Visualising the digital landscape at Inter IKEA

A user-centred approach to data visualisation

Olof Ingelsten

ingelstenolof@gmail.com

Interaction Design

Bachelor

22.5HP

Spring 2023

Supervisor: David Cuartielles

# Abstract

T.B.A

# Acknowledgements

I want 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 rather than what I necessarily want to hear. Thirdly, thank you coffee for always being there. 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 2](#_Toc134802761)

[2 Acknowledgements 2](#_Toc134802762)

[3 Table of Contents 2](#_Toc134802763)

[4 Introduction 4](#_Toc134802764)

[4.1 Context 4](#_Toc134802765)

[4.1.1 Software Development Governance 5](#_Toc134802766)

[4.1.2 Background 6](#_Toc134802767)

[4.1.1 Delimitations 7](#_Toc134802768)

[4.2 Related works 7](#_Toc134802769)

[4.2.1 Connected Papers 7](#_Toc134802770)

[4.2.2 Sense.us 9](#_Toc134802771)

[4.2.3 Google Maps 10](#_Toc134802772)

[4.2.4 Summary 11](#_Toc134802773)

[5 Theory 12](#_Toc134802774)

[5.1 The current state of Visualisation Design 12](#_Toc134802775)

[5.2 Terminology and Definitions for interactive visualisations 14](#_Toc134802776)

[5.3 Designing user-centric visualisations 16](#_Toc134802777)

[5.4 Summary and Implications going forward 17](#_Toc134802778)

[5.4.1 Research question 18](#_Toc134802779)

[6 Design process and methods 18](#_Toc134802780)

[6.1 Planning 18](#_Toc134802781)

[6.1.1 User Centred Design 18](#_Toc134802782)

[6.1.2 Literature research 20](#_Toc134802783)

[6.1.3 The Double Diamond research method 20](#_Toc134802784)

[6.2 Early design stages 21](#_Toc134802785)

[6.2.1 Exploratory discussions 22](#_Toc134802786)

[6.2.2 Analysis process 22](#_Toc134802787)

[6.2.3 Results from interviews 23](#_Toc134802788)

[6.3 Sketching & Prototyping 27](#_Toc134802789)

[6.3.1 Prototyping methodology 27](#_Toc134802790)

[6.3.2 Three-point approach 28](#_Toc134802791)

[6.4 Usability testing 31](#_Toc134802792)

[6.4.1 Testing with users 32](#_Toc134802793)

[6.4.2 Iterating on the design 32](#_Toc134802794)

[7 Proof of Concept 37](#_Toc134802795)

[8 Discussion 42](#_Toc134802796)

[8.1 Answering research questions 42](#_Toc134802797)

[8.2 The Proof-of-Concept and Literature 43](#_Toc134802798)

[8.3 Design Process 45](#_Toc134802799)

[8.3.1 Pivoting to coding 45](#_Toc134802800)

[8.3.2 Reflections on the interview process 45](#_Toc134802801)

[8.3.3 Reflections on user testing 46](#_Toc134802802)

[8.4 Stakeholder 47](#_Toc134802803)

[8.4.1 Prototype data faked 47](#_Toc134802804)

[8.4.2 Access to users 47](#_Toc134802805)

[8.4.3 A balance between stakeholder and thesis 48](#_Toc134802806)

[8.4.4 User groups 48](#_Toc134802807)

[8.5 Governance 49](#_Toc134802808)

[8.6 Ethics 50](#_Toc134802809)

[8.7 Future work 50](#_Toc134802810)

[9 Conclusion 51](#_Toc134802811)

[10 References 51](#_Toc134802812)

# Introduction

## Context

What we could today call data visualisation began in the 1600s when Descartes designed one of the first 2D coordinate systems to plot mathematical equations. However, simple tables have been used far earlier (Few, 2023). Since the 1600s, there have been technological advancements in graphing, plotting, collecting, and storing data. In addition, the advent of the digital computer with a visual interface introduced new ways to design visualisations. Foundational for the modern perception of visualisation is the publication *Sémiologie Graphique* by Bertin and Barbut (Bertin & Barbut, 1968; Few, 2023). Bertin and Barbut’s contribution was a complete classification of how to design and display data graphically. This is the first published book that aimed to set a foundation for what was later called Information Design. *Visual Complexity* (Lima, 2023) is a modern database that collects various visualisations, including art projects, government visualisations, and other scholarly visualisations that contribute with novel approaches to the field.

Another seminal addition to Information Design in the computer age is *The Eyes Have It: A Task by Data Type Taxonomy for Information Visualisation* by Ben Shneiderman (1996). Shneiderman’s contribution provided a taxonomy which described a novel approach to Visualisation Design. Most commonly, Shneiderman is cited through what is referred to as *Shneiderman’s mantra*:

“Overview first, zoom and filter, then details-on-demand.

Overview first, zoom and filter, then details-on-demand.

Overview first, zoom and filter, then details-on-demand.

Etc…”

(Shneiderman, 1996, p. 2)

The mantra relates to how humans cognitively organise and perceive data and how designers can leverage that knowledge. For example, instead of revealing all data at once, display a minimised summary that the user can interact with and filter to get more detailed information. This user flow externalises cognition from the user, as Bertin and Barbut would say, and creates a better user experience (Bertin & Barbut, 1968; Fry, 2004).

However, there is a divide between the different design fields related to data visualisation (Fry, 2004)(Figure 1).

A picture containing text, receipt, font, screenshot

Description automatically generated

Figure 1. The division in the design process (Fry, 2004, p. 13).

Each field has internal methods and processes, but there is little shared over scholarly borders (Fry, 2004), a notion supported by Andrienko et al., (2020), which will be discussed later in *The current state of Visualisation Design*. The issue arises in the border between fields where the data context is lost in translation and becomes, as Fry says, a “game of telephone”. This presents a novel opportunity for Interaction Designers to approach data visualisation not as computer scientists or graphic designers but as researchers connecting all parts of the design process.

There are several commercial mapping solutions available for a multitude of purposes. For example, for data analytics and statistics, there are Microsoft BI (Microsoft, 2023b), Tableau (Tableau, 2023), Google Sheets (Google, 2023b), and Excel (Microsoft, 2023a) and many more. These powerful applications allow users to map complex data through several graphing options. However, they are aimed at larger companies and often require a paid subscription. Beyond the scope of business-related analytics, several mapping options exist that aim to aid individuals by mapping everyday actions like *running* (On The Go Map, 2023), *cycling* (Komoot, 2023; Strava, 2023), *health* (Apple, 2023; Fitbit, 2023), and *Internet of Things* (IoT) (Home Assistant, 2023).

With the highly technological society we live in and the vast amounts of data generated daily, there is a greater need for ways to parse and visualise it. The complexity of data today is way beyond what one mind can comprehend, and visualisation is an effective tool that externalises the cognitive load making data easier to understand (Bertin & Barbut, 1968; Fry, 2008).

### Software Development Governance

The design of new software often requires a team of several people working in different roles. However, as people move on to other projects, who is responsible for the upkeep of legacy software? Moreover, what determines who is accountable for the software?

This introduces the concept of *Software Development Governance* (SDG). Chulani et al. (2008) defines SDG through two main aspects:

“(1) Establishing chains of responsibility, authority, and communication to empower people within a software development organization (static or structural component of governance).

(2) Establishing measurement and control mechanisms to enable software developers, project managers and others within a software development organization to carry out their roles and responsibilities (dynamic or measurement component of governance)”

(Chulani et al., 2008, p. 1).

Understanding governance and how it affects a solution roadmap is, according to Chulani et al. (2008), a way to connect the business needs of the organisation at large to the processes and outcomes of software. Furthermore, Chulani et al. (2008) develops three main SDG goals: *Managing value*, *Developing flexibility*, and *Controlling risk & change*. By setting and managing reasonable and relevant governance goals, organisations can ensure that the software development process uses reasonable resources, defines who is responsible, and has a roadmap for the software’s life cycle.

During the design process, governance became increasingly important, the extent of which will be proven later in the Design process and methods and Discussion sections.

### Background

This thesis is written and designed in collaboration with Inter IKEA of Sweden. A part of this collaboration consists of a pre-determined need established by the supervisor provided by IKEA. Through this project, I can research and design something that could provide real value to an outer stakeholder and research value in Interaction Design. Below is an abbreviated outline of the problem statement, background, and expected outcome.

Data & Technology provides internal IT Solutions for product development at IKEA. The portfolio of IT solutions at Data & Technology consists of roughly 60 different solutions, each in different stages of product development. Some of them are built internally, and others are bought. Each IT solution has a Solution Owner and a team working on improving or maintaining the solution. Each solution has several user groups (Product Design Engineers, Product Design Developers etc.), each of which can have different organisational roles. Each user also has internal roles connected to each IT solution, such as admin, editor, reader, approver etc.

There is only a rough sense of which users have access to what tools, and most information is inconsistent as to what roles use the solution, for how long, and for what purpose. Several processes provide high-level overviews (Organisational groups within IKEA) and very low-level views (Detailed steps in specific software for user flows and actions). An intermediate, mid-level overview of these functions and processes needs to be added, which connects the low-level overviews to the high-level.

Several attempts have been made to map IKEA’s digital landscape. However, most mapping solutions have been limited to one viewpoint, such as technical, process or IT solutions, limiting their usefulness when applying them elsewhere.

Furthermore, due to the complexity of the present systems, most solutions become overwhelming and complex to such a point where maintaining them becomes a larger hassle in the long term than the value they provide.

The expected outcome, on behalf of IKEA, is as follows:

* A maintainable mapping standard, created in a suitable tool, with support for multiple perspectives (IT Solutions, Roles, Function & Purpose, Process etc.).

### Delimitations

This project has several stakeholders with differing perspectives in its execution. Firstly, there is Malmö University, for whom I am writing this thesis. Secondly, there is the supervisor at Inter IKEA who has issued and organised the thesis project. Lastly, the UX Designers at Inter IKEA are the project's intended end-users.

The scope of this project and its subsequent design is contained to the needs of the UX Team at *Data & Technology* inside Inter IKEA based on the provided problem statement. The design process' outcome is to design the foundation for a future mapping solution. Therefore, the proof-of-concept design is intended to be a vertical slice to show how an eventual implementation could provide business value to the organisation.

The prototype's content is also limited to what is reasonable to create within the limitations of a bachelor's thesis. However, even though all data is not present, the base methodologies and interactions applied are the same and scale appropriately. Furthermore, the information relating to solutions and internal functions presented in this report is sanitised and made up to *represent* real data, as most solution-specific data is held under secrecy.

## Related works

### Connected Papers

*Connected Papers* is an online tool for scholars and researchers to search for papers relevant to their field. Connected Papers differ from similar scholarly databases, like IEEE, ACM, and Google Scholar, through a search algorithm that indexes papers based on similarity rather than citations (Tarnavsky Eitan et al., 2023). This allows researchers to find relevant literature that could be missed through conventional search methods.

The tool works on top of a dynamic database (Ammar et al., 2018) that regularly updates and adds new papers as they are published through conventional journals. This allows for an efficient user experience for the end user as the information is as recent as possible. Similarly, the team developing Connected Papers offloads the work of updating papers to each field's journal database allowing for more development time on the application.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 2. The Connected Papers application. Shown here is the graph for "What Do Prototypes Prototype" (Houde & Hill, 1997).

The application consists of three main interactive dialogues.

1. A main dialogue that displays a 2D force-directed graph map surrounding the “origin” paper (Figure 2, centre). Each node in the map represents a paper that, in some way, is similar to the original paper. The nodes are spread out and clustered from the origin according to their relevance and similarity. Their colour and size represent the year published and the number of citations.
2. The leftmost dialogue (Figure 2, left) displays a list view of the papers shown in the main dialogue and is further searchable through text. This list is also filterable through a variety of metrics.
3. The rightmost dialogue (Figure 2, right) displays detailed information regarding the selected paper, like name, author(s), citations, DOI links, and abstract. The user can save papers for later reference if they have an account.

The design language of Connected Papers embodies the visualisation design principles proposed by Shneiderman (1996). By “funnelling” the information in visualisations after Shneiderman’s mantra, Tarnavsky Eitan et al. (2023) eases the user's cognitive load during the interaction. The way Connected Papers structure the user’ access to information allows for multiple approaches when searching for related papers. Users who only want a simple visual overview to reference can do so without distraction. Similarly, users searching for specific papers can *zoom, filter,* and get desired *details on demand*.

### Sense.us

Heer et al., (2007) presents a design that implements aspects of asynchronous collaboration in a data visualisation program, *sense.us (*Figure 3*)*. 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 3. sense.us application by Heer et al., (2007, p. 3).

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) identify 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 beyond 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.

### Google Maps

*Google Maps* (Google, 2023a) is probably the most used map application. Google Maps allows users to search and filter locations worldwide, find directions, read detailed information about businesses and points of interest, and leave reviews.

Like the previous examples, the Google Maps application is structured in such a way as to provide varying levels of information to the user. The main screen of the applications is a geographical map initially centred around the user position from which the user can zoom and pan (Figure 4). When searching, users are presented with a dialogue displaying detailed information about the requested location (Figure 4, left). The filtering system present within Google Maps is quite granular and allows for custom user queries. Since it is a geographical mapping tool, filters are specialised after finding restaurants, public transport, museums, activities etc. (Figure 4, buttons at the top).

The database powering Google Maps is partly sourced through its users, third-party databases, and algorithmically scanning the internet (Russel, 2019). Users can update and add new data to Google Maps, which is then analysed and approved through various processes. This, like Connected Papers, allows the developers to serve a rich end-user experience without manually updating every location worldwide.

A screenshot of a map

Description automatically generated

Figure 4. Google Maps.

### Summary

What do Connected Papers, sense.us, and Google Maps have in common, and how do they relate to data analytics and visualisation? If we disregard the current content the applications present and look at them from an outside perspective, we can see:

1. We have a mapping tool that takes data and represents it visually on an interactive 2D interface.
2. The data is updated dynamically by crowdsourcing updates to third parties.
3. The tool provides a base set of categories that each data point can adopt.
4. End-users can filter the visualisation through metrics relevant to the application's main purpose.
5. Depending on user interaction, the application presents the data in varying detail.

By looking at tools already in use and learning from them, we can reapply and recontextualise their methods in new applications where such solutions have not been tried.

In the following chapter, I will discuss and look at how both Visualisation Design and Human-Computer Interaction relate to data visualisation and outline methodologies and terminology to explain this type of visualisation.

# Theory

## The current state of Visualisation Design

Andrienko et al., (2020) produced a report covering the future of Data Visualisation (VIS) built from many different expert perspectives on the current goals and challenges in VIS. Each researcher was tasked with presenting their predictions and thoughts for the field based 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 study participants is the focus on the visualisations' interactivity and accessibility.

Data analysis often leads to results which require expert interpretation. Thus, future Data interfaces “*should provide sustainable insights and insight recommendations*” (Andrienko et al., 2020, p. 3). Andrienko & Gennady argue that the tools currently available have been science- and research focused, and that are built to be advanced at the cost of usability. From the perspective of Andrienko & Gennady, visualisations should be used 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.

There is an issue when designing visualisations in existing software and handling sufficiently complex data. 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 interactive 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 for non-data scientists to interact with the visualisation’s 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 suggests a change in design thinking moving away from the current focus on technical solutions done at the backend and towards to the user-centric ideologies of Human-Computer Interactions (HCI) and Interaction Design (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 pushes for 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.

Regarding the future of VIS, Oulasvirta sees value in building foundational research into the psychological aspects of human understanding rather than technological advancements. This is 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 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 studying and designing 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 go.

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 human and the interface. Fisher further invites designers to create design systems that allow for continuous interactions and that are scoped in on specific visualisation tasks. By designing for specific use cases, designers will better understand users' wishes, how they wish to interact, use patterns and the fidelity level of the tools. Through this user-centric design approach, 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* from the perspective of VIS and 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 conceptual approaches for interaction: system-, task-, and human-centric. Where the third one is concerned with *user intent* and high-level 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 define “*seven concepts that characterise interaction in HCI*”, namely: Dialogue, transmission, control, tool Use, optimal behaviour, embodiment, and experience.

Dimara & Perin (2020) identify 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) identify 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) have 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.”9

(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 ones.
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) identify three design challenges for making constructive visualisations (Figure 5):

1. **Keeping it simple**. A key to designing for simplicity is identifying and 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 data dynamics**. The most significant challenge is to design and adapt as the source of data changes. Currently, the most common way of designing dynamic visualisations is through coding.

A picture containing text, screenshot, font, number

Description automatically generated

Figure 5. 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 (2014) 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 (Section 5.1).
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 (Section 5.2).
3. Lastly, we have an established model for explaining data interfaces from a user-centred perspective (Section 5.3).

### Research question

The implications of the above-mentioned aspects inform both the design process and the prototype design, and will be further developed throughout the text. Therefore, with the context provided through the introduction and further specifies by scholarly reports, the primary research question is presented as such:

How might we apply a user-centred design approach when designing data visualisations?

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

Who owns and maintains software after a design process like this, and whom does it affect?

# 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 prototype iterations resulting into a final prototype.

### 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). UCD situates the user at the “centre” of the design process, designers gain a deep understanding of the users’ needs, through which a design can be made that better aligns with the users’ experiences and expectations. The benefits of UCD on design and research projects are numerous and often lead to more efficient design processes and higher acceptance of 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 implementation of accessibility and useability laws has become more commonplace, as well as now common useability conferences (Mao et al., 2005; Marcus & Rosenzweig, 2020).

I have approached the project from a *qualitative research* perspective as the user group is a single team of eight. 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. 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 how 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 the interview. The core function of an interview is to (1) build a personal report 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) 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.” (Knox & Burkard, 2009, p. 2)

In addition to Knox & Burkard’s (2009) broader definition of semi-structured interviews, Wood (1997) proposes a methodological context 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-, case focused-, and native language questioning.

* *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.
* *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 6):

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 6. The revised Double Diamond (Design Council, 2021)

## Early design stages

After concluding the initial literature research, I wanted to meet all users and begin the user research process. Before individually meeting each team member 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, expectations, and get their first impressions 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

Meetings were booked with all the members of the UX team at IKEA’s site during the project's first official week for a round of initial exploratory interviews. This included UX Designers, UX Design Leads, and the Technical Manager. The audio of the meetings was recorded and the recordings were analysed and summarised afterwards in an affinity diagram using the digital whiteboard tool *FigJam*.

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,* with only one predetermined question given to each interviewee, “*Why would a system solution overview bring value 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 and How* 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? How do they imagine such a tool looking like? Through this, I could situate and contextualise the project within the greater scope of the digital landscape at Inter IKEA.

### Analysis process

Affinity diagrams as a method of data analysis focuses on unveiling 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 sticky-notes and scattered in no specific order. The sticky-notes 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 7. Finished layout after applying the affinity diagram method. Red sticky-notes consist of raw data, and green sticky-notes are generated insights.

After taking notes, analysing, and condensing the raw data to manageable insights through the affinity diagram (Figure 7), 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

The following tables show the results from user interviews, sub-headings are organised horisontally and insights vertically.

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, *context and connectivity*, *user information*, *solution specifics* and *interactions* (Table 2).

|  |  |  |  |
| --- | --- | --- | --- |
| **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.

A recurring theme amongst the designers was the need for a common understanding of the digital solution landscape in which they worked. Most designers work against a few solutions which they have in-depth knowledge about. However, most don’t know how their work is situated in the larger context of the entire solution landscape. This leads to inconsistencies between both designers internally and when meeting users to discuss solutions. By better understanding the larger context, some users hypothesised that projects could be better defined and have more focused design processes.

Beyond interpersonal communication and understanding, several users wished to have a better understanding for their own sake. Junior designers did not have an entire picture of the landscape and were anxious about not being aware of existing solutions. Senior designers that already had a larger picture did not have up-to-date information and had no simple way of checking (Table 3).

|  |  |
| --- | --- |
| **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.

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 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).

|  |
| --- |
| **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.

There is a need for consistent access to information about software solutions 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 overlaps (Table 5).

|  |
| --- |
| **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.

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)?

|  |
| --- |
| **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.

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 was often not created by the UX team but were the result of thr work of external consultants leading to a situation with no clear mapping owner once the project had been delivered. Lastly, the largest factor in why no mapping has been maintained is that input mechanisms were manual requiring many work hours spent to keep the mapping up to date (Table 7).

|  |
| --- |
| **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.

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

|  |
| --- |
| **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.

## 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 8. Sketch prototypes

After ideation and sketching (Figure 8), 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 9). 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, 2023a) 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 9. 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 10).

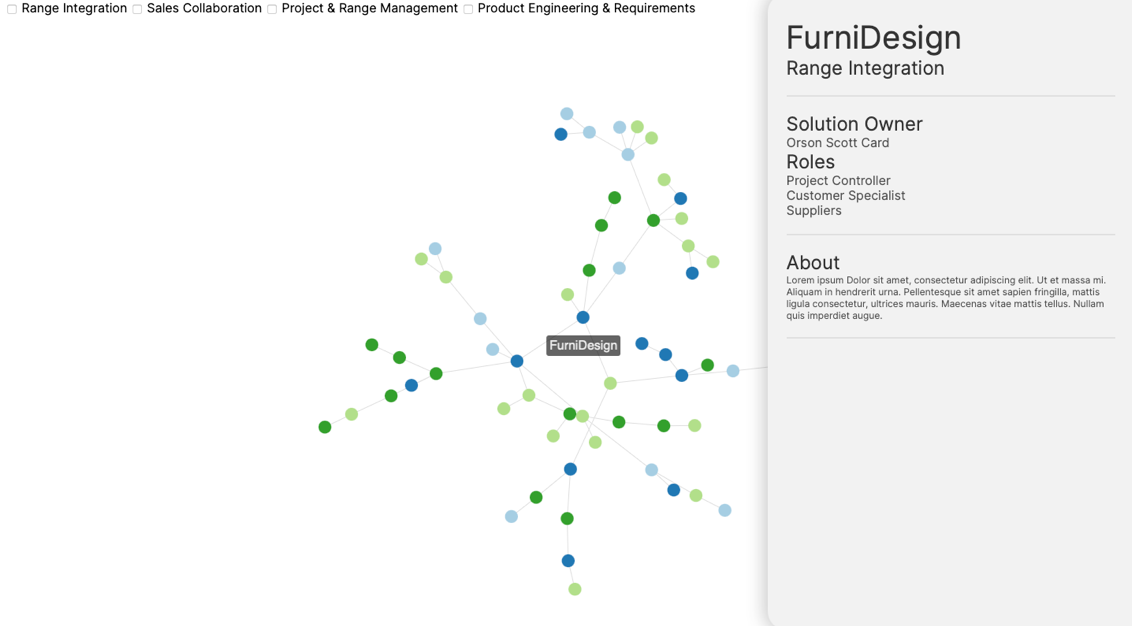


Figure 10. Early functional prototype for the first user tests

The data used in the public prototype (Ingelsten, 2023)(Figure 10) 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 11). 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 11. 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 12. 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 13. 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 14. Gen 4 unfiltered (left) and filtered (right).

#### Figma prototypes

Diagram

Description automatically generated

Figure 15. 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 15, 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

The following section will briefly outline the proof-of-concept prototype presented to stakeholders.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 16. The primary environment of the application.

The main screen displays an overview of the entire digital landscape network (Figure 16). From this screen, the user can pan, zoom, move the solutions around, click on the solutions, and filter solutions after the sections at the top of the screen.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 17. Information about each solution is in a selectable drawer.

Hovering over a solution displays its name more clearly in a popup box, and clicking on it opens a dialogue with detailed information (Figure 17). The detailed dialogue displays information most relevant to the UX Designers: solution name, area, solution owner, user roles, description & processes, and ingoing/outgoing data.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 18. Every solution sends and receives different types of data.

Hovering the links that connect solutions displays what data is being sent in a popup dialogue. The arrow at the end of the link displays the direction of the data flow (Figure 18).

A screenshot of a computer

Description automatically generated with medium confidence

Figure 19. Filters based on user research.

The buttons on the top of the screen open filtering options for the user to pick between (Figure 19). Selected filters are displayed in the dialogue *Active Filters*. The filtered solutions that fill the filter requirements are displayed in the *Filtered Solutions* dialogue. To remove filters, users can either unselect them in the respective filter dialogues or clear them all through the *Clear Filter* button.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 20. Multiple methods to gain access to information.

Clicking the filtered solutions is another way for users to open the detailed information dialogue (Figure 20).

A screenshot of a computer

Description automatically generated with medium confidence

Figure 21. Meta-information about the application to help users.

Hovering the question mark icon at the bottom of the screen opens a dialogue that details information about the application’s interactions and a legend on the solution area colour coding (Figure 21).

We can further explain the PoC through Huron et al.’s (2014) Constructive Visualisation components and processes (Table 10):

|  |  |
| --- | --- |
| **Components** | **Description** |
| *Base token* | Each solution node in the interface represents a base token. The links are another token that connects nodes. |
| *Token grammar* | Nodes and links. Nodes have different colours and name tags. Links have directionality according to the flow of data. |
| *Environment* | The environment is the 2D space in which the nodes are. |
| *Assembly model* | Nodes are sectioned by *area* and are linked together according to the *data flow*. |
| **Processes** |  |
| *Environment initialisation*. | An interactive environment with graphical representations of each solution. |
| *Mapping data to tokens*. | Tokens are divided into relative information of each solution and are only connected through a data connection. |
| *Assembling the tokens*. | The visualisation shows users how each solution fits into a larger context. |
| *Evolution over time*. | The visualisation updates dynamically as data is changed and updated, either manually or through a database. (**Not implemented**) |

Table 10. The PoC described through Huron et al.'s (2014) components & processes.

# Discussion

## Answering research questions

The research questions posed in the beginning are as follows:

1. How might we apply a user-centred design approach when designing data visualisations?
2. What are the implications of designing a user-centred data interface for Interaction Design and Visualisation Design?
3. Who owns and maintains software after a design process like this, and whom does it affect?

It is possible to design rich visualisations that provide real value to users by applying standard Interaction Design methodologies, such as interviews, prototyping, and user testing. Through insights gained in the design process, Interaction Designers can contextualise existing data to generate user value. However, insights gained through the design process provide value beyond the immediate need for visualisation by designing a Constructive Visualisation described by Huron et al. (2014), Interaction Designers can empower users to take control of their needs and use visualisations with *intent*.

As Andrienko et al., (2020) posed a need for specialised user-centred designs in VIS. I see an opportunity for Interaction Design to bridge this gap and provide methodology and projects. Furthermore, given the recent rise in AI technologies and general advancement in technology, there will be a need to understand better *how* to visualise data rather than *what* data is to be visualised. The proof of concept presented in this thesis is part of what can be a viable method for designing new data interfaces from a user-centred perspective.

Building upon the concepts and ideas presented by Andrienko et al. (2020) I have found that data visualisation is mainly a means to an end. Large organisations like IKEA have vast amounts of data in various tables and databases, making it easy for computers to parse and search. However, where does the human come in that equation?

Visualising this data require some user to interpret, analyse, and recontextualise the data. The critical part is the importance of contextualising data. When you understand the user, their needs, how they work, and why they do what they do, you can make an informed decision on contextualising relevant visualisations. The way you construct the visualisation itself comes secondary to this. Different user needs require different types of visualisations. In this case, a force-directed graph was the most fitting visualisation for the users’ needs.

Furthermore, the effects of conducting Interaction Design research this way have a larger span than the project's scope. Users began reflecting upon and questioning the structures in which they work, most of which they have taken for granted. A discussion was brought forward about how IKEA approaches software governance and data management by designing a visualisation in a small part of the organisation. A discussion, if taken seriously, could have a significant impact on their digital footprint as they implement new solutions.

## The Proof-of-Concept and Literature

In the following section of the discussion, I will look at the proof of concept and see how it relates to literature and related works.

Firstly, how does the PoC relate to existing visualisation software? Five key aspects could be identified (4.2.4 Summary) that provided a direction for the design. In total, the PoC fills four out of five of these guidelines. The PoC:

1. Represents relevant data on an interactive 2D interface.
2. Does **not** update data dynamically. It picks data from a predetermined dataset.
3. Divides data into solution nodes connected by data links.
4. Allows users to filter a selection of solutions after relevant metrics.
5. Displays varying levels of information depending on user interaction.

Given this thesis's scope and technical restraints, guideline (2) could not be implemented. Furthermore, implementing such a system would require restructuring how data is handled in PoC code and at IKEA. It was, however, discussed in depth with users and is intended to be an integral part of the concept. A discussion of how (2) would impact the user experience will be developed further in *Ethics*, *Governance*, and *Future work*.

Secondly, an analysis of the PoC and the related works through the lens of Huron et al.’s (2014) design challenges (Figure 5, Table 11):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Visualisation application* | *Keep it simple* | *Enabling expressivity* | *Incorporating data dynamics* | *Manipulation of Visualisation* | *Skills learning* |
| PoC | ✓ | ✗ (✓) | ✗ (✓) | Indirect | Easy |
| Connected Papers | ✓ | ✗ | ✓ | Indirect | Easy |
| Google Maps | ✓ | ✗ | ✓ | Indirect | Easy |
| Sense.us | ✗ | ✓ | ✗ (✓) | Indirect | Unknown |
| Constructive Visualisation | ✓ | ✓ | ✓ | Direct | Unknown |

Table 11. How do the PoC and related works relate to Huron et al.'s design challenges? ✗ (✓) signifies a half-implementation.

The PoC, as it stands, does not fill the requirements for a Constructive Visualisation. However, neither do any of the other related applications. However, what would make the concept a Constructive Visualisation? As previously mentioned, implementing a back-end system that allows for dynamic data updates would push it in the right direction. In addition, a greater expressivity would be achieved if the PoC allowed for greater user control in the visualisation design at runtime, e.g., letting users change data, manipulate connections, and tailor custom filters. These additions would take the PoC close to what Huron et al. (2014) calls a Constructive Visualisation, but to fully embrace their idea, the PoC would have to allow the user an even greater agency in changing and adding to the application (more akin to *sense.us* (Heer et al., 2007)).

* A link to Dimara & Perin 2020 as well?
  + Action, Action-reaction, Reaction perceived as such, Data-related intent.
  + “*Interaction for VIS 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*”

## Design Process

### Pivoting to coding

Initially, there were no plans to code an implementation prototype but keep all prototyping strictly visual. The initial fear was that I would encounter some technical hurdle too large to solve and spend too much time working on that instead of engaging with users and prototyping.

Before going down that path, I did some technical exercises to check the viability of coding a prototype. Luckily, I found the *force-graph* (Asturiano, 2018/2023) library and found that it was easy enough to work with and provided me with enough flexibility to create my visualisations. Pivoting to coding a prototype resonated with users when testing as well. The Figma prototypes provided a visual aid and spurred discussions but would now, in retrospect, have needed more depth. Designers and non-designers who interacted with the live prototype better understood the concept's purpose and future possibilities.

### 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 mean discussions that stayed within a tighter scope and were more structured.

Although recording each session allowed for more engaging discussions, the 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. In retrospect, this is okay as it demonstrates commonalities between the users and a direction for the project. More specific questions could have been asked if I had staggered the interviews weekly.

### Reflections on user testing

User testing was mainly conducted through one main prototype, which was iterated upon. In addition, smaller exploratory prototypes were done in parallel that fed into the main prototype.

Choosing this path instead of a broader exploratory prototyping method is due to two main factors. Firstly, the expected outcome of the stakeholder was to have a concept that was as “production ready” as possible, which prompted designing around a singular concept. Secondly, inspiration prompted by Fisher’s (Andrienko et al., 2020) thoughts on the future of data visualisation. Namely, to reinvigorate and find new ways of visualisation, there should be designs that follow a user-centric ideology and aim to be as focused as possible. We can conclude and take learnings into other projects by solving specific user issues. I found this perspective interesting, and it begs the question, *if you design for everyone, are you even designing for anyone*?

All users tested the prototype only once (disregarding the supervisor, who was briefed more often) and tested different iterations. Doing so provided some benefits but is not without fault. On the positive side, moving to a new tester between iterations allowed for fresh perspectives each time. In addition, since it was the testers’ first time experiencing the prototype, they picked up on issues not caught in earlier iterations and provided further feedback regarding future implementations. However, testing the prototypes with users at least once more would have allowed for a greater re-connection with the users' previous feedback in newer iterations. This could have led to users building upon their feedback to develop some ideas and interactions further and allow the users to see the progression of the design.

## Stakeholder

### Prototype data faked

Due to organisational secrecy, the data used in the prototypes is entirely fake and fabricated. However, testing with internal users allowed me to use actual data. Prototype testing was conducted through a publicly accessible website. Therefore data had to be sanitised to resemble the original data set rather than represent it. To counteract possible user input regarding this, 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”.

This approach worked well with the UX Designer users, who are accustomed to this type of concession. However, when presenting the proof of concept for users in more managerial/non-UX positions, this method proved more challenging to demonstrate as feedback often circled back to the need for real data. Thankfully, an earlier iteration had real data implemented and could be shown, which gave these users a more tangible experience.

This experience, however, further demonstrates the value of getting to know and understand your stakeholders. You can never make assumptions about how the user will react. You might hypothesise, but no one other than the user knows how they will react. A deeper understanding and knowledge of how the user’s intent and expectations can avoid situations like the one explained above and be more efficient.

### Access to users

A benefit of working with a company is that the end user group is predefined initially. Similarly, the user needs and scope were easy to determine, given the problem statement issued by the company. These benefits that come with companies offload the work of finding and contacting users at the cost of flexibility.

However, working with a company introduces other factors that need to be considered. As an external observer, you must be adaptable to and aware of the culture you are researching, like ethnographic research. Every culture (in this case, a company) has specific social structures and interaction methods. Meeting users through their expected methods makes facilitating connection easier, easing the research process. At IKEA, this entailed learning and using their preferred methods of communication, namely Microsoft Teams, e-mail, and, most importantly, meeting in person at the offices in Älmhult.

However, adapting to their preferred interaction methods resulted in other issues to consider when planning. Firstly, gaining access to the digital systems required to communicate entailed a three-week-long process with IT Support, which in the end did not work. To work with this limitation, I could book meetings with the help of the supervisor and use my e-mail to contact users. Similarly, access to solution documentation was included in the technical limitations, prompting the supervisor to assist with access to proper documentation. Secondly, booking meetings with users required at least two weeks of planning beforehand to secure multiple users on the same day. This is fine since I, as an outsider, am asking for their time at their place of work, but it did require rigorous planning.

### A balance between stakeholder and thesis

As discussed previously, working with a company has both benefits and drawbacks. On a different level, a delicate balance needs to be struck between stakeholders, meaning I need to take on different roles. For example, I am, in part, a designer researching a project for IKEA, a bachelor's student at Malmö University, and a stakeholder in my interests and development. All this means that collaborating with an outer stakeholder requires balancing expectations.

Early in the process, I had discussions with the supervisor from IKEA about limitations and expectations from the university regarding thesis projects and related them to IKEA’s expectations. We could then compromise together and define a scope that satisfied the needs of both parties. Transparent communication and mutual respect were crucial for successful cooperation.

Engaging in a project like this as part of a thesis presents an interesting perspective on the role of an Interaction Designer. Namely, *how does it impact the designer and the stakeholder dynamic when the designer has an intrinsic stake in the project*? I am, in fact, not an unbiased third party with altruistic ambitions to improve IKEA (even though that would be nice). Instead, much of my interest lies in finishing the thesis and graduating. This interest divide did not cause significant friction between the stakeholders and me. However, it was ever-present when discussing a design hand-off or other future prototype implementations outside the project scope, some of which will be discussed in Future work.

### User groups

The primary user group on this project consists of UX Designers. User testing benefited from the users’ expertise in design. Discussing prototypes from their perspective as UX Designers, feedback included practical aspects such as visuals, interactions, user flow, and accessibility. In addition, 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 in this way can yield results that would otherwise be missed if you held a more traditional view of the user/designer relationship.

The primary users that conducted user tests were the UX Designers. However, some others were also included with varying stakes and expectations.

Firstly, the supervisor with a Lead UX position has approached the project and design from a different perspective than the other “regular” UX Designers. Understandably, since they have a role that requires more responsibility, the mapping solution constitutes a complete birds-eye overview (compared to the comparatively zoomed-in perspective of the UX Designers).

Secondly, the manager, “one step up” from the supervisor, is responsible for several tech and design teams. They provided a fresh perspective, not from a UX point of view but from a business perspective. Their input was seldom regarding interactions or specific user flows but instead on purpose and possibilities, which proved valuable when ideating. Similarly, another lead, “one step up” from the manager, was present at the final presentation and provided similar business perspectives.

## Governance

Another aspect uncovered through the design process is the concept of *governance*. The PoC has provided a new perspective on a real issue that the stakeholder has, but who *owns* it and who is *responsible* for developing it? As companies grow larger, so does their digital footprint, further worsening software governance. At a certain point, the company behaves more like a single organism where you cannot point out one individual solution as vital, but all solutions are interconnected.

The number of solutions present at IKEA has led to a situation where no one person can grasp the entire landscape. This presents itself as a severe governance issue, as previously discussed, which can, and has, led to wasted money and time. The UX Designers' work and the SDG (*Software Development Governance*) at IKEA are closely intertwined. The designers do not own the solutions but are integral to developing new and existing solutions. This creates an interesting symbiotic relationship between the two parties but leaves the governance unclear.

The governance of the thesis has had me as the lead author and designer with the stakeholder in a position of examination. However, what happens when the project ends? There will be a hand-off of design documents and insights to relevant parties at the company, but my involvement will leave a vacuum in governance. If an interest in the concept persists, a more considerable effort would be required by IKEA to greenlight a new development process and assign a new SDG.

Another aspect that arises as a governance effect is who becomes *accountable* for software. Software changes, and people move on to other projects. This creates a *Ship of Theseus* problem in the accountability and governance of software. Who is accountable when all the original designers have moved on, and the software has updated beyond its original scope? These issues are easy to track on a micro-scale. However, on the scale that IKEA operates at, it increases in complexity exponentially, leading to the current situation where no one knows how it all fits together.

Beyond affecting business goals and organisational planning, governance affects people on every company level. Moreover, it leads to some ethical dilemmas, which will be discussed in the next section.

## Ethics

The implications of a design like this are much more significant than the direct impact it can have on a designer's workflow. It discusses data accessibility, equality, empowerment, transparency, and accountability.

Should all data in an organisation be accessible by everyone, and who decides that? For example, at IKEA, there is so much data that could be valuable to the UX Designers, but it needs to be clearer by consistent documentation, labelling, and availability. In addition, there is a need for transparency internally.

An insight from user research was that most designers rely on social connections at the office to get the needed information. Social connections are, of course, good but are highly situational and vary from person to person. New designers have a different size of social networks or knowledge of the digital landscape than senior designers do, limiting how they can contribute. This means that junior designers rely heavily on the help of others to do their work, limiting their independence. The demands are equal for all designers, which presents an ethical dilemma, *who is accountable when junior designers make mistakes due to not being aware of better solutions*? Is it the junior designer for not taking the time to learn? What if the onboarding is flawed? Is it the team leader then who is responsible? Or is the organisation at fault for not having clear directives and guides for data access?

User research uncovered this issue, which was widely known among users, but no one was actively involved in solving it. The proposed process and concept do not solve the underlying issue but act to equalise data availability. Consistent and dynamic data interfaces would empower designers to work more independently, make more informed decisions, and act as a method for onboarding new team members.

## Future work

I see opportunities for Interaction Designers and Visualisation Designers to work intersectionally to create better user experiences in data visualisation. Firstly, further research could be conducted on creating dynamic visualisations that empower non-data analysts to build and visualise complex data. Secondly, further research could be conducted on the effects of using a user-centred approach to data accessibility and SDG. Lastly, further material explorations could be made to analyse existing visualisation solutions to explore the design space from an Interaction Design perspective.

From the perspective of IKEA, this thesis is a starting point for a discussion regarding the future of the digital landscape and how it could evolve.

# Conclusion

This thesis has presented a user-centred approach to data visualisation from an Interaction Design perspective. This has been done through analysing related visualisation solutions, related scholarly works, and user research at Inter IKEA. The research resulted in a proof-of-concept design that visualised the digital solution landscape from a UX perspective providing users with a better understanding of the context in which they are working.

Key learnings from this design process can be summarised as follows:

1. Understanding user needs and contextualising data accordingly is critical for a good user experience.
2. The actual visualisation method comes second to user needs and should be designed based on those needs.
3. The ethical implications of a user-centred visualisation can empower users and promote equality.
4. I had one more but I forgot it so this here is TEMP atm.

# References

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

Ammar, W., Groeneveld, D., Bhagavatula, C., Beltagy, I., Crawford, M., Downey, D., Dunkelberger, J., Elgohary, A., Feldman, S., Ha, V., Kinney, R., Kohlmeier, S., Lo, K., Murray, T., Ooi, H.-H., Peters, M., Power, J., Skjonsberg, S., Wang, L. L., … Etzioni, O. (2018). *Construction of the Literature Graph in Semantic Scholar* (arXiv:1805.02262). arXiv. http://arxiv.org/abs/1805.02262

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

Apple. (2023). *IOS - Health*. Apple. https://www.apple.com/ios/health/

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

Bertin, J., & Barbut, M. (1968). *Sémiologie graphique: Les diagrammes, les réseaux, les cartes*. La Haye, Mouton.

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

Chulani, S., Williams, C., & Yaeli, A. (2008). Software development governance and its concerns. *Proceedings of the 1st International Workshop on Software Development Governance*, 3–6. https://doi.org/10.1145/1370720.1370723

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. (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

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/

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

Few, S. (2023). *Data Visualization for Human Perception*. Interaction Design Foundation. https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/data-visualization-for-human-perception

Fitbit. (2023). *Fitbit*. Official Site for Activity Trackers and More. https://www.fitbit.com/global/se/home

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

Fry, B. (2004). *Computational Information Design* [PhD]. Massachusetts Institute of Technology.

Fry, B. (2008). *Visualizing data*. O’Reilly Media, Inc.

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

Google. (2023b). *Google Sheets*. Online Spreadsheet Editor | Google Workspace. https://www.facebook.com/GoogleDocs/

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

Home Assistant. (2023). *Home Assistant*. Home Assistant. https://www.home-assistant.io/

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

Komoot. (2023). *Komoot | Find, plan and share your adventures*. Komoot. https://www.komoot.com

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

Lima, M. (2023). *Visualcomplexity.com*. A Visual Exploration on Mapping Complex Networks. http://www.visualcomplexity.com/vc/

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

Microsoft. (2023a). *Microsoft Excel*. Kalkylprogram | Microsoft 365. https://www.microsoft.com/sv-se/microsoft-365/excel

Microsoft. (2023b). *Microsoft Power BI*. Datavisualisering. https://powerbi.microsoft.com/sv-se/

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.

On The Go Map. (2023). *On The Go Map*. http://onthegomap.com/

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

Russel, E. (2019, October 1). *9 things to know about Google’s maps data: Beyond the Map*. Google Cloud Blog. https://cloud.google.com/blog/products/maps-platform/9-things-know-about-googles-maps-data-beyond-map

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

Shneiderman, B. (1996). *The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations*.

Strava. (2023). *Strava | Running, Cycling & Hiking App—Train, Track & Share*. https://www.strava.com/

Tableau. (2023). *Tableau: Business Intelligence and Analytics Software*. Tableau. https://www.tableau.com/node/62770

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