is the low cost and time efficiency of sketches early in the design process. In a sense, they become disposable. Prototypes, conversely, are more intricate and require a more considerable time investment, making them less disposable. Both are different but still intertwined with the design process.

(Lim et al., 2008) further divides the function and purpose of any prototype into *filtering dimensions* and *manifestation dimensions*. Lim et al.’s (2008) prototype structure allow designers to analyse, reflect and contextualise their prototypes to determine direction and purpose. Combining Buxton’s (2007) definition and mindset with Lim et al.’s (2008) methodological analysis, we can understand how prototypes *inform* the design process while also being *integral* to it.

Wizard of Oz (WOZ) prototyping is a methodological tool where designers mock up an interaction flow without implementing full functionality. The key aspect of a WOZ prototype is that the user is unaware of the “lack” of fidelity in the prototype, leading them to believe it is more finished than it is (Dahlbäck et al., 1993). The main benefit of using WOZ is that it can save development time while yielding valuable insights regarding flows and interactions. Bill Buxton comments on the method:

“It is much easier, cheaper, faster, and more reliable to find a little old man, a microphone, and some loud speakers than it is to find a real wizard. So it is with most systems. Fake it before you build it”

(Buxton, 2007, p. 239)

### Three-point approach

The prototyping process, in the early stages, implemented a three-point approach using multiple mediums, each with benefits and drawbacks to the prototyping process. The prototyping mediums used were:

1. Sketching designs on paper
2. Visual & semi-interactive prototypes in Figma
3. Interactive prototypes in code

By working iteratively through the different processes, I could quickly sketch concepts and ideas on paper, sort through them to see which best would fit the intended interaction, mock up the designs in Figma and then implement those that were most viable into the live code prototype. This workflow enabled me to work as an ideator, a designer, and a developer while doing design hand-offs between each stage.

The choice of these three mediums over other potential means of ideation and prototyping comes mainly down to their accessibility, availability, and specialised areas of function. Another covers the limitations of one medium, and they complement each other in the workflow I have chosen to use here (Table 9).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Material* | *Resolution* | *Scope* | *Speed* | *Interactivity* |
| Paper | Low | Layout & interaction concepts | Fast | Limited |
| Figma | High | Visual design & user flow | Average | Semi |
| Code | Medium | Implementation & interactions | Slow | Full |

Table 9. Lim et al’s (2008) manifestation dimensions applied to the prototyping methods.

Diagram

Description automatically generated

Figure 5. Sketch prototypes

After ideation and sketching (Figure 3), the most viable options for further development were picked to be iterated in Figma. Building custom components using Figma’s component library function allowed me to quickly implement a design system that could be iterated upon to create different variations and designs (Figure 4). The benefit of iterating further in medium fidelity using Figma was that design flaws hidden in the abstractions of the paper sketches showed themselves, such as text positioning and sizing. Another benefit of working in a higher fidelity was the change of mindset and focus that came with it. While the paper sketches mainly focused on broader interactions and layout, the higher fidelity in the Figma prototype brought forth designs that focused on the content displayed and how the user would see and interact with it.

The layout design and content displayed were based on user reflections from the initial interviews and related visualisation software such as *Google Maps* (Google, 2023) and *Connected Papers* (Tarnavsky Eitan et al., 2023). According to the research done by (Heer et al., 2007), creating a design reminiscent of software the users are used to and comfortable with you can leverage their prior experience and create an experience with a lower point entry.



Figure 6. Iterations on the layout and design language

Figma can mock up a semi-interactive user flow. Without programming in code, designing clickable user flows with simple animations is possible. The benefit of making interactive flows in Figma in comparison to an actual code implementation is the speed of implementation. With the design system made with modular components, interactions and flows requiring a larger time investment to code can be first prototyped and tested. The interactive prototypes made in Figma provide an insight into the look and feel (Houde & Hill, 1997) of how a fully working implementation would work. However, despite the prototypes *looking* like a working implementation, they only exist statically within the limited scope in which they were created. They cannot respond dynamically to changing data and more complex interactions. The prototypes made in Figma are a type of WOZ prototype.

It is necessary to implement in code to prototype more complex interactions that use dynamic input and output where there is an action-reaction relation between the user and prototype.

Programming takes more time than the other, more visual prototyping methods and focuses more on *implementation* than *look and feel (Houde & Hill, 1997)*. The implementation prototypes build upon the *force-graph* library created by Vasco Asturiano (Asturiano, 2018/2023), a JavaScript library built upon the visualisation library *D3.js* (Bostock, 2021) and allows users to implement dynamic *force-directed graphs*. Besides using JavaScript, the prototype is built upon a base HTML website with CSS styling.

The early prototyping stages when doing an implementation prototype consist largely of material exploration and technical exercises. By exploring and iterating on simple concepts separately, such as reading data from JSON and filtering data points, I could later consolidate all learnings to construct a more comprehensive user flow (Figure 5).



Figure 7. Early functional prototype for the first user tests

The data used in the public prototype (Ingelsten, 2023)(Figure 5) is comprised of made-up solution names, solution owners, and roles, as the data provided by Inter IKEA is under confidentiality law. However, even though the data is a placeholder, the interactions and overall user experiences are the same.

For the coming user tests, there are two major prototypes to be tested, the Figma prototype and the implementation prototype. Both approach the design from different perspectives. The Figma prototype is focused on visual aesthetics and the interactions therein. The implementation prototype shows the interaction interplay between the data interface, data, and user. By separating the prototype into two more specialised prototypes, I aim to have more focused and directed discussions, taking a lesson from the informal interviews in the research stage.

## Usability testing

“Testing with end users is the most fundamental usability method and is in some sense indispensable. It provides direct information about how people use our systems and their exact problems with a specific interface.” (Holzinger, 2005, p. 3)

Usability testing (here referred to as *user testing*), as a general method, is just as interconnected with Interaction Design as user research, sketching, and prototyping. Like prototyping, user testing is a large field with many different methodologies that all aim to engage a user with some type of task (Holzinger, 2005; Moran, 2019). Through user testing at various stages in the design process, designers can learn valuable lessons regarding user expectations, assumptions, and usability issues. By iteratively including users in the design process through several prototypes, issues can be brought forward early, which could be costly later in the process (Holzinger, 2005).

A direct user testing method applied in this thesis is the *Thinking Aloud* method (Holzinger, 2005; Martin & Hanington, 2012). The Thinking Aloud method is quite straightforward. As the user is presented with the prototype to interact with, they are encouraged to verbalize their thoughts while interacting. Through this 1:1 connection between the interaction and verbalization, researchers can understand the user more clearly in their feedback and have the possibility to ask questions. Another benefit of having the users speak during the interaction (*Concurrent Think-Aloud* (Martin & Hanington, 2012)) compared to vocalizing after the fact (*Retrospective Think-Aloud (Martin & Hanington, 2012)*) is that you then rely on the user’s ability to recall past thoughts and impressions (Holzinger, 2005).

### Testing with users

User testing was conducted in two sessions with four of the users that were in the initial interviews. Additionally, some testing was done informally through other means (mainly supervisors and colleagues from MAU). The testing sessions were staggered with a few days between. This was partly due to the practicalities of finding time with the users and allowing iteration between sessions.

The user test process was like the informal interviews during the early research study. In short, the tests were conducted as follows:

1. Reiterate for the user the purpose of the design regarding the initial problem statement and what we discussed in our last meeting.
2. Explain the purpose and difference of the implementation prototype compared to the visual.
3. Guide the user through the prototypes while having an open design discussion regarding their moment-to-moment thoughts about the interactions, the visualisation, and the data displayed.
4. Lastly, a discussion with the user where they could go back through the prototypes and point to aspects that could be changed or improved.

After the user test concluded, I compared and showed the notes I took to ensure I had not missed or misinterpreted the user’s feedback. This process is not unlike the *Thinking Aloud* method described by (Holzinger, 2005).

### Iterating on the design

Several prototype iterations share multiple user testing groups with multiple perspectives of feedback and input. Therefore, the following section will focus on each distinctive *generation* of prototypes rather than user testing sessions.

#### Generation 1

Two major themes arose from this first testing phase, visual and interactive. Testers found the overall visual design of the prototype to be unappealing to the point of distracting from the intended interactions. The lack of labelling on each solution made finding specific solutions more difficult. Similarly, in this initial iteration, users could neither see the direction of the data flow nor what type of data. This was a highly requested addition from end users.

The filtering interactions made possible through the checkboxes at the top of the screen were naively implemented. They resulted in a jarring change in the visualisation, further confusing the testers (Figure 6). One tester also requested a more in-depth filtering system with tags connected to each solution. Users also requested a clearer way of visualising what filters were currently active in the visualisation. A part missing from these prototypes was the implementation of data flow and content, which was one of the requested functions from the user research. Users reiterated this through testing as well.

Overall, end users enjoyed the prototype conceptually, although not visually, and could see how it could be implemented into their workflow. One user said regarding the difference of this type of visualisation compared to the Excel sheets they are accustomed to: “*At the moment, the digital solution landscape is quite abstract for me, and I don’t get how any of it fits together. This really makes it concrete.*”.



Figure 8. Gen 1 unfiltered (left) and filtered (right).

#### Generation 2

After the first round of user testing, further technical and visual iterations were done to arrive at the final iteration of *Gen 2*. The user feedback on this prototype still involved some aesthetic discussions but focused more on interactions and user flow.

Visually, users liked the higher fidelity aesthetics of Gen 2 compared to Gen 1. However, the colours chosen (white, grey, red, and orange) were found distracting as they did not follow IKEA’s design system *SKAPA* as the users expected them to. Similarly, from the point of accessibility, the colours did not allow for easy reading due to differing contrasts. As a further effect of the visual design having poor contrast, high usage of box shadows, inconsistent text sizing, and use of white space, users found the prototype to have much visual noise, distracting from the intended user flow.

To address earlier feedback about data flow, Gen 2 implemented a particle system that visualised the data flow between solutions along the lines connecting each node. However, though implicitly understood by the users on a micro scale between individual solutions, the particles on a zoomed-out macro scale more resembled thousands of ants and did not convey the data flow in any practical sense. This implementation showed users *how* and *where* data flowed but missed the crucial aspect of *what* data flows between solutions. This was not part of the data set provided and had to be mocked up in the following iteration.

As each prototype got iterated upon and became more detailed, so did the feedback. Some user feedback regarded bugs in the prototype, which was a programming limitation. At this level of fidelity, discussions regarding the interactions no longer lingered on basics or bugs but looked forward to possible future implementations of new interactions and data connections. Such future implementation. Requests from users included the ability to specify their own processes and tags, a way to clear filters, clearer user flows for filtering, a search function, a way to edit/ add solutions, the ability to “favourite” solutions, and other data-specific requests.

Graphical user interface, application

Description automatically generated

Figure 9. Gen 2 unfiltered (left) and filtered (right).

#### Generation 3

At this prototype fidelity level, users started requesting features and polished interactions. For example, users want to be able to click on the filter solutions dialogue to enter specific solutions and see what filters are active. On the visualisation side, users wanted to see the area filter as a default instead of it being behind filtering. The ability to see the data flow was implemented in an earlier generation prototype. However, in this iteration, users wanted a faster way of seeing what data flows between solutions.

Graphical user interface

Description automatically generated with medium confidence

Figure 10. Gen 3 unfiltered (left) and filtered (right).

#### Generation 4

Consolidating feedback from all user tests that had not been implemented and organising them according to implementation viability, a final prototype generation could be iterated to be presented as the final proof of concept to stakeholders. A more detailed account of the final prototype can be found later in *Proof of Concept*.

Graphical user interface, text, application

Description automatically generated

Figure 11. Gen 4 unfiltered (left) and filtered (right).

#### Figma prototypes

Diagram

Description automatically generated

Figure 12. Visual Figma prototypes. (1) Area’s overlapping selection, (2) Process selection, (3) User overlap, (4) Manual editing of solutions, (5) Solution changes over time, future/past

1. Users conceptually liked the idea of visualising where areas overlap with a literal-coloured area. However, in the way implemented (Figure 10, picture 1) the colour semantics did not equate to user expectations and made the overall visual impression “muddy”. Furthermore, as some users pointed out, this example shows a small, best-case scenario and would not likely scale well with the more chaotic structure of the actual solution landscape.
2. End users most requested this prototype. Seldomly does one UX Designer work with the entire solution landscape. More commonly, each designer works with 2-5 solutions on a project-by-project basis, meaning having all solutions displayed at once is unnecessary. Users specified a wish to be able to specify their own processes on top of existing processes at the workplace. However, removing all other solutions besides those specified in the process filter was not a good alternative; the other solutions remained, albeit toned down.
3. A common need for designers when approaching new solutions is understanding what users are active there and what they do. Users understood the distinctions and overlapping users in the example, and several stated how this would allow them to understand the solution user’s workflow better.
4. Since most users find information through interpersonal connections instead of using digital databases, having the option to edit and add solution data manually would empower designers to utilise the wealth of verbal information that is present. This sentiment of enabling and working with the existing and preferred data collection method instead of trying to fix it resonated with users as they felt that having the option to edit and add would empower and strengthen their autonomy in the solution landscape.
5. Finally, one part of the visualisation’s utility is to see the solution landscape as it is currently. However, when planning and designing for new solutions, it is valuable to know how the surrounding landscape changes over time. Therefore, this visual prototype was iterated with user feedback to allow the designer to “peek” into the future of the solution landscape to see how it changes as solutions get replaced, new solutions get added, and solutions get removed.

# Proof of Concept

# Discussion

## The prototype

* **The prototype**
  + Quick run-through of the final proof of concept
  + Through the lens of research in the lit. review.
  + User testing
    - Meeting User feedback
    - Benefits/drawbacks of testing with different users each session
    - Invited users to dream bigger and ideate freely about future possibilities and solutions.
      * The prototype was never the end-all-be-all and was not interpreted as such 🡪 It is a proof of concept that has opened up a new design space at the organisation.

## Design process

### Reflections on the interview process

How the interviews were conducted and how the data was processed could have been improved to save time and yield finer results. Finer results, in this case, would entail discussions that stayed within the project scope and were more structured.

Although recording each session allowed for more engaging discussions, the following post-processing required considerable time. Most interviews still needed an additional hour to take notes, correct, and summarise. To save time in the post-processing stage, a more semi-structured interview with more precise goals and questions could have provided more efficient interviews, thus minimising time spent after the fact.

However, given the stakeholders within this project, semi-structured interviews are more beneficial than the traditional interview question format. Allowing the user more space to reflect upon their thoughts and practices demonstrated greater depth in each answer, which otherwise could be lost in a more rigid setting. A benefit of urging the users themselves to reflect on the discussion is that I, as the “designer”, cannot make any grounded assumptions about the user's work and values. They know their situation best.

Another reflection of the informal interview is its improvisational feel which not all users responded to. The thought processes of some interviewees worked very well with this format and encouraged an open environment with a workshop-like feel where spontaneous whiteboard collaborations arose. On the other hand, some users were less engaged with this type of interview and would have benefited from a more structured method.

Finally, booking six meetings in one week, especially the first week, was not the most efficient use of that time. As all interviewees were asked the same initial prompt combined with my shallow knowledge of their experiences, most users responded similarly. This is okay, as it demonstrates commonalities between the users and a direction for the project. However, this insight was gained after the first three interviews leading the remaining three to be more confirming discussions based on assumptions gained through the first three. By staggering the interviews weekly, more specific questions could be asked as new knowledge was gained.

#### Prototype data faked

Due to organisational secrecy, the data used in the prototypes is completely fake and fabricated. Testing with internal users, I was allowed to use real data. However, since most prototype testing was conducted through a publicly accessible website, the data had to be sanitised to resemble the original data set rather than represent it. To counteract possible user input regarding this change, the data was formatted in a way that seemed like real solutions at a cursory glance. This was done by imitating the naming schema commonly used in IKEA products, e.g., a capitalised Swedish word, often a location or noun, for example: “MÖCKELN”, “INSEXNYCKEL”, or “AGUNNARYD”.

#### Choice of user 🡪 UX Designers

Since the main user group on this project consists of UX Designers, the user testing method benefited from the users' expertise in design. Pivoting the tests to allow the user to freely discuss the prototype from their perspective as a UX Designer, I could get feedback that improved practical aspects such as visuals and interaction and aspects of the design that regarded user flow and accessibility. The users’ familiarity with design methodologies common to both UX and Interaction Design promoted an environment where, instead of the usual designer/user dynamic, the users and I could approach the prototypes as design equals. Although this project is not strictly intended to be about or aims to be about co-design, implementing co-design methodologies and principles can yield results that would otherwise be missed if you held a more traditional view of the user/designer relationship.

* **What have I learned about visualization?**
  + The actual visualization is not as important as the data around it.
    - Working with data, filtering, interacting, and researching users’ needs provided the valuable foundation the visualization is bult upon. The visualization is a means to an end.
  + It can change the way people see their organization and their place within it.
* **Pivoting to coding an implementation prototype**
  + Initial idea was to keep it strictly visual –> what changed?
  + How can coding a prototype add to my and the end -user experience?

## Working with an outer stakeholder

* **Working with a company/outer stakeholder**
  + Access to users
    - Very clear user base and needs.
    - You need to jump through hoops to get in touch.
    - Fitting in to a new organization culture.
    - Working with that culture and finding solutions around (eg. Three week process to set up accesses and emails that still don’t work)
    - Working with UX Designers as stakeholders
      * A different approach compared to other types of users
  + Using real data vs fake
    - Users wished for real data in prototypes.
    - Got access to incomplete sets of information.
    - Can’t share data publicly, company secrets 🡪 user testing gets really difficult.
    - Mockup similar fake data as a compromise
    - Requires careful explaining when doing tests
  + Balancing the stakeholders needs with the thesis needs
    - Creating an end-product that produces value for the stakeholder while remaining true to the thesis.
    - Clear communication and compromises with the stakeholder 🡪 thesis needs to come first
      * Discussions about what constitutes a satisfactory result for both parts.
      * How does it impact the dynamic between designer and stakeholder in a user centric design process when the designer has an intrinsic stake in the project

## Governance and handoff

* **What are the ethical aspects?**
  + Transparency within the company
  + Democratic empowerment to people working at the company
  + Who is accountable? If you don’t see other solutions/aren’t aware of better paths then who is responsible? Usually accountability falls on the unknowing worker.
  + New workers are overwhelmed when orgs get to large 🡪 one solution can be categorization through VIS
  + Meta discussion about ethics
    - How can Interaction Design projects/practices make workplaces more accessible?
    - Using design methodology for onboarding, empowering, and belonging.
* **Governance**
  + A real issue in large companies
  + At a certain size the company act more as an organism 🡪 not one clear driving force but several that all build a whole
  + “Accountability” when tools change and people move around -> You can’t always foresee changes but how can you maintain some relevant governance in digital solutions
* **Other thoughts**
  + Where else can the prototype be used
    - Inside IKEA / Outside use cases.
    - Any type of data that is serializable into JSON could work.
    - When does a solution become too generic?
    - Is there a limit to when being generic is bad, because that can also be a strength.

# Conclusion

# References

Abras, C., Maloney-Krichmar, D., & Preece, J. (2004). User-Centered Design. *Encyclopedia of Human-Computer Interaction*, 14.

Andrienko, G., Andrienko, N., Drucker, S. M., Fekete, J.-D., Fisher, D., Idreos, S., Kraska, T., Li, G., Ma, K.-L., Mackinlay, J. D., Oulasvirta, A., Schreck, T., Schmann, H., Stonebraker, M., Auber, D., Bikakis, N., Chrysanthis, P. K., Papastefanatos, G., & Sharaf, M. A. (2020, March 30). *Big Data Visualization and Analytics: Future Research Challenges and Emerging Applications*. BigVis 2020: Big Data Visual Exploration and Analytics. https://hal.inria.fr/hal-02568845

Asturiano, V. (2023). *Force-graph* (1.43.1) [JavaScript]. https://github.com/vasturiano/force-graph (Original work published 2018)

Beyer, H., & Holtzblatt, K. (1998). *Contextual design: Defining customer-centered systems*. Morgan Kaufmann.

Bostock, M. (2021). *D3.js*. D3.Js - Data-Driven Documents. https://d3js.org/

Brinkmann, S., Jacobsen, M. H., & Kristiansen, S. (2014). Historical Overview of Qualitative Research in the Social Sciences. In *Historical Overview of Qualitative Research in the Social Sciences*.

Buxton, W. (2007). *Sketching user experiences: Getting the design right and the right design*. Elsevier/Morgan Kaufmann.

Dahlbäck, N., Jönsson, A., & Ahrenberg, L. (1993). Wizard of Oz studies: Why and how. *Proceedings of the 1st International Conference on Intelligent User Interfaces - IUI ’93*, 193–200. https://doi.org/10.1145/169891.169968

Design Council. (2007). *Eleven lessons: Managing design in eleven global brands. A study of the design process*. Design Council UK. https://www.designcouncil.org.uk/our-work/skills-learning/resources/11-lessons-managing-design-global-brands/

Design Council. (2019, May 17). *Framework for Innovation: Design Council’s evolved Double Diamond*. https://www.designcouncil.org.uk/our-work/skills-learning/tools-frameworks/framework-for-innovation-design-councils-evolved-double-diamond/

Design Council. (2021). *Beyond Net Zero: A Systematic Design Approach* (p. 60). Design Council. https://www.designcouncil.org.uk/fileadmin/uploads/dc/Documents/Beyond%2520Net%2520Zero%2520-%2520A%2520Systemic%2520Design%2520Approach.pdf

Dimara, E., & Perin, C. (2020). What is Interaction for Data Visualization? *IEEE Transactions on Visualization and Computer Graphics*, *26*(1), 119–129. https://doi.org/10.1109/TVCG.2019.2934283

Fossey, E., Harvey, C., Mcdermott, F., & Davidson, L. (2002). *Understanding and Evaluating Qualitative Research*.

Google. (2023). *Google Maps*. Google Maps. https://www.google.com/maps/@55.5880384,13.0322337,14z

Gustafsson, D. (2019). *Analysing the Double diamond design process through research & implementation* [Master of Arts, Aalto University]. https://aaltodoc.aalto.fi/bitstream/handle/123456789/39285/master\_Gustafsson\_Daniel\_2019.pdf?sequence=1&isAllowed=y

Heer, J., Viégas, F. B., & Wattenberg, M. (2007). Voyagers and voyeurs: Supporting asynchronous collaborative information visualization. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1029–1038. https://doi.org/10.1145/1240624.1240781

Holzinger, A. (2005). Usability engineering methods for software developers. *Communications of the ACM*, *48*(1), 71–74. https://doi.org/10.1145/1039539.1039541

Hornbæk, K., & Oulasvirta, A. (2017). What Is Interaction? *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 5040–5052. https://doi.org/10.1145/3025453.3025765

Houde, S., & Hill, C. (1997). What do Prototypes Prototype? In *Handbook of Human-Computer Interaction* (pp. 367–381). Elsevier. https://doi.org/10.1016/B978-044481862-1.50082-0

Huron, S., Carpendale, S., Thudt, A., Tang, A., & Mauerer, M. (2014). Constructive visualization. *Proceedings of the 2014 Conference on Designing Interactive Systems*, 433–442. https://doi.org/10.1145/2598510.2598566

Ingelsten, O. (2023). *IxD Thesis 2023*. http://www.inglestone.art/

Knox, S., & Burkard, A. W. (2009). Qualitative research interviews. *Psychotherapy Research*, *19*(4–5), 566–575. https://doi.org/10.1080/10503300802702105

Lim, Y.-K., Stolterman, E., & Tenenberg, J. (2008). The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. *ACM Transactions on Computer-Human Interaction*, *15*(2), 1–27. https://doi.org/10.1145/1375761.1375762

Mao, J.-Y., Vredenburg, K., Smith, P. W., & Carey, T. (2005). The state of user-centered design practice. *Communications of the ACM*, *48*(3), 105–109. https://doi.org/10.1145/1047671.1047677

Marcus, A., & Rosenzweig, E. (Eds.). (2020). *Design, User Experience, and Usability. Interaction Design: 9th International Conference, DUXU 2020, Held as Part of the 22nd HCI International Conference, HCII 2020, Copenhagen, Denmark, July 19–24, 2020, Proceedings, Part I* (Vol. 12200). Springer International Publishing. https://doi.org/10.1007/978-3-030-49713-2

Martin, B., & Hanington, B. (2012). Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions. *Choice Reviews Online*, *49*(10), 49-5403-49–5403. https://doi.org/10.5860/CHOICE.49-5403

Moran, K. (2019, December 1). *Usability Testing 101*. Nielsen Norman Group. https://www.nngroup.com/articles/usability-testing-101/

Norman, D. A., & Draper, S. W. (Eds.). (1986). *User centered system design: New perspectives on human-computer interaction*. L. Erlbaum Associates.

Ralph, P., & Baltes, S. (2022). Paving the way for mature secondary research: The seven types of literature review. *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 1632–1636. https://doi.org/10.1145/3540250.3560877

Sharp, H., Rogers, Y., & Preece, J. (2007). *Interaction design: Beyond human-computer interaction* (2nd ed). Wiley.

Tarnavsky Eitan, A., Smolyansky, E., Knaan Harpaz, I., & Perets, S. (2023). *Connected Papers*. Connected Papers | Find and Explore Academic Papers. https://www.connectedpapers.com/

Thorne, S. (2000). Data analysis in qualitative research. *Evidence-Based Nursing*, *3*(3), 68–70. https://doi.org/10.1136/ebn.3.3.68

Wood, L. E. (1997). Semi-structured interviewing for user-centered design. *Interactions*, *4*(2), 48–61. https://doi.org/10.1145/245129.245134