A process for tangible Design System contributions

Hyper Island UK
MA Digital Management
Industry Research Project

November 2021
13.766 words

Caio Salles Manzotti

Table of contents

1. Background

1.1 Introduction

The increasing popularity of Design Thinking, the Google Design Sprint, and other frameworks helped design gain more attention from companies worldwide. In addition, the collaborative nature of these frameworks increased the understanding that design could contribute to businesses in a meaningful and tangible way. Designers started to be seen as facilitators and specialists that can connect multiple stakeholders and end-users to solve a problem (Aronowitz, 2018). As design maturity grows, it permeates the company environment at all levels, including senior management. This is demonstrated by companies adopting design a competitive strategy (Kuang and Fabricant, 2019), up to a point where designers also get appointed to executive-level positions. While this can take years to happen in a traditional company, many startups are founded with the design practice already integrated into their way of working and culture. These companies tend to push conventional boundaries to grow as fast as possible, especially regarding process maturity (Sutton, 2000). Methodologies such as Agile have been adopted since the beginning to adapt ways of working in a faster and flexible way.

This combination of design and development frameworks have numerous benefits when it comes to collaboration. For example, Ferreira et al. (2007) identified that Agile methods could fundamentally change the way developers and designers communicate, reinforcing rituals and improving their relationships. But there are still gaps between these two areas, one of them being that these frameworks have no overlap (Fig 1). The ones in the design space primarily focus on working in the discovery and problem spaces, while the Agile ones focus on delivering solutions as fast as possible (Gurusamy et al. 2016). The bigger this gap becomes, the less a product experience is consistent over the whole user journey and that the brand is represented correctly. Therefore, a different approach is needed (Perez Cruz, 2019).

Large enterprises and startups have been exploring ways to bridge this gap. For years, solutions such as interface pattern libraries and brand style guides oriented designers and developers on implementing solutions. But when considering the current complex nature of modern software development, these solutions are not enough. As a result, companies such as Airbnb, Uber, and IBM have invested in a different solution, adopting what is known as a Design

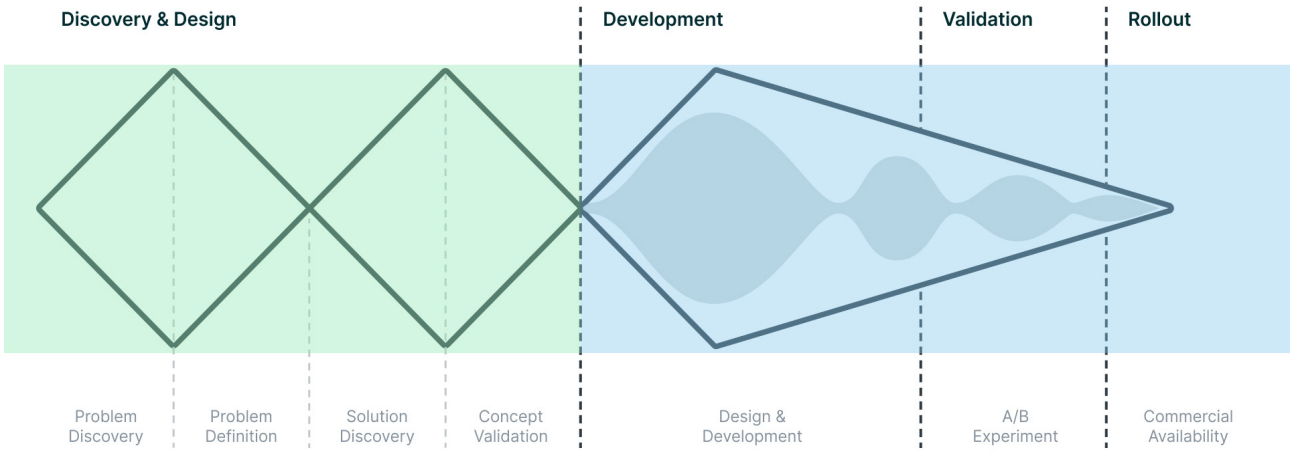

Figure 1 - Design Thinking (green) and Agile focus areas (blue), using the Zendesk Triple Diamond as reference.
Source: https://medium.com/zendesk-creative-blog/the-zendesk-triple-diamond-process-fd857a11c179

System (DS). According to Kholmatova (2017), a DS is "a set of interconnected patterns and shared practices coherently organised to serve the purpose of a digital product.". The author explains that these patterns are the repeatable elements that, when combined, create a digital interface. Examples can be buttons, text fields, icons, colour palettes and flows. The shared practices relate to how patterns are created, used and shared among teams.

Adopting a DS by itself, it's not a guarantee of success. It needs to be adapted and organised in a way that works for the company. Therefore, this study will focus on the challenges of maturing a DS and further adapting it to the company context, more specifically towards contribution models. An approach that can potentially expand the system's capabilities while maintaining its core objective of connecting designers and developers.

1.2 Thesis structure

The thesis content is presented in two main sections. The first one is composed of an introduction to the problem in the workspace, followed by the literature review, an analysis of the evolution of design patterns, from early concepts to the more modern approaches used by digital products existing today. This section also includes an explanation of the benefits of a DS and the ethical considerations regarding the approach. The second part of this study presents a way of solving the problem in the author's workplace. In this section, a new process will be researched, iterated and validated, with the final objective of implementing a proof of concept, the first of its kind in the company.

1.3 Key terms

The following terms are essential for the understanding of the context of DS and modern software development practices:

Product Designer: A designer who uses user-centric methods and processes to create solutions. Its responsibilities usually range from uncovering user problems to delivering ready to build designs.

Developer: Professionals responsible for building interfaces for software applications, they implement design solutions in platforms such as for Apple iOS, Google's Android and web browsers.

Digital Product: In this context, it refers to digital software products. Those are created to transform a user-specific need into a business opportunity, such as streaming movies, buying groceries online, among other activities.

Software as a Service (SaaS): A way of delivering digital products to end customers. In this model, the software is centrally hosted and is distributed on a subscription basis. This approach also allows for constant software updates without the need for a customer to buy a new version.

Design System (DS): A set of interconnected patterns and shared practices coherently organised to serve the purpose of a digital product. A combination of pattern libraries, style

guides, principles, rules and guidelines, that allows designers and developers to dedicate more time discussing and evaluating the right problem instead of how an interface will be built (Perez Cruz, 2019).

Product Teams: A multi-practice team composed of specialists in design, technology and product management. It is usually responsible for a section of a digital product, and its responsibility is to deliver features and improvements into the user flow.

Design System Team (DS Team): Very similar to the product teams, but instead of creating solutions directly to end-users, it is responsible for creating and maintaining an internal product that Product Teams use to standardise and accelerate their work.

1.4 Scope of the research

This thesis is an activity for Hyper Island's Digital Management Masters program. Its objective is to demonstrate a systemic and critical understanding of digital management, while also showing originality in applying professional and academic knowledge to solve a problem in the workspace. This research will occur in the author's workplace, Adevinta. As standard company practice, the insights and learnings will be implemented first as a proof of concept in one of the company markets, before being scaled to the whole company. In the case of this research, it will be performed in the DS currently implemented in the Benelux market, represented by its two products Markplaats and 2dehands.

The outcome of this research will be a new section into the existing HDS documentation (Fig 2), containing process diagrams and guidelines on how designers from product teams can contribute to the HDS. This section will be complemented by practical guidelines on acting in every ramification of these processes, so teams can have autonomy on moving forward.

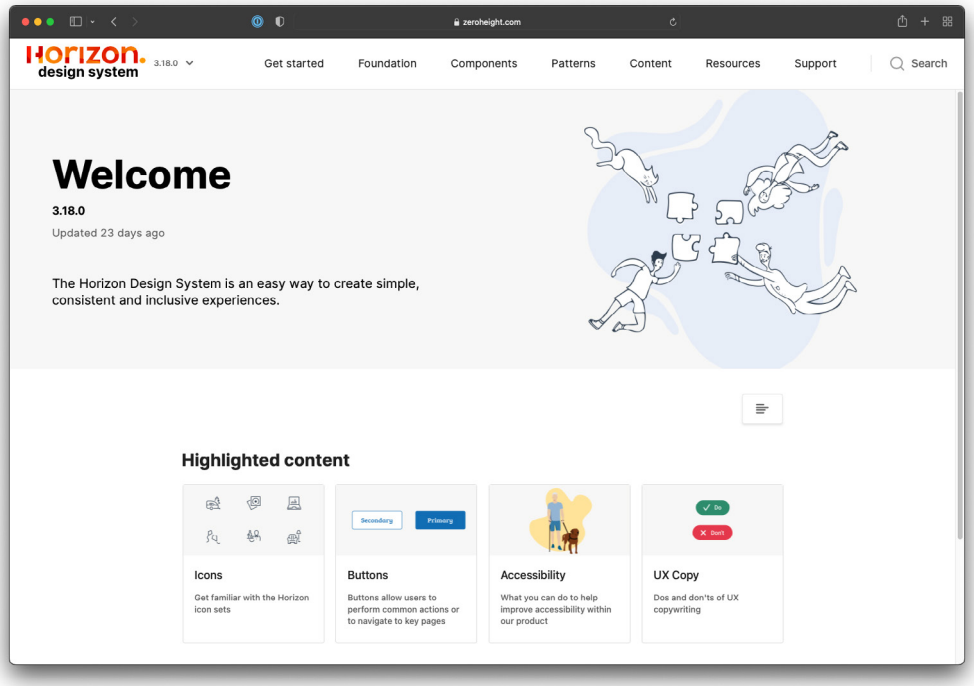

Figure 2 - HDS documentation.

1.5 Research question

Design and collaboration should go hand in hand. Frameworks such as Design Thinking thrive on the contribution between stakeholders. But for a DS, this is not usually the case. Instead, most DS Teams control what changes can be made into a DS in a very centralised way. This mindset probably comes from System Thinking principles, where authors such as Meadows and Wright (2008) affirm that for outsiders, it can be quite challenging to comprehend the interconnections of a system, making it difficult to anticipate what impact a change may have over the whole system. While this is probably true, it should not prevent collaboration from happening.

This research will examine the following question: How to implement a collaborative process where Product Teams are self-guided into creating tangible contributions to a DS? An answer that will be pursued by mitigating the risks of decentralising a DS, exploring the motivations from these teams to contribute and what they will get in return for their collaboration.

1.6 Thesis objectives

Design as a practice has been evolving in an accelerated rhythm in recent years (Buley, 2019). However, as with any other practice, this evolution is usually supported by tools and processes. This study aims to create new company knowledge on how to mature an internal DS and expand it as a product by exploring more modern and collaborative processes. This initiative is part of a more comprehensive strategy that aims to elevate the company's maturity of design as a whole. An effort that started back in 2017, when design leaders across the company got together to trace a long term design maturity plan, after identifying a deficit in how the company designers worked compared to its competitors (Buley, 2017).

This new approach will allow Product Teams to have more autonomy over contributions to the DS, by decentralising its core and reducing the dependency on the DS Team. The project will then run for the defined period of one quarter (Q4/2021). Once this period is over, the results, next steps and rollout plan will be collected and presented to the company's executive team. If successful, the process will receive funds and will be scaled to a global level.

1.7 Problem in the workplace

1.7.1 About the company

This study will take place at Adevinta, a company that owns and operates an international group of local brands. As of June 20201, it merged with eBay Classifieds group. The combination of these two portfolios created an expressive global portfolio of digital online classifieds products in 16 countries, with an average of 1.5 billion visits and 250 million unique users each month (Adevinta, 2021). All these products serve the same purpose of connecting buyers and sellers in their markets. The author of this research is responsible for maintaining a local design system that supports two of the company's products: Markplaats and 2dehands (Fig 3).

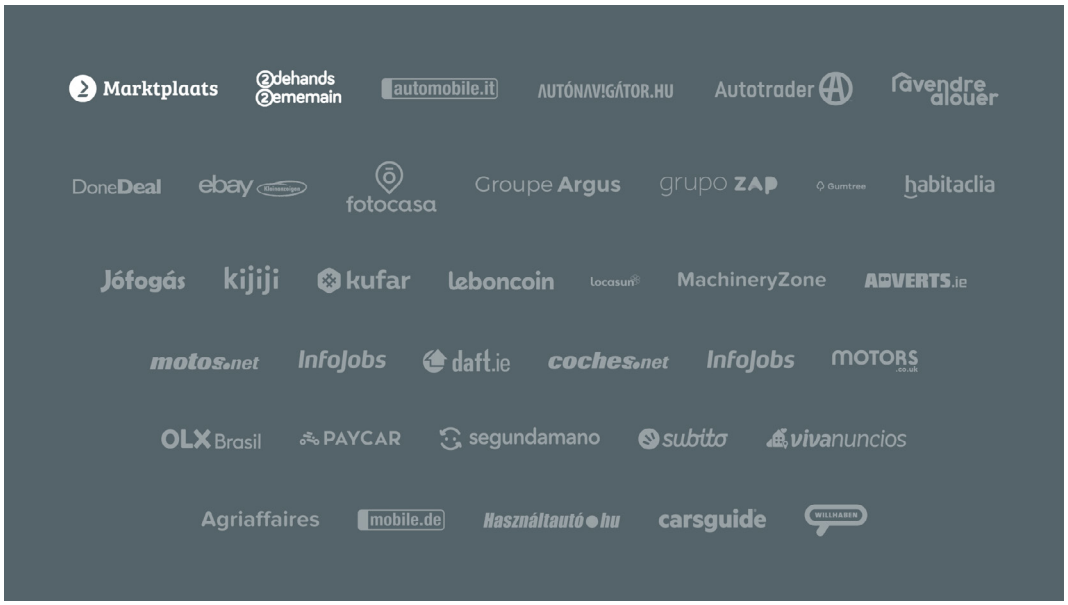

Figure 3 - Adevinta's global product portfolio with the local Benelux products highlighted.
Source: https://www.adevinta.com/brands/

1.7.2 The Horizon design system

As a Lead Product Designer, I led the efforts into building the company's first DS for the Benelux market, the Horizon design system (HDS). The main objectives behind this initiative were two: First, to create a scalable and manageable single source of truth for designers and developers. The second was to anticipate potential development issues during the design phase. In this context, a single source of truth means that the DS becomes the one place that contains all the definitions of how a design should look and behave. It is essential because it can help designers and developers to align their communication by creating a shared common understanding (Fig 4). In addition, anticipating potential development issues while in the design phase, such as lack of proper colour styles or if the length of a translation fits in the width of a button, helps save time and money. Research suggests that the cost of fixing a bug during the design phase is multiple times cheaper than in production (Dawson et al., 2010).

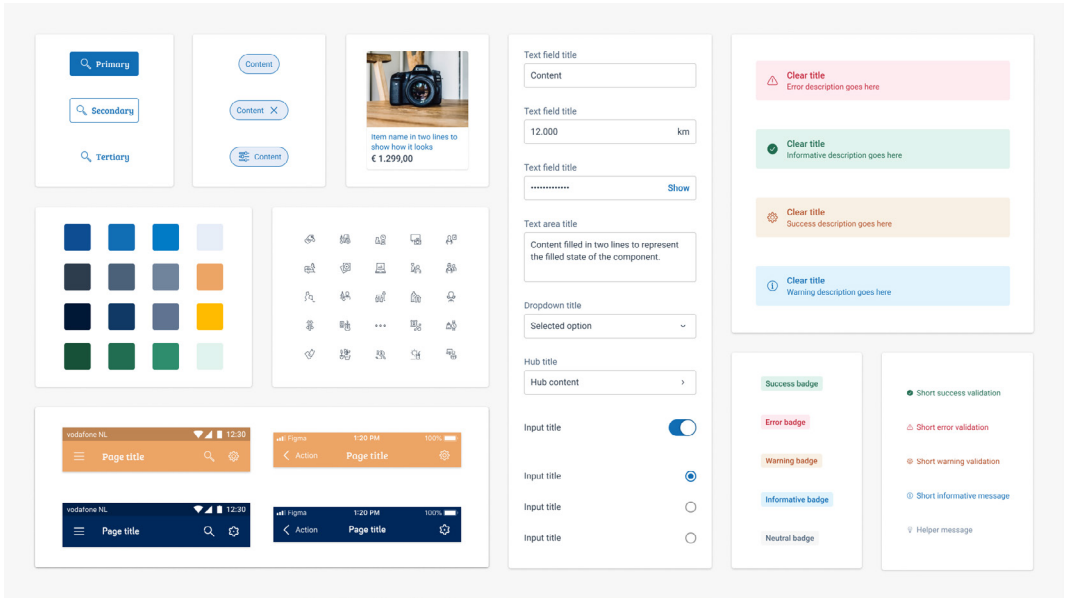

Figure 4 - HDS patterns used by designers and developers to build digital interfaces.

1.7.3 An opportunity to innovate the way we work

The HDS has proven to be a success by overcoming the main challenges for this type of project, such as staffing difficulties, lack of executive championship and low adoption (Sparkbox, 2020). Then, however, an unforeseen event emerged. Product Teams using the HDS started to dedicate time to explore how they wanted the system to behave, expanding its features in ways that would best serve them and providing this information as feedback. Although the DS Team was able to handle these requests at first, with the increased adoption of the HDS, this ended up becoming the bottleneck. This trend started in May, and it kept growing towards August (Fig 5). The length of the period showed that it was not an isolated spike in usage, but an actual increase in adoption by designers and developers. While this can be seen only as a problem at first, it also indicates something positive, that the project is evolving and achieving new maturity levels.

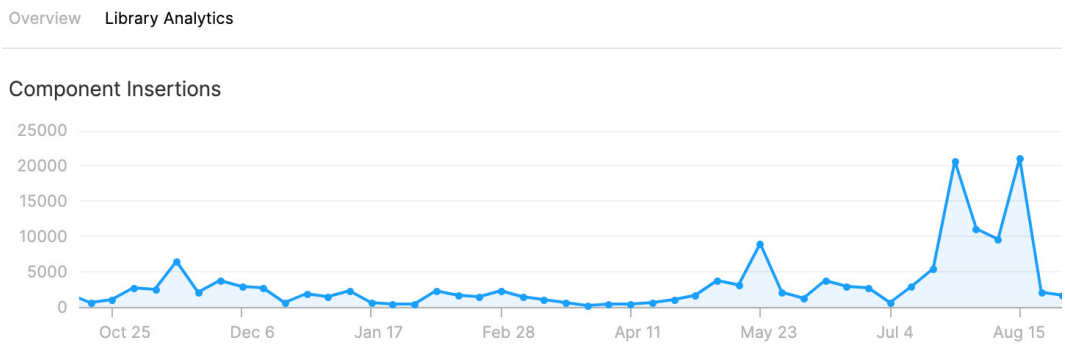

Figure 5 - HDS usage by UX Designers, measured by component insertion in Figma (design tool).

The lack of a contribution process was intentional. Since the beginning, the mindset was that the only ones with enough knowledge to work on the HDS was the DS Team. The main reason to support this was the fact that Product Teams have different objectives. In addition, they are responsible and accountable for maintaining other areas of the product, so they would not have the time or the responsibility towards the HDS. But with the increasing adoption, the DS Team now believes that without tangible contributions from other Product Teams, the HDS will fail, and its adoption will decline. So exploring a process that allows tangible collaborations from Product Teams is not only an opportunity to scale, but also a way of keeping it relevant. At the same time, the team does not intend to decentralise the project entirely, since the risks and effects of decentralising such a complex problem are yet unknown.

1.8 Limitations of the research

This research is limited by its sample and market size. Since, for now, it only considers the company's Benelux market and the Product Teams responsible for the local brands. Also, there is a limitation regarding the duration of the research. Based on previous process changes in the company, it is estimated that a new process would require to run for a period between three to six months to be evaluated appropriately. Since this time will not be available, this research will focus on the implementation of a new process, and its evaluation will occur at a different moment.

2. Literature review

2.1 The evolution of design patterns

DS's are becoming crucial in the way digital products build their solutions. To understand how and why they become so important, it is necessary to go back a few years to a series of events that preceded their existence, since its history starts way before the current modern ways of designing and developing software. This first section of the literature review explores the evolution of design patterns, starting from the early days of personal computers, the advent of web design, and the revolution of modern mobile app interfaces. This evolution overview is inspired by an article from Malone (2017) titled History of Patterns in User Experience Design (Fig 6). Where the author reflects on her teaching experience at California College of the Arts, and how the evolution of design patterns changed the way designers build interfaces.

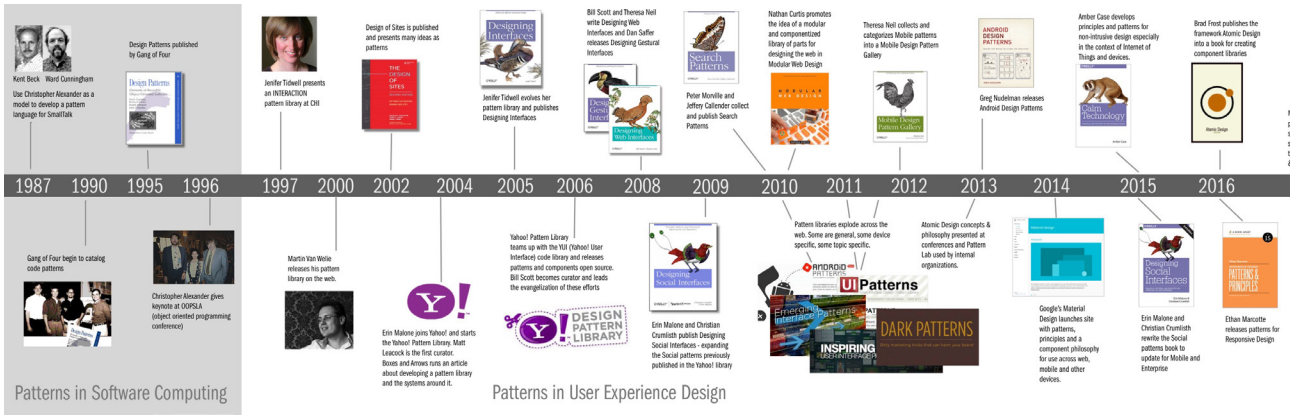

Figure 6 - Section of the History of Patterns in User Experience Design timeline (2017).
Source: http://www.emdezine.com/deziningInteractions/wp-content/uploads/2017/04/HistoryofPatternsTimeline.jpg

2.1.1 The early days of digital interfaces and digital patterns

Introduced in the mid-1960s, the command-line interface (CLI) is a type of operating system (OS) that supports a series of commands via a keyboard in order to perform different tasks. Its usage is limited to linear processing, so to complete a task, a person would input the series of desired commands, one by one, until the task is complete. A CLI requires a digital display and electronic peripherals to work. Its digital interface, usually composed on a dark screen with light tone typography and loading states, visual elements that can be considered rudimentary for today's standards. All of a sudden, this new type of machine handled all sorts of documents and calculations, making analogue typing machines a thing of the past. Furthermore, this was the first interaction with what would become known as the personal computer for many. By the 1980s, CLI's became even more popular, with the launch, in 1981 of the Microsoft Disk Operating System (MS-DOS). An operating system that democratised the usage of personal computers, making its creator, Microsoft, become one of the biggest software companies in the world, a title held until the current days.

However, almost simultaneously with the launch of the MS-DOS, a team of researchers from the Xerox Palo Alto Research Center was about to revolutionise the way people interact with computers. Their research culminated in the creation of the Xerox Star (Fig 7). This new OS introduced the concept of a Graphic user interface (GUI). A type of interface that is

characterised by the usage of visual analogies to represent actions, such as a trash can to indicate where to put files intended for deletion. Instead of only typing commands, its users were also able to point and click with a mouse device, a faster and more intuitive way of interacting with an OS. It quickly became evident that this could be a better way of interacting with computers and digital interfaces. Thus, the Xerox Star paved the path for a massive change in all post-launched OS's. However, being a pioneer in such technology did not guarantee Xeros the leading in the market. Inspired by their product, more prominent companies such as Apple and Microsoft adopted the GUI as a default way of interaction with their versions of OS. Many of the patterns introduced in this era are still being used today, from the usage dropdown menus to folders to represent files structures, demonstrating that design patterns can last a long life.

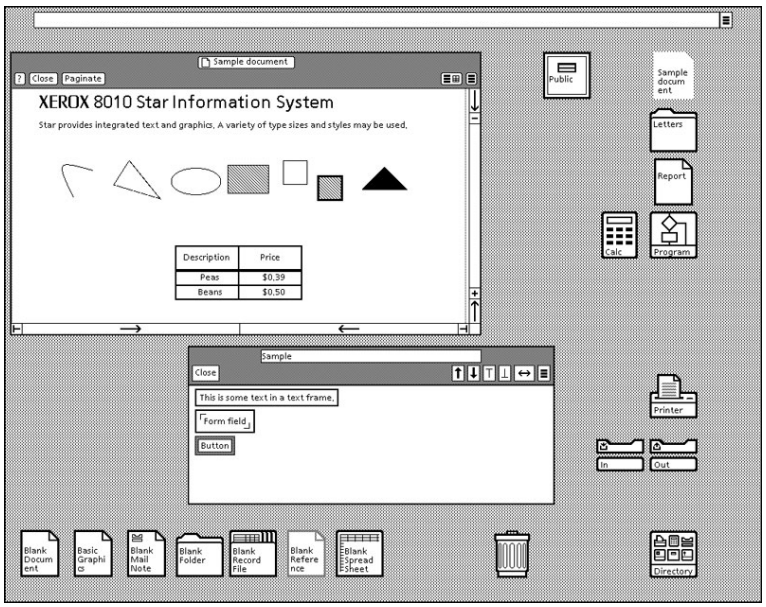

Figure 7 - Xerox Star OS. Source: https://interface-experience.org/objects/xerox-star-8010-information-system/

From the beginning, the creators of these OS's understood the benefit of guidelines, style guides and reusable design patterns, elements that are also base for modern DS's. By the time, the usage of these elements was not due to a lack of creativity or resources. On the contrary, the knowledge that it could help new and existing users interact with their software was already present. It was an intentional move towards standardisation, consistency and usability. Park et al. (2011) affirm that a digital product's usability can benefit from using predefined standards, helping people easily identify patterns across the product. An example of this early knowledge is Apple's Human Interface Guidelines. Launched in 1987 and first published as a printed manual (Fig 8), its objective was to guide developers on how to build software solutions for Apple devices. Elements now used in DS's were already present on these guidelines, such as solid design principles. The manual intended to help developers not only create any software but to build something that would look and feel like part of the OS. In a contemporary reflection, Hodson (2017) affirms that many of the concepts presented in the manual still prevail to this day.

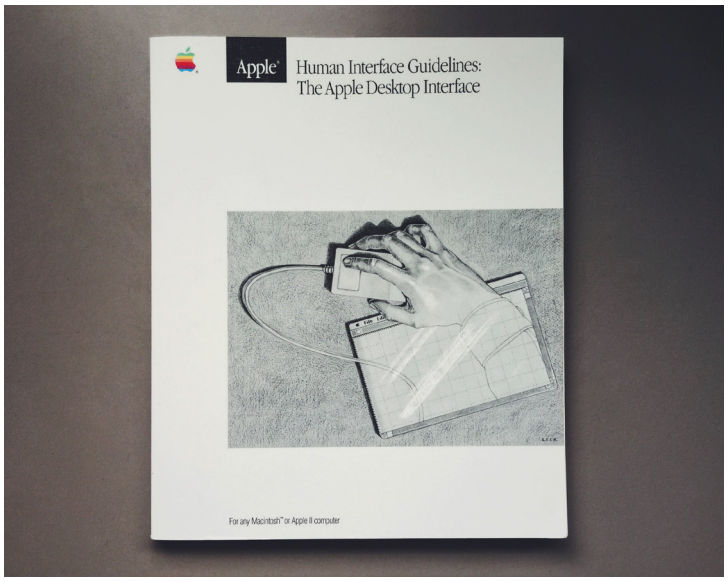

Figure 8 - The Apple Human Interface Guidelines.
Source: https://blog.prototypr.io/rediscovering-apples-human-interface-guidelines-1987-59731376b39e

2.1.2 Adaptations from graphic design and the advent of web design

The migration from command lines to GUI made the interaction with OS's more intuitive and visual. Something that, by the time, was not part of the curriculum from computer science classes. So the ones interested in knowing more about the topic had to look for inspiration in other fields, such as graphic design. Designers were not new to visual systems. For years, they have been doing it for magazines, advertisements, and brand style guides (Fig 9). So the jump into doing the same type of work in software seemed like a natural and exciting path. The GUI was just the beginning of this transformation for designers, going from physical to digital mediums. They were a chance to make solutions more interactable, something not possible with printing. However, in the early 90s, the opportunities were still niched, and the need for software designers was still minimal. The real revolution for these professionals and design patterns came with the advent of the internet.

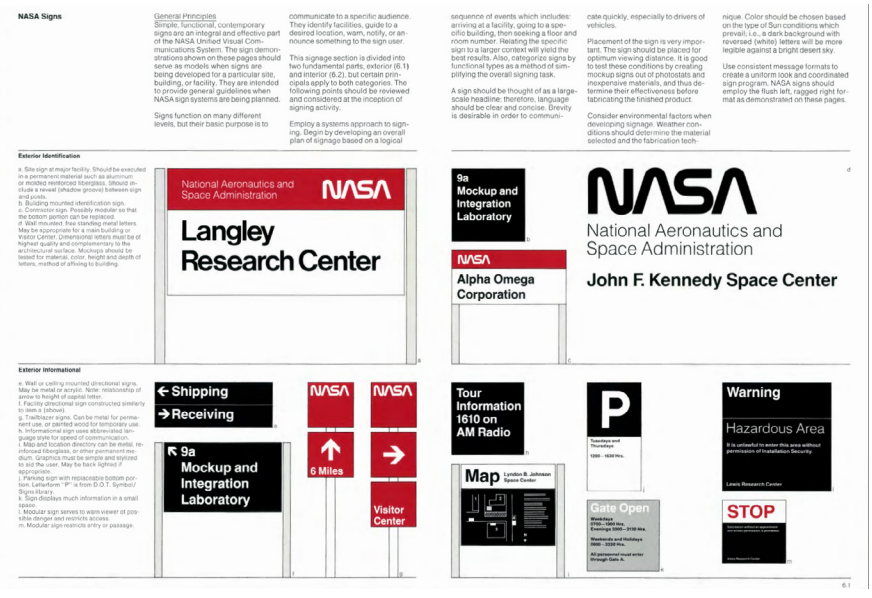

Figure 9 - NASA Graphics Standards Manual.
Source: https://www.nasa.gov/image-feature/nasa-graphics-standards-manual/

The first years of the world wide web were marked by exploration. Inventive websites with no apparent standard started to be seen everywhere. Moreover, technologies such as Macromedia Flash made rich media content available for the masses, meaning that non-static content such as animations, audio and video could now also be added to websites (Bedingfield, 2019). However, once companies started to realise the business opportunities provided by the internet, this scenario started to change. Suddenly new professions such as web designer, interaction designer and information architect started to be seen on job boards. Most taking these jobs were people that come from graphic design jobs. These professionals saw on this opportunity the chance of digitalising their way of work, transforming visual patterns into interactive design patterns.

According to Malone (2017), the first digital design pattern libraries started to be theorised around 1997, with books and open projects starting to be published by the year 2000. However, much of this content intended to tackle the general concept of design patterns and their adaptation for the web. The idea of creating a set of design patterns with one company in mind was not a disseminated idea. However, in a move potentially ahead of its time, Yahoo! decided to create its own web set of standards for design patterns. Introduced in 2004, the Yahoo Design Patterns (Fig 10) contained a series of guidelines for the company's designers and developers, which is maybe the first case of a big corporation using a DS for its digital products.

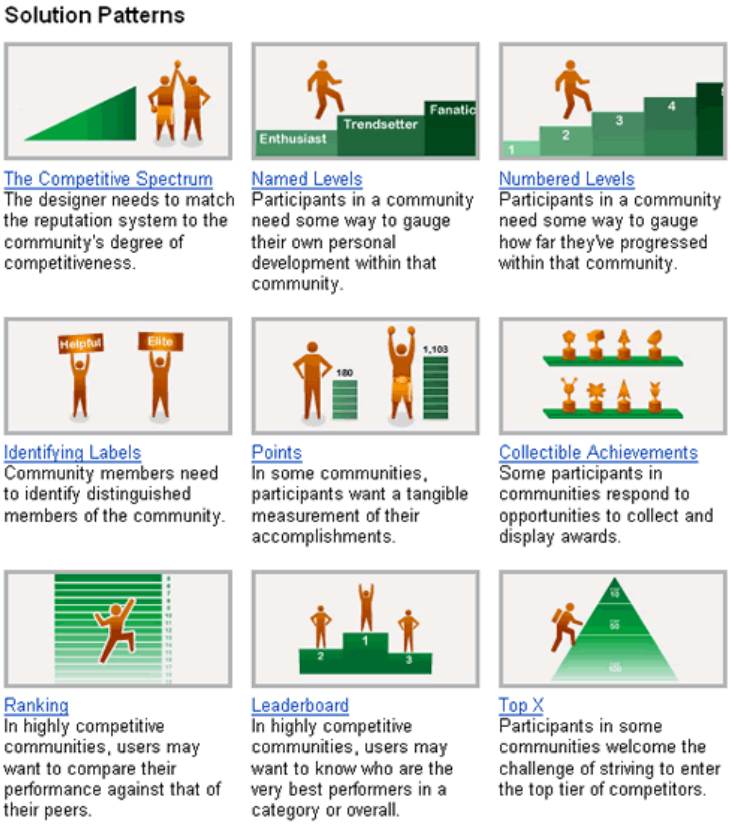

Figure 10 - Yahoo! Design Patterns.
Source: https://thewavingcat.com/2008/06/yahoo-releases-reputation-design-patterns-yay/

In the 2000s, web design continued to evolve and started to adapt to new mediums, such as the growing adoption of smartphones. New technologies known as 'Responsive web design', a term coined by Ethan Marcotte in 2010, helped websites adapt to smartphones and tablets by automatically resizing the content depending on the screen size. This technology was potentially

also responsible for the survival of websites since, with the introduction of smartphones, native mobile applications became available for these mobile users.

When Apple launched its first iPhone in 2007, it completely changed the market. It quickly became a sensation, which led to many companies rushing to adapt their products and services to this new device. Knowing that designers and developers would need guidance, Apple worked on a comprehensive update to its Human Interface Guidelines (Fig 11). The purpose of these guidelines was still the same, but it was adapted to help their audience learn how their design principles and patterns would adapt to the new technology of these devices. Again, a way to make sure that these applications would look and feel like part of their now touch and mobile OS. Tidwell et al. (2020) explain this evolution and adaptability. The author affirms that interface design patterns, like everything else, change to keep up with the changing times. So as new mediums and ways of interacting with interfaces come to life, design patterns will adapt to it.

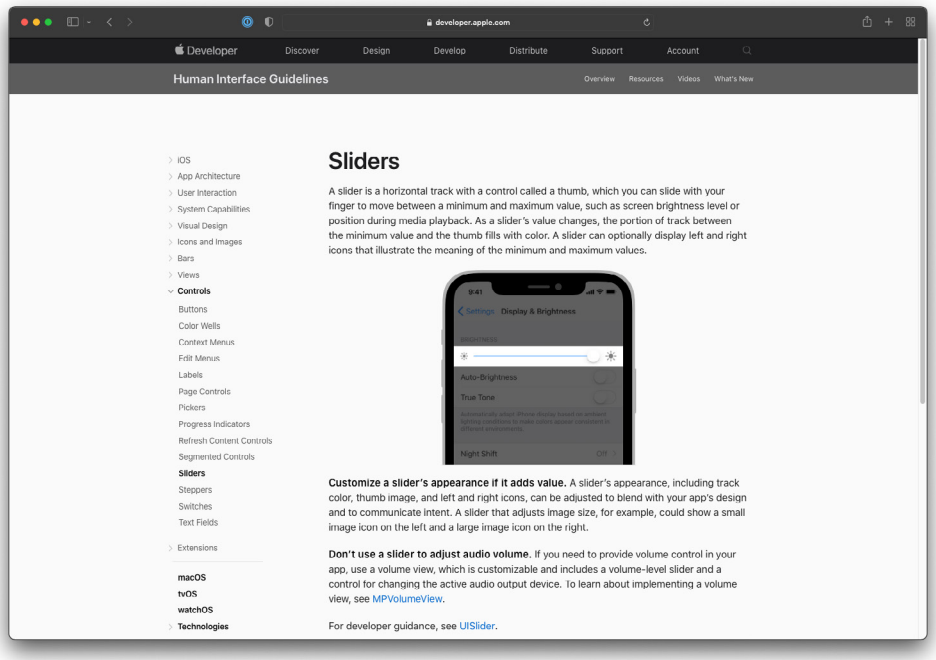

Figure 11 - Current version of the Apple Human Interface Guidelines.
Source: https://thewavingcat.com/2008/06/yahoo-releases-reputation-design-patterns-yay/

2.1.3 The challenges of cross-platform pattern libraries

The usage of mobile devices continued to grow, and companies started to notice the need for cross-device experiences. So in 2014, Google revamped the way design was handled at the company. They decided to align their digital products to be consistent in visual and behaviour. The effort was similar to what Apple tried to do with their Human Interface Guidelines. The main difference is that this solution intended to align these experiences cross-platform, so the company's Android and iOS apps and their website versions were treated as variants as touchpoints of the same digital product. The solution was called Google Material Design (Fig 12). Nowadays, one of the most famous and used DS and one of the first to introduce the concept of cross-platform design patterns.

The introduction of this DS was a significant move because the way people interacted with digital devices was starting to change. Companies like Google started to realise that their users needed to use their digital products in different devices and contexts. So the same information would need to be available on their phones and desktops, and the same tasks were expected to be accomplished on both, independent of the device limitations. A trend that over the years could easily include new mediums devices and become even more complex. For companies, the risk is that digital products that do not adapt to different usage contexts are faded to fail to deliver a good user usage experience (Park et al, 2011). Google somehow predicted this change in usage, and it did only create a cross-platform DS, but it also paved a new standard for other companies when creating cross-device experiences.

As an exciting extra move, Google made their DS available for free. Allowing other companies to enjoy all the same benefits to focus more on their business, while offering a high standard design and user experience. The downside of this approach is that it can be challenging to customise it, making companies unable to express their branding and design principles adequately when using a third-party DS.

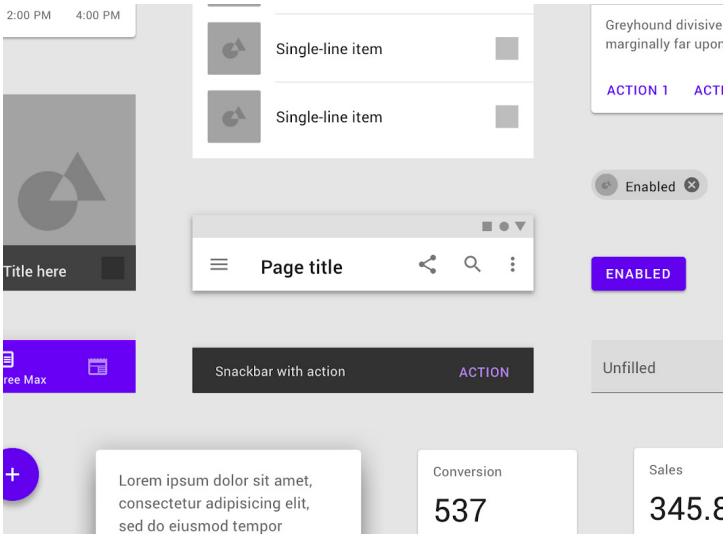

Figure 12 - Google Material Design DS. Source: https://material.io/design/introduction#principles

2.1.4 From design patterns to design systems

In modern days, tech companies and their digital products are becoming some of the biggest companies in the world (Murphy, 2021). In order to continue to scale, they need an unprecedented change in the way they build their products. However, scaling for the sake of it can generate issues in the long term. As their products get more complex, the adoption of third-party DS's such as Google Material design is not enough. So large software corporations such as Salesforce, Uber and Adobe (Fig 13) started to create their own customised solutions for an in-company DS. These brands perceived that a tailored mix of pattern libraries, style guides, principles, rules and guidelines could offer numerous benefits to make their brand stronger. While also allowing their product to scale and provide end-users with a standardised experience. Nowadays, companies of all sizes have their own DS's, and the exact moment these companies decide on this move depends on the industry. However, research suggests that even smaller companies are starting to realise the value of investing in the initiative earlier (Buley, 2019).

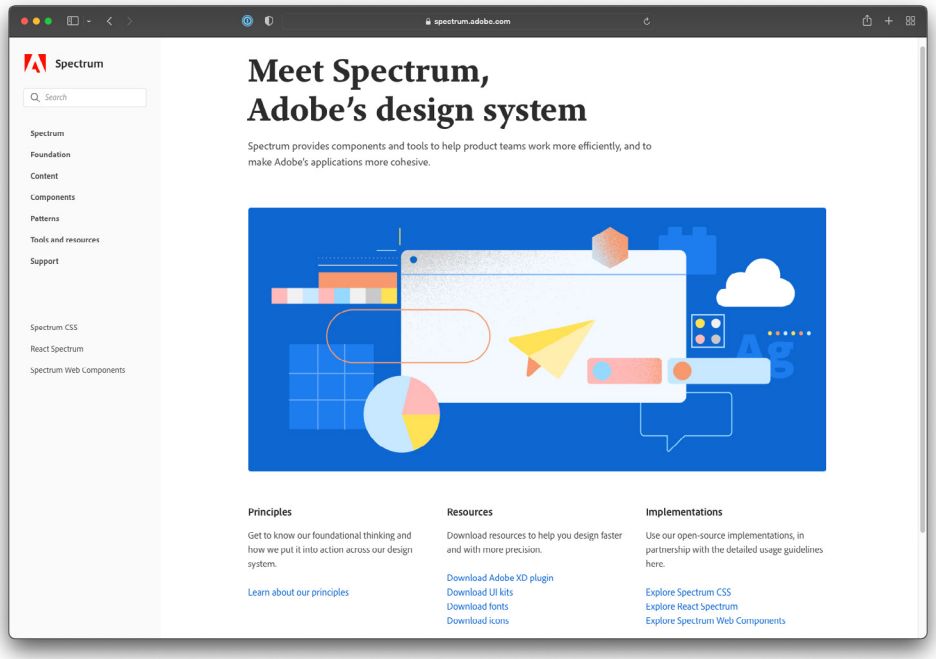

Figure 13 - Adobe's Design System (Spectrum). Source: https://spectrum.adobe.com/

The search for better design patterns and the adaption to new mediums shows how adaptable and complex a digital interface can be. Examples such as the first Human Interface Guidelines shows that the motivations for adopting a DS have not changed. From the beginning, these companies understood that the standardisation of design patterns could offer more efficiency for digital products (Vesselov and Davis, 2019). What changed over the years is the scale and how these design patterns adapt to new user needs. By encompassing all these concepts into one term, DS has become an industry standard to express a company's design maturity and commitment to the quality of its user-centric solutions.

2.2 Design system parallels with architecture

Creative areas can have many parallels, and it is not different in the case of design and architecture since these two areas make clever usage of patterns to improve their working methods and align communication with project stakeholders. Designers and architects are responsible for guiding people through space and flows by creating ambients that allow people to perform a series of desired tasks. Biswas (2019) explains that, for example, using a password to access your personal information on a website is an analogy to using keys to enter your house. Alternatively, the simple act of navigating to spaces that have a specific purpose in a home can be related to navigating between screens in a mobile application (Fig 14).

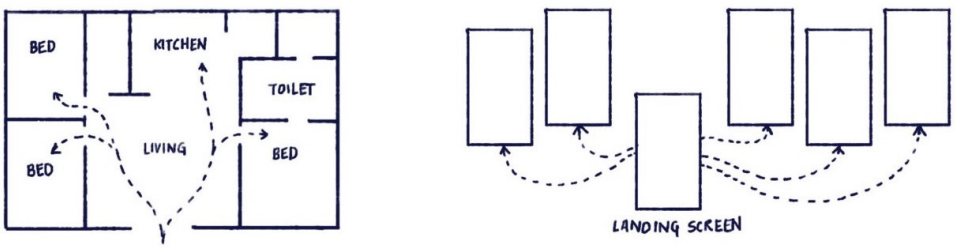

Figure 14 - Analogy of spatial navigation in the physical (left) and the digital world (right). Source: https://uxdesign.cc/architecture-and-ux-design-dacefa2a6478

The similarities are also reflected in the way patterns are conceived and represented. Alexander et al. (1977) affirm that patterns are ordered from the largest to the smallest in architecture, starting from regions and towns, down to buildings and rooms and ending with construction details. Meaning that for a more significant area to exist and be characterised as such, a combination of smaller connected patterns need to be in place to create the intended experience. To exemplify this concept, the authors use the case of creating an accessible green space. First, it should be considered the neighbourhood it will be placed in. Then, from the neighbourhood, it moves the plot and the addition of clear visual patterns such as green walls, community gardens and walkable paths, and last but not least, the types of trees and other plants. The author's example is no different from what a DS does for a digital product, indicating what needs to be used based on a set of patterns and visual details so that the user flow can proportionate the desired experience.

Both practices are responsible for creating the project for a solution, but they are often not the ones building the final product. Instead, the act of building itself is usually handled by another specialised practice. In the case of designers, this is done by developers, while for architects, this is handled by builders. Design patterns in architecture or a DS in the case of product designers come in hand to help in this usually complex workflow. When it comes to executing such projects, communication becomes essential, since it tends to involve multiple stakeholders and the need for constant and challenging decision-making. The challenges on the execution of a project from a product designer's perspective can be seen in a variant of Zendesk's Triple Diagram (Fig 15), which shows that once a solution moves to the development phase, many unknowns and variables are still present (Chen, 2020).

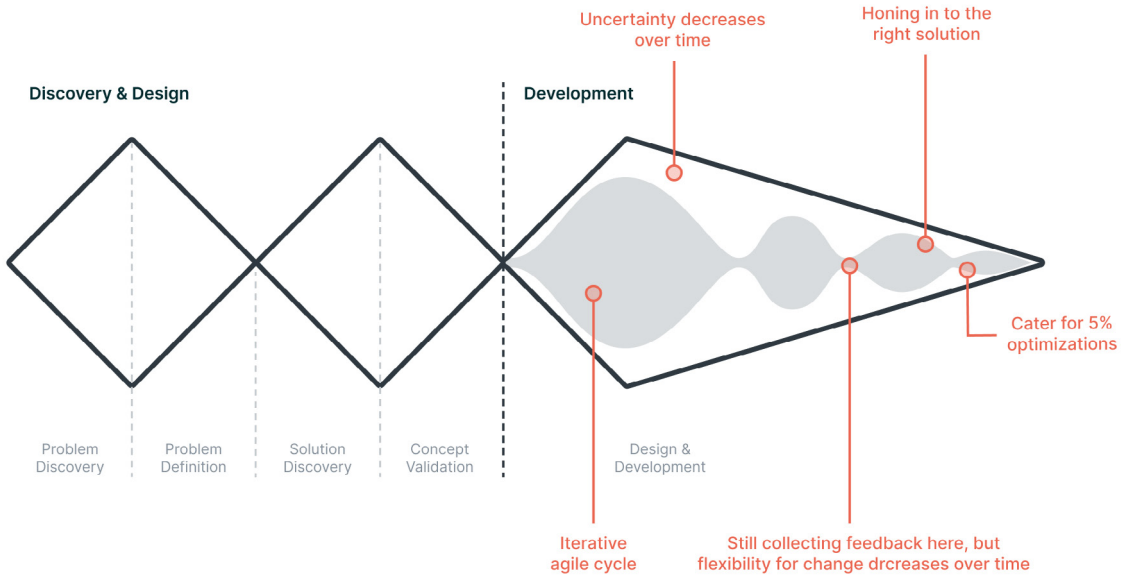

Figure 15 - Potential unforeseen events during the development phase of a digital product.
Source: https://medium.com/zendesk-creative-blog/the-zendesk-triple-diamond-process-fd857a11c179

A set of patterns or a DS can simplify these communication challenges between project and execution in two ways. First, by providing alignment of language and naming conventions for elements, streamlining communication by creating a shared language that can be carried from the beginning to the end of a project. The second way is by helping decision making when an unforeseen situation arises. In this case, documentation such as guidelines are helpful, but one method, in particular, stands out. Process diagrams and decision trees are a quick and intuitive

way of navigating the potentially complex world of patterns, providing guidance on which pattern to use depending on the situation.

Perez Cruz (2019) exemplifies this situation with an architectural example of the New York City Transit Authority Graphics Standards Manual. The author describes the usage of a decision tree to guide station planners on how to provide the sequence of information for a subway rider. This is complemented by a series of principles to improve guidance, such as "The subway rider should be given only information at the point of decision. Never before. Never after.". In a parallel with modern DS's, the same approach can be seen in an example from the American startup Lyft (Fig 16), where designers are guided on how to provide multiple choice interaction users on the company mobile application, also with the usage of decision trees (Goswami, 2021).

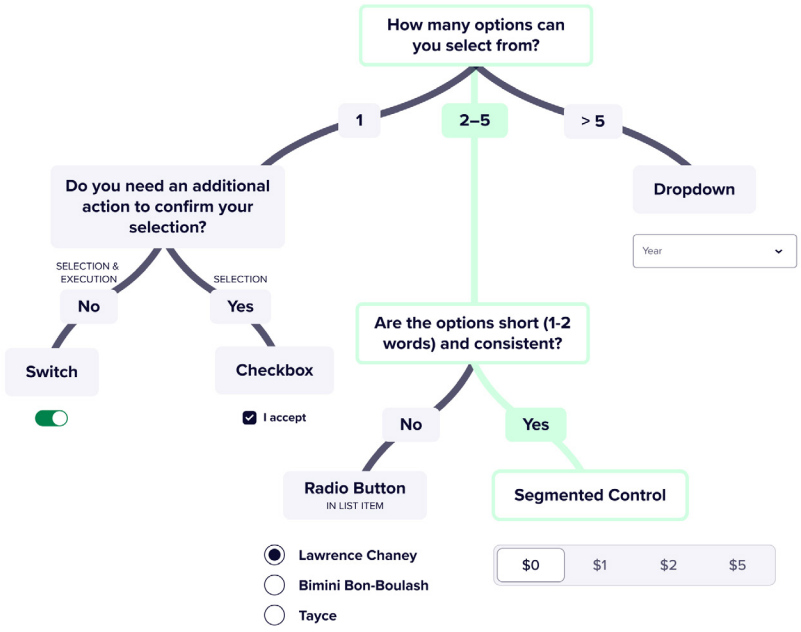

Figure 16 - A section of Lyft's selection control decision tree.
Source: https://medium.com/tap-to-dismiss/a-better-segmented-control-9e662de2ef57

The examples from the architecture fields show that some practices have been dealing with similar problems for years. The parallels from these two areas show that digital products and, in particular, product designers can always benefit from seeking knowledge in other creative fields.

2.3 The importance and benefits of adopting a design system

A DS is not only a way of aligning design and development or offering a standard set of guidelines and patterns. A DS is also a way of supporting a brand's experience consistency. In recent years, being user friendly or customer-centric has become a way for companies to differentiate from competitors (Berkun, 2020). A well-made DS allows these companies to genuinely provide a scalable and standard user-centric experience. Like any practice, design is constantly evolving, especially when it comes to digital products. This evolution has been accelerated in large software companies, especially those that commercialise their solutions as a Software as a Service (SaaS), such as Netflix, Uber, and others (Saad et al., 2021). This distributing method allows companies to constantly update their products and features via seamless updates (Benlian et al., 2011). In practice, this means that a company can publish

and validate multiple ideas quickly without the need of a customer to buy a new version of their product. So a new idea can go from ideation to delivery in hours.

One of the steps needed to support such speed is by creating a DS, since this type of visual system allows its interconnected patterns of visual elements to be quickly reused and rearranged. Without the need of building things from scratch. An analogy for this approach is like playing with LEGO kits, where a set of principles can be rearranged in different ways while still respecting the system rules. A noticeable caveat to this approach is that, different from the LEGO case, new "pieces" can be created to support new objectives (Tanrim, 2021). The patterns in a design system are divided into groups, and as in architecture, the combination of these patterns create a more specific intent. In a practical example from the author's workplace. A webpage can be built by combining tokens, components and nested components, resulting in a page aimed at the specific intent of users to browse a feed (Fig 17).

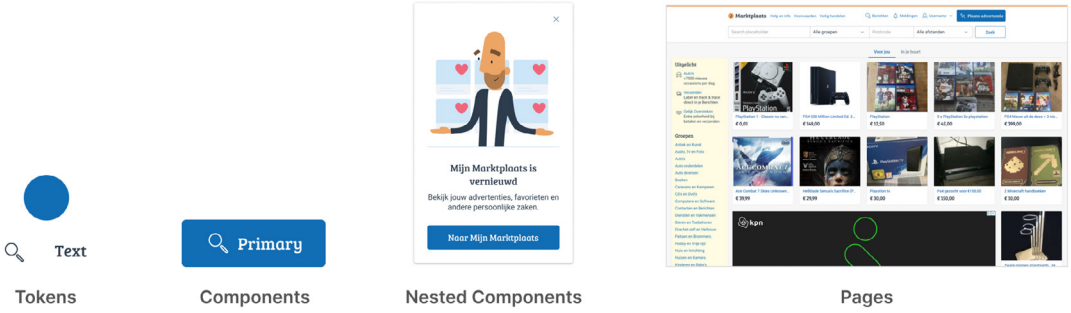

Figure 17 - HDS patterns.

The number of pattern layers depends on the complexity of a product, with most specialists agreeing that the most common ones vary between four or five layers (Frost, 2016). For example, the author's workspace DS is composed of four layers, each one with characteristics (Fig 17):

Tokens: The smallest design patterns in a DS. It is usually composed of icons, colours, typography and spacings. They are used as the base to build more complex patterns, and their style represents the visual characteristics of a brand.

Components: The combination of tokens creates patterns with distinct functions that can still be used anywhere in the product with little customisation. Examples of this type of pattern are buttons, text fields and dialogs.

Nested Components: Groups of components that can be combined to create a more complex pattern that can be used in multiple parts of the product. Such as the modal to indicate a new feature or a form with login credentials.

Pages: The combination of all previous elements results in the creation of modular pages. While having a specific intent, they also allow for easy manipulation of the design duo to the modular benefits of a DS.

2.3.1 How a DS saves time from designers and developers

A DS does not only provide "LEGO pieces" for designers. These components are usually embedded with multiple states and extra features. From usability to accessibility, all the recommended visual characteristics for a successful user experience are present (Garrett, 2011). For example, this means that when a designer uses a button in a design, they do not have to rethink how the button should look and behave, such as hover and pressed states (Fig 18). This happens because these components are centralised in the DS, so other product teams only have to use them. This also means that once this component receives changes and improvements, all the places used will be automatically updated with the new version.

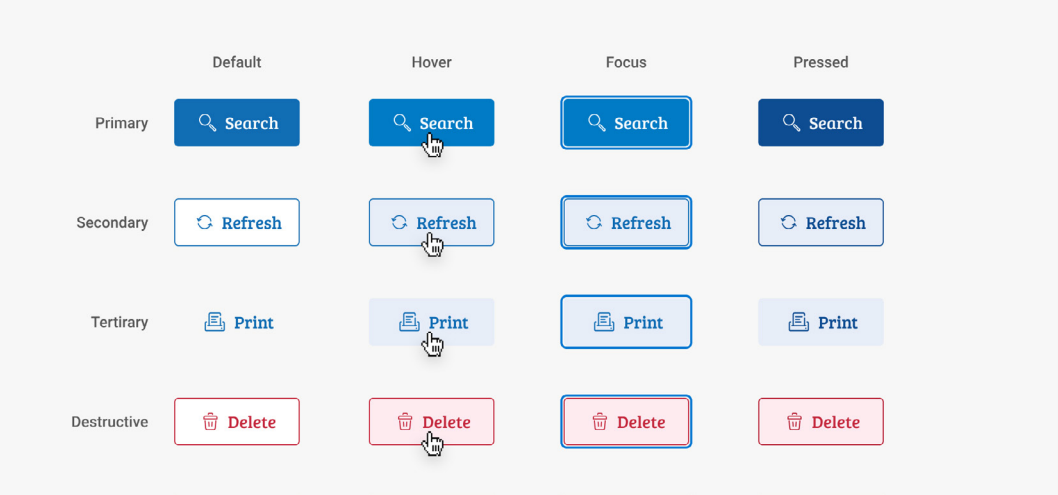

Figure 18 - States embedded into the HDS button component.

These embedded rules in DS components are especially useful in cross-platform scenarios. A Netflix user, for example, can start watching a series episode on a train and finish it on their TV at home. As seen in the literature review, this cross-platform usage has been an increasing phenomenon in recent years. These differences between devices, platforms, or brands multiply the complexity and introduce overhead in decision-making, design, development. This can grow to a point where product teams are battling the complexity of a product to get something developed, instead of dedicating time to researching and validating problems and solutions. A DS can handle this complexity on its behalf by embedding all the necessary and guidelines within its elements (Kholmatova, 2017). For example, besides providing components such as buttons, a DS can also control how this component will behave in different devices. On smartphones, it can automatically be stacked and use the entire width of the device, while on tablets, it can automatically align to the right and provide for affordance for right-hand touch (Fig 19).

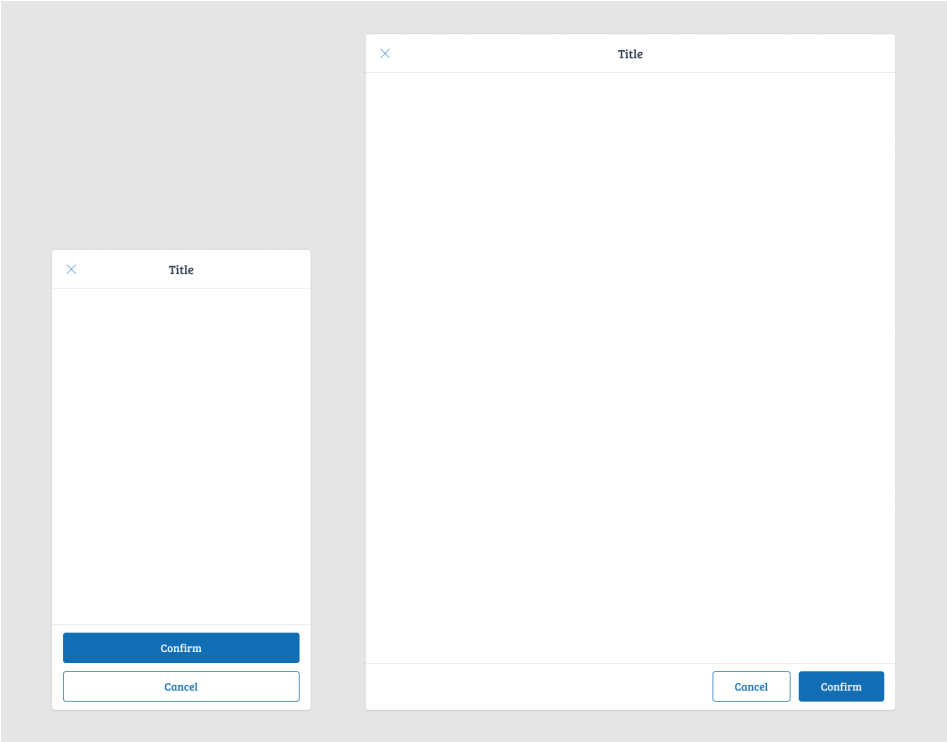

Figure 19 - HDS buttons component with different rules for smartphones (left) and tablets (right).

2.3.2 The risks of not adopting a design system

According to Perez Cruz (2019), a DS intends to, among other things, create a unified design language. Otherwise, digital products will drift towards inconsistency. This does not happen by option, and as a product scale, so does its complexity. The author uses the example of a startup and its small team. As it grows, new people join, and new teams start to be created to deal with new challenges and opportunities. These individual teams, due to the pressure for results, may develop their own styles and standards. This is the beginning of a product inconsistency and may end up as brand inconsistency. In addition, the lack of a design vision may impact the perception of users towards a brand. The author shares an example from Pinterest, a successful product for visual search references, who published an article explaining that this situation caused the product's core to get lost. They identified that the main reason people used the product was being affected by the inconsistency created by newly added features. What happened for the lack of a proper vision and thinking on how it would impact the current experience. Perez Cruz (2019) also explains that the problem is not only visual. The more significant issue is that it exposes the lack of alignment and communication between teams, making the brand weaker, and users have a disconnected experience. The author concludes by observing that a digital product has way fewer chances of falling into this issue by having a DS in place.

2.4 Learnings from system thinking

Corporations are constantly looking for strategies to make their business more successful, and for some years, System Thinking had its moment. Collopy (2019) affirms that the methodology was perceived to be a powerful guide for management in the challenge of dealing with complex phenomena more holistically. However, over the years, due to its evolving complexity, it became a niche framework. According to the author, Design Thinking was a more successful approach,

and its fame could have been enjoyed by System Thinking as well. This parallel comes from the fact that both methodologies share foundational elements, such as the collaboration between stakeholders, the ability to map problems and the possibility of working on iterations.

However, with the growth of DS adoption, there has been a resurgence of interest in the topic. While still niche, its concepts are helping shape a new way of making design, system-based design. To avoid DS having the same destiny that system thinking had in the past, its practitioners need to make the process more welcoming. As intended in this research, the simple act of allowing external contribution into a system can be a first step towards using the learnings from this powerful concept to help DS achieve new levels of maturity.

2.5 Ethical considerations regarding DS's as an accessibility enabler

According to (Kalbag, 2017) any decision made by a team affects a product's accessibility. Therefore, anything from content, interaction design, or web performance needs consideration regarding how accessibility is impacted. A DS has a crucial role in this equation. Having a DS that considers accessibility in its core comes at no extra cost for product teams to use it, since the best practices and standards can be embedded into the library of visual components.

It is estimated that around 1.8 billion people live with some kind of disability globally (Donovan 2020). Numbers that account for approximately 24% of the total world population. If a product does not consider accessibility in its way of creating solutions, it means that by default, that it will not reach the largest world minority group. To improve the situation of these people, governments worldwide have been putting more emphasis on making sure these people have access to more accessible digital products. In 2019, a new act took effect, and the European Accessibility Act (EAA) became a law (European Commission, 2019). A directive that aims to align the current differences per EU country on how a range of products and services, including e-commerce sites, comply with accessibility best practices. This act also defines that by 2025, all the EAA requirements will be mandatory, and the lack of compliance with its rules and regulations will result in financial fines for companies.

Unfortunately, the fact that a DS considers accessibility is not enough to guarantee that a product will provide an accessible user experience. The adoption of a DS plays a significant role in ethical considerations regarding customer-centricity. However, without proper guidelines for Product Teams on how to use it, this benefit will not have its intended impact. It is necessary to constantly educate Product Teams on the importance of building access solutions via the DS. A practice that has gained prominence in the author's workplace, where a whole section of the DS documentation was created to inform about common disabilities, and how the work of different areas impact these people (Fig 20).

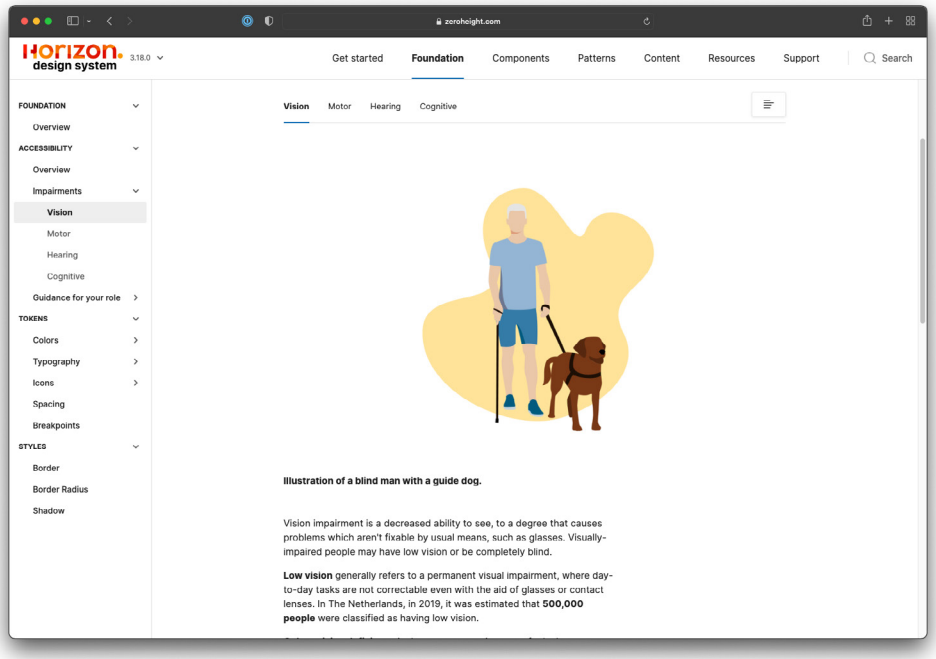

Figure 20 - HDS documentation on impairments.

2.6 Collaborative design methods for tangible contributions

According to Curtis (2015), a DS cannot thrive and scale if it depends only on its DS Team. The author affirms that reliance on making decisions, design and code needs to be shared. For starters, the author suggests that a centralized team can be an easier way to get things started. However, as the maturity of DS evolves, a federated instead of a centralized model can be a better way of scaling. The author presents three possible team models to deal with DS scaling issues, each one with its moment, pros and cons:

Solitary team model: A team that makes a DS available based on their own rules and guidelines. While this gives them complete control of the content, it does not allow other teams to provide feedback or have usage support.

Centralized team model: This model allows for one team to be dedicated to building and maintaining a DS for other teams to use. This intermediary model allows companies to solve their basic standardization needs, but it lacks the power to support active collaboration.

Federated team model (Fig 21): In this model, a selected group of people from different teams become a committee for the DS. Thus, the work is split and delegated, and decisions are made collectively and communicated to other users via communication channels and living documentation.

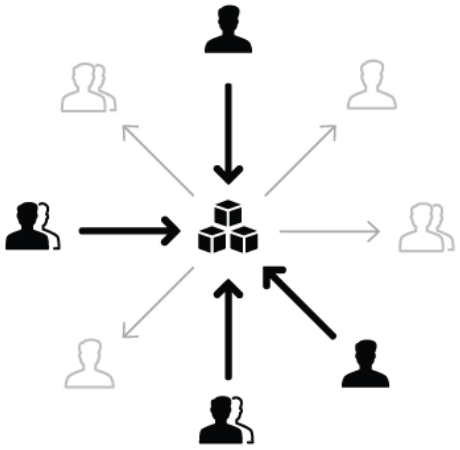

Figure 21 - Designers from different product teams contributing to a DS.
Source: https://medium.com/eightshapes-llc/team-models-for-scaling-a-design-system-2cf9d03be6a0

The author affirms that a federated approach can help broaden the relevance of cross-platform concerns, limit bias and help ease the spreading of DS cross teams by enabling more evangelists instead of a solo person. However, for a federated model to work, a solid foundation of processes needs to be established. Furthermore, for these processes to succeed, they need to be created in a collaborative way. By partnering with the future contributors, so their concerns and challenges are taken into account. Otherwise, the adoption will not be as expected, which may lead the initiative to fail.

Foundation, communication and alignment are common sense for successful teams. However, a federated team can quickly become ineffective due to its members also having responsibilities with other teams in the company. According to Lencioni (2002), if a team wants to succeed independent of the present challenges, it needs to embrace trust as a base to become successful. If trust is present, people have more constructive conflicts, leading to high commitment, accountability, and ultimately achieving shared goals. Otherwise, without trust, teams may lose the sense of a shared objective, a sentiment that gets replaced by stagnation, personal goals and in worst cases, people may start leaving the team.

2.6.1 Effective feedback and clear communication

DS's depend on feedback to evolve, especially when issues or opportunities need to be quickly incorporated. Luckily, insights on how to properly make it can be extracted from business books. Some recommended principles include: Feedback needs to be shared constantly and in context, it needs to intend to achieve a specific outcome, it needs to be a two-way conversation, and most importantly, it needs to show respect by the recipient (Harvard Business Review, 2014). Otherwise, a DS Team can easily spend a fair amount of time dealing with non-useful or out of context feedback.

Communication is vital to help team members not only align by also trusting each other. When using a DS, a significant portion of the process relies on people relying on documentation. So guidelines, rules such as dos and don'ts need to be explained in an obvious way. Large software companies tend to have multicultural environments. So as recommended by Meyer (2014), when it comes to communication within a company, especially in the case of multicultural teams, a low-context style should be adopted. A mostly western style used by Anglo-Saxon

cultures to communicate. It is characterized by its clear and direct, often followed by a repetition of what was said to make sure the recipient received the message correctly. An example of this approach in the business environment is when a meeting is finished. The participants receive a list of the decisions, followed by appointed people responsible for each remaining task. The United States is the lowest-context culture in the world, as presented in the diagram of dominant communication styles per country (Fig 22).

Figure 22 - Scale of low-context vs high-context countries. Source: Meyer (2014).

So any artefact created for a DS that requires communicating with its internal users, should be done in a low context way so that there is no room for misinterpretations. Otherwise, the wrong interpretation of a guideline could result in a solution built in a way that does not fit the intended objectives of the brand and product.

2.7 The future of design systems

The future seems to hold numerous opportunities for how DS's can evolve. The most prominent is the fact that the guidelines and rules of a DS can become the base for a revolution in design automation. Currently, designers and developers have to consume this content in a passive way, usually by reading some type of documentation. However, this same set of rules and guidelines can become the base for AI-powered tools that can proactively design solutions based on a defined problem. An existing and powerful example comes from using an AI generative language model, known as GPT-3. In an experiment with the technology,

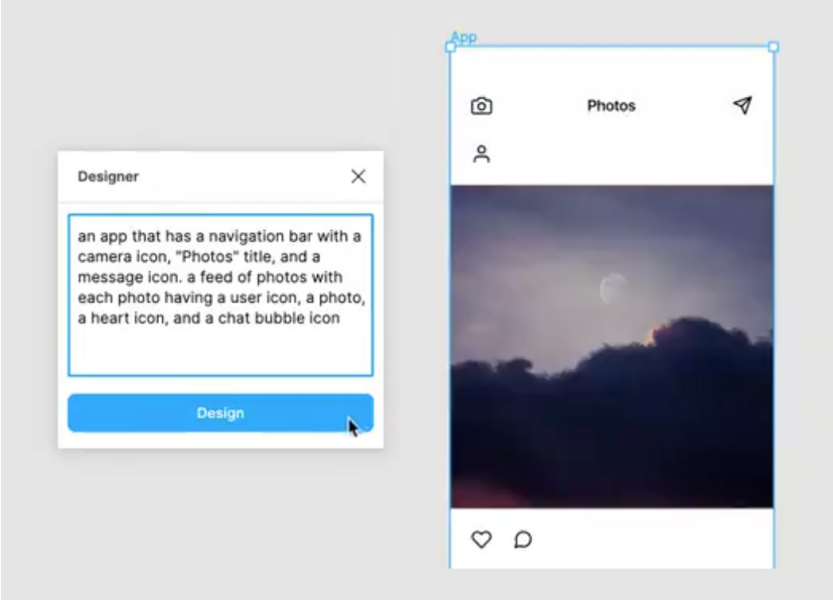

Figure 23 - AI design plugin.
Source: https://uxdesign.cc/lets-talk-about-that-gpt-3-ai-tweet-that-shook-designers-to-the-core-d2b31ad3d63b

a developer built a plugin for a design tool called Figma (Fig 23), that can comprehend the content from a text and generate an interface based on its interpretation of what was described in the text input. In other words, it can, to a certain point, replace the work that a designer currently does manually. However, according to Lee (2020), this should not be seen as a genuine replacement or something unfavourable for designers. On the contrary, this can become an opportunity for designers to dedicate more time to validating their explorations, leaving the heavy work on generating solutions to the AI.

These new technologies can also be used to bring ideas to life in a faster way, and Airbnb has been exploring this kind of solution for years (Wilkins, 2017). They built a team responsible for internal tools, including their DS. This team was able to create solutions that, for example, can transform the sketches on a piece of paper into a working prototype (Fig 24), by identifying the design patterns drawn on the paper and finding their equivalents into the company DS. According to Wilkins (2017), the primary motivation for building this tool is to validate ideas as quickly as possible, helping Product Teams validate hypotheses way faster than using traditional methods.

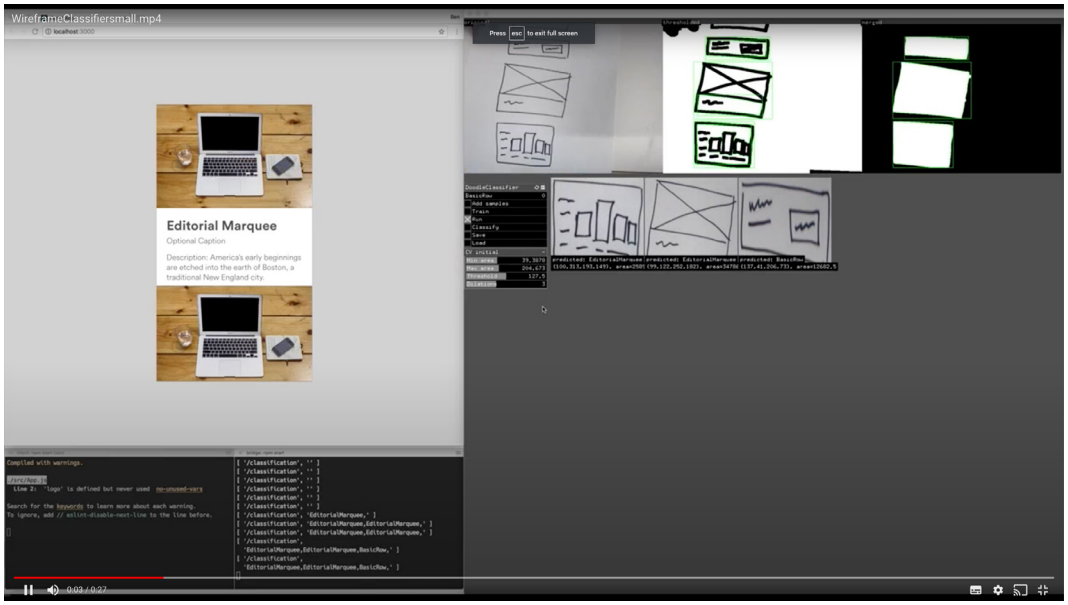

Figure 24 - A tool to generate prototypes based on paper sketches.
Source: https://airbnb.design/sketching-interfaces/

While most of these tools are explorations, some have already made their way into the workflow of designers. In another example from Lyft, the company created a set of tools that help their design team to automate the design check (Fig 25), a stage before the work gets transferred for developers to build. This tool allows designers to validate if they are using the correct styles and design patterns according to the company DS. In the case of nonconformity, the tool promptly suggests alternatives on what to do. According to Gold (2020), this is an exciting moment for design tools. They are becoming more connected and extendable. In addition, these tools are moving from the need for very manual input to an intelligent and automated way.

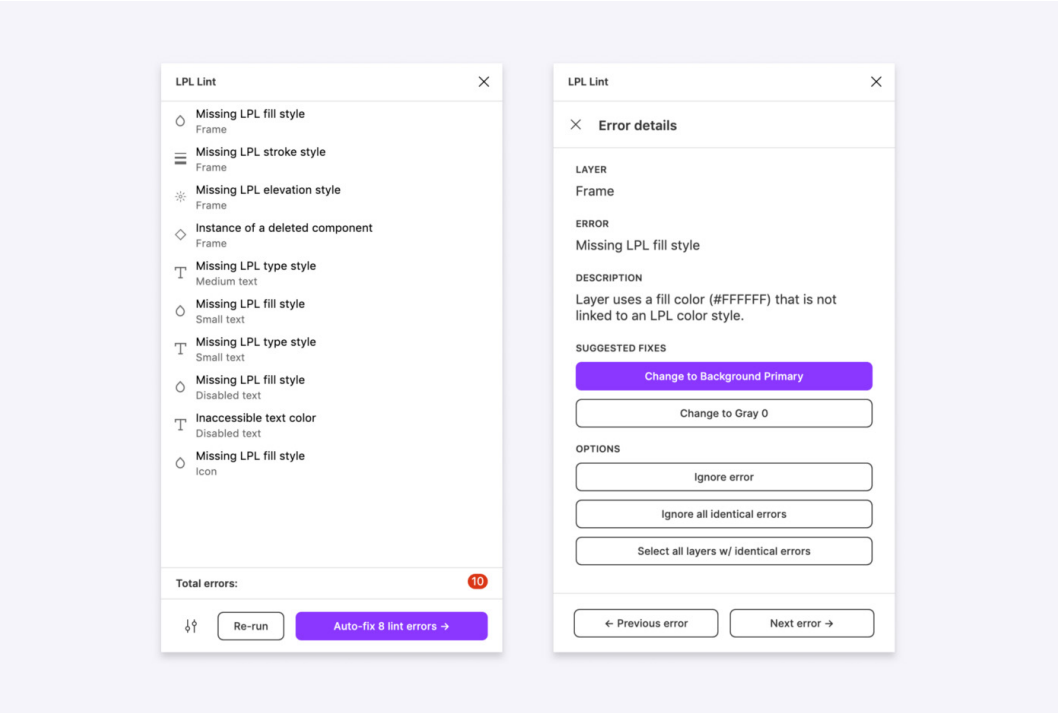

Figure 25 - Lift customised version of Design Lint.
Source: https://medium.com/tap-to-dismiss/color-within-constraints-d6f777a3b72d

2.8 Insights

The literature review gave an essential and insightful overview of the topic of DS. Moreover, some of the findings were crucial to shaping the next section of this research, in particular:

- Design patterns have been adapting to the evolution of technologies and device types. However, this flexibility has not impacted its importance, and they continue to be essential to project consistent and intuitive solutions. However, a DS is not only about the interface and design patterns. It is also a scalable way of portraying a brand.
- The parallels with other creative fields demonstrated that these areas have numerous synergies, such as the processes and methods adapted into product design from architecture. Therefore, the act of opening one's horizons to look for similarities in other fields can positively impact an emerging field.
- The influence of a DS goes beyond the design practice. Companies are constantly looking for ways to work not only faster but better. DS's can support them on this objective while guaranteeing usability, accessibility and brand standards with minimum effort.
- There are ethical considerations on the usage of a DS. By creating one, companies can double down their impact on accessibility. In addition, the approach allows for easier and faster implementation of required regulations on the topic.
- As the DS maturity increases, the way companies need to handle it also needs to adapt. From a solo contributor to a federated team, there are multiple stages and approaches to be taken. However, their efficiency will depend on the environment, so they need to be validated in the context of each company.

- The customization and flexibilization of a DS are crucial to scale, but this can only be achieved with proper process and agreements between its users. Also, a dedicated team or a group of responsible individuals must work together to guarantee that these processes have the support needed.
- When it comes to tangible collaboration with a DS, there is a need to create artefacts that support it, such as decision trees and diagrams, that help indicate what to do from a user and contributor perspective.

In the next section, the previously presented problem in the workspace will be explored. Finally, the learnings from this literature review will be combined with an innovative methodology to find a solution that can be implemented in the author's workplace.

3. Research Process and Methodology

The identified problem in the workspace marks the beginning of a different stream of work for the DS team. The focus on mapping processes and creating diagrams is more strategic than the previous challenges faced by the team. Therefore, the methodology chosen for this research is the Design Thinking framework, a user-centric and hands-on approach to innovation composed of six distinct phases (Fig 26). There are many diagrams to represent this framework. However, they all focus on the same objective of understanding the roots of a problem, exploring potential paths and materialising a solution that can be implemented to solve the identified opportunity of improvement (Gibbons, 2016). The decision for this framework comes from two main reasons. First, the topic of tangible contribution within a DS is new to the company, so a methodology such as Design Thinking can help to empathise with the current users and the potential problems they are facing. Second, since the outcome of this research will not require software development, a methodology with less focus on fast implementation is recommended. Also, there are many unknowns regarding how the possible solution will unfold, so a process that emphasises the iteration and prototyping phases seems to be a better choice.

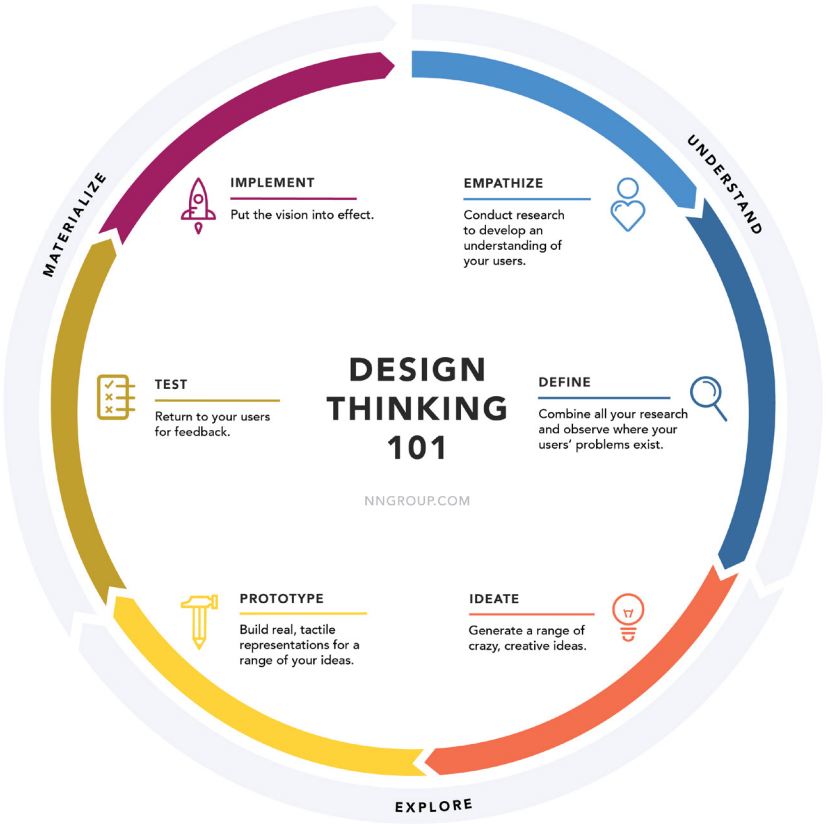

Figure 26 - The Design Thinking framework. Source: https://www.nngroup.com/articles/design-thinking/

Before starting the usage of the framework, a plan was traced considering the time and resources constraints of the research. The objective was to make sure that all phases would be executed, and that all the necessary preparations would be made upfront. This plan also considered the learnings from the literature review, which were used as guidance in pursuing the best possible solution for the problem. Some steps were left with more flexibility, so more time could be dedicated without compromising the agreed dates for finishing the research. This

was especially important for the validation phase, where findings could make the explorations for the solution start from scratch if not successful. The Design Thinking framework variant chosen for the project is from Gibbons (2016). It is used by the Nielsen Norman Group, a worldwide renowned consulting firm that is specialised in design and user experience. This is a brief explanation of the plan for the six phases that compose the framework:

Empathise
For the first phase, an in-company research with designers from different product teams was planned. The method chosen was a series of semi-structured interviews to uncover the potential pain points in the current process of contributing to the HDS.

Definition
This phase focuses on refining the project boundaries and scope based on the first phase's learnings and insights. A company presentation framework will be used for this. The objective is to create a clear and shareable definition of the problem being solved. So stakeholders can easily understand the rationale behind the project and contribute with comments and feedback.

Ideation
In this phase, a series of explorations for possible solutions will be designed. For being a process that can be composed of multiple steps, the method chosen is the usage of sketches, going from low to high fidelity ones, as the confidence in the solution grows. The objective is to identify the iteration(s) with the most potential so it can be used as a base for a prototype.

Prototyping
For this project, two prototypes will be required. The first one is the process diagram itself, which needs to be prototyped in a way that supports easy iterations. The second is a simulation of the integration of these processes within the HDS documentation. The objective of these prototypes is to support the validation phase.

Validation
Three rounds of validation were planned. With the opportunity to iterate the prototypes between each session. So the following sessions can benefit from the insights and learnings obtained from previous ones:

- **First validation:** A co-validation session with the design team. The objective is to validate and discuss the steps of the process diagram.
- **Second validation:** Also hosted as a co-validation session, it will be conducted as a demo of how the process diagram will be integrated into the HDS documentation, followed by the opportunity of designers to leave annotations on possible improvements. The objective is to evaluate if the placement and content in the documentation are precise and intuitive.
- **Third validation:** For the last round, three individual interviews will be conducted. The objective is to simulate the usage of the process within a hypothetical situation. Designers will be provided with a task briefing and need to use the documentation to describe what steps they would take to complete the task. The objective is to identify how confident designers are in using the solution. So the decision if it can be implemented or need more iterations can happen.

Implementation
If the solution is proven to be effective after the validation phase, the potential pending insights will be applied, and the prototype will be converted into the final implementation. Once this new section of the HDS becomes available containing the new proposed contribution diagram process with its guidelines, a company-wide announcement will be made.

Once the solution is implemented, the focus will shift to the identification of potential next steps. To help with this phase, two industry specialists will be interviewed to evaluate the proposal and share their experience regarding DS, DS teams and contribution models. These are essential pieces of information to create a stronger case for the project, since the last step will be a presentation of the results of this work to the company leadership team.

3.1 Empathise

According to the Design Thinking framework, before jumping into ideas or sketches of possible solutions, it is essential to observe and collect information about the problem being solved. The objective is to avoid bias towards solutions and prevent vital information from getting uncovered too late in the process. Thus, making its usage unfeasible or too costly to be integrated.

Based on the nature of the problem and scope of the project, remote interviews were the method chosen for this phase. The usage of a semi-structured interview script allows for flexibility on what topics are worth pursuing during the conversation. Furthermore, the usage of open questions allows the interviewees to explain their thoughts more freely. As a preparation for the interviews, a script was created, with topics that could help bring clarity around the problem, such as:

- The comprehension and experiences of using the current contribution methods to the HDS.
- Clarity on channels available to get in contact with the DS team regarding collaboration.
- The knowledge about the HDS and the way the DS team works.
- Potential blockers to their current and future contributions to the HDS.

Summary of the research
Stimuli: Semi-structured interview script.
Method: Remote interview (30 min).
Participants: Four participants, split between 3 product designers and 1 lead product designer.

Research Insights
The interview with designers (Fig 27) revealed precious insights into their perception of tangible collaboration to the company's DS. The objective of these interviews was not only to identify issues. It was also essential to understand what is currently working well. So any change in the process can be done with these points in mind. The main insights and learnings from the interview were:

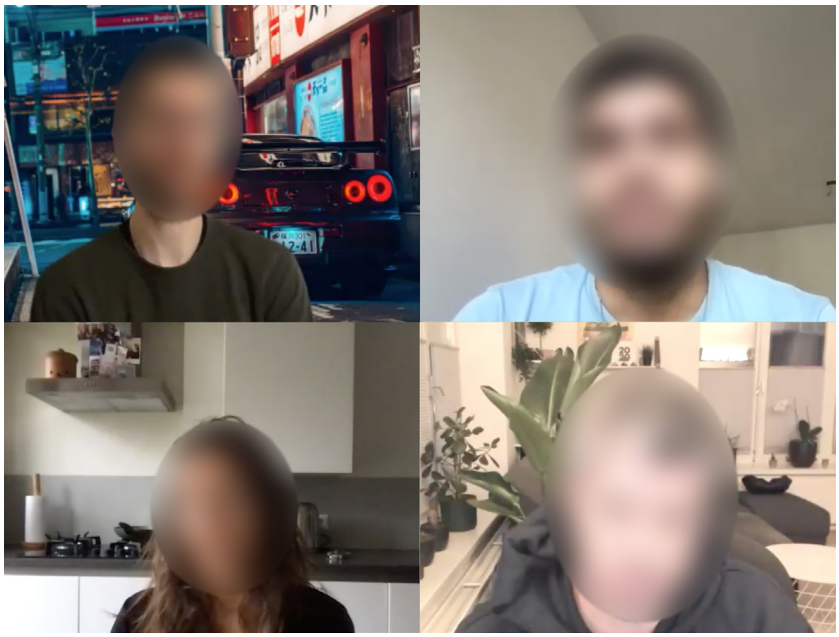

Figure 27: Interview with designers from product teams.

What is working well:

- The DS team offers a centralised communication channel, making it clear and easy to know where to go if any kind of support is needed.
- Before asking for help, designers will often check the HDS documentation to avoid raising topics that may have already been addressed. This is supported by the current weekly release of changes to the documentation.
- The work of the DS team is appreciated, and there is complete trust that they can handle decisions towards the future of the DS, even if it implies changing the way the team handles requests from other teams.

Potential improvement points

- Due to the current centralised way the DS team works, designers from product teams tend to rely on the team for almost all the work. Often holding back or changing needs that they consider the DS team will not be able to handle considering their workload.
- Some designers consider the skills required to contribute to a DS not part of their skillset. However, some admit that this may come from the fact that they rarely do it, and with proper guidance, they could be able to gain confidence regarding their contribution.
- Designers say they lack resources to understand the process of building solutions for the company's DS. The processes applied by the DS team to create or change components and tokens are not documented anywhere.
- There is no explicit agreement and process on how contributions towards the HDS works. What makes designers not feel responsible or accountable for any type of contribution.
- There is a clear understanding that a contribution towards a DS needs to be tangible, meaning that a fundamental change in design and code needs to be present. However, this seems complex, time-consuming.

- The only collaboration aimed page of the documentation is focused on developers. There is no explanation of how designers can collaborate. When something emerges, they tend to reach the team via direct messaging.

The learnings obtained clarified some of the potential causes and barriers for designers to contribute to the HDS. They also provided important insights on how they could be solved, such as improving the guidance per role and expanding the collaboration and contribution sections in the documentation. However, designers' confidence in contributing to the HDS is low, so this needs special attention.

3.2 Definition

To start a project at the author's workplace, the person or team responsible should prepare a presentation with a clear and shareable definition of the problem intended to be solved. This content should be summarised and easy to digest while covering all the basics related to the intended audience, scope and deliverables. This is then made available in a shared repository for everyone in the company and used in presentations for different stakeholders to explain its value quickly. This research had to follow its requirement, and an existing template composed of five questions was used for this phase. An exercise that perfectly aligns with what the Design Thinking framework requires for its definition phase. Also, the fact that the project started earlier than this definition is no issue. Otherwise, there would be no assessment if the research question was valid. So the early interviews were used as a base for the definition and scope of the project:

What are we trying to improve?
The DS team wants to improve and clarify how designers from product teams can create tangible contributions to the company DS. This is intended to be achieved by providing a new documentation section composed of diagrams and guidelines.

Why is this important?
As learned from the literature review and cases from other companies, the only way of scaling a DS is by focusing on decentralising some of its responsibilities. Something that can be achieved by allowing external, tangible contributions from other teams. This could also bring autonomy and empowerment to designers from product teams, since they would rely less on the DS team to get the necessary work done for their projects.

How will it help the company?
It will accelerate the creation of a unified design and brand experience, avoiding the product from becoming inconsistent as it grows and becomes more complex. Also, by scaling the DS and its capabilities, its benefits such as speed of implementation, consistency, usability and accessibility will become more available to all product teams in the company.

How will we know if we are successful?
The primary proxy to measure the success of the process is the number of tangible contributions made by designers from product teams. Because this process will result in the implementation

of a new token or component, therefore being countable. Also, even the number of potential issues identified by its users can be used as an indirect proxy, which means that the process is being used but may need to cater to new scenarios.

Who is this for?

The primary audience of this research and its outcome are designers from the product teams working in the Benelux region. Therefore, its implementation will comprehend the region's two products, Marktplaats and 2dehands. However, its learnings and results can be used to create similar processes for any other product in the company. Since they share all the characteristics from these local teams.

With this phase complete, the project was shared with multiple stakeholders in the company. The objective was to identify potential partners and recruit people who could help with the project validation phase.

3.3 Ideation

From the early interviews, it was clear that a process diagram just for the contribution process would not be enough. The lack of clarity regarding the processes used by the DS team could undermine the intentions of this research. So besides the process diagram for a tangible contribution process, two others were added into the scope of explorations. These two other processes would cover the creation or changes to the DS components and tokens. The ideation phase started with paper explorations. These early sketches' intentions were not to create something presentable or focus on the details. Instead, they were intended to quickly conceptualise the notes from the research into a visual concept (Fig 28). They also served as a first exercise on identifying the main steps of the process and the copy that could be used to explain them.

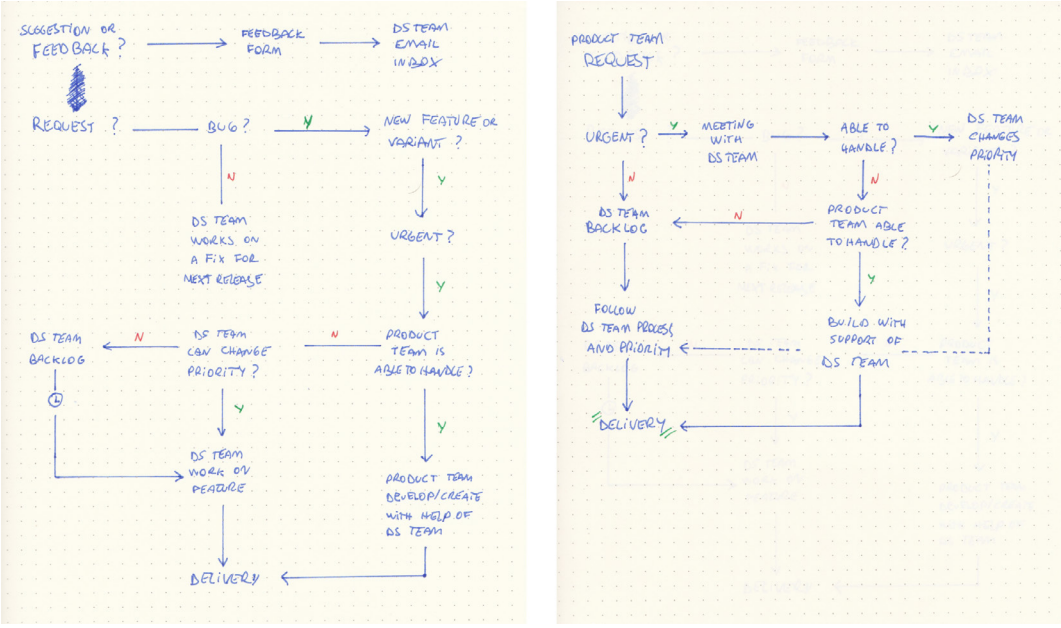

Figure 28 - Early sketches on paper to maximise productivity.

The three process diagrams were explored in parallel. Once the potential main steps for each one were identified and a few different flows sketched, the explorations moved to a digital process. For this stage, Miro, a collaborative artboard tool, was used. This tool was chosen because it has embedded features to design flowcharts and supports remote collaboration, which would be helpful for the validation phase. The same exploration process for possible iterations was repeated for this stage, but with more details and applying industry-standard diagram shapes for each process step (Fig 29).

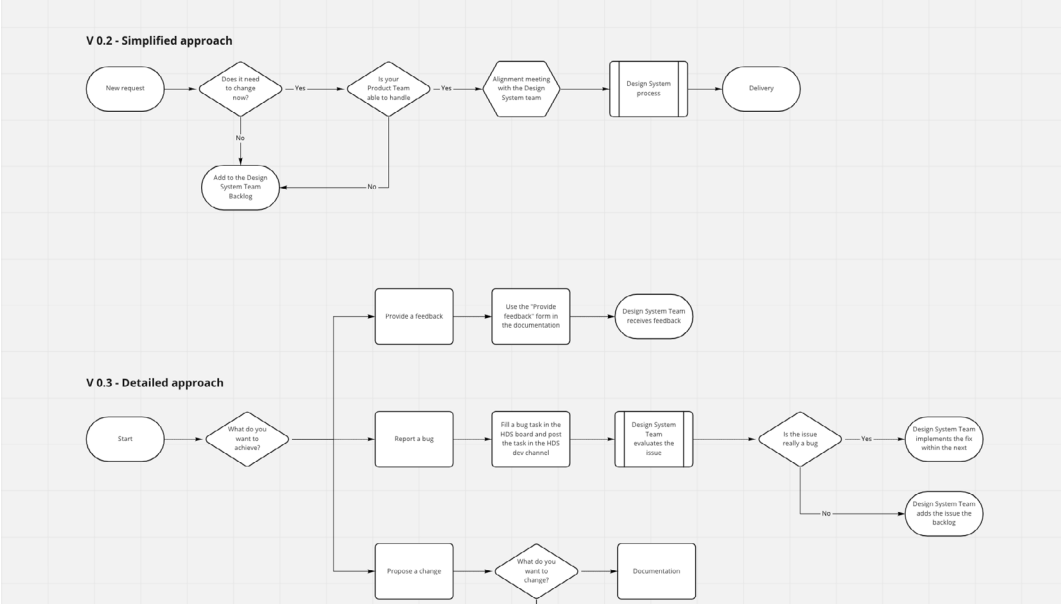

Figure 29 - Iteration exploration of the collaboration process using a digital tool.

Once explorations were in a more advanced stage, the DS team was invited for a one-hour co-creation session (Fig 30). The virtual workshop had two main objectives. The first was to align the team by validating if the diagrams clearly represented the processes and responsibilities. The second was to identify the version of each diagram that had more potential, so that it could be prototyped and used in the validation phase. This session proved valuable since, thanks to the input of the DS team, important points were raised, discussed and fixed as early as possible in the process.

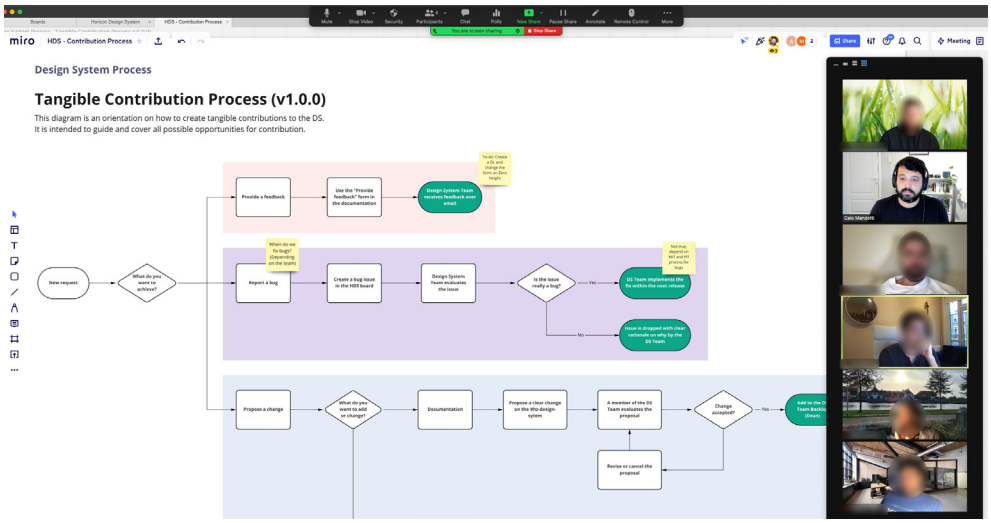

Figure 30 - Co-creation with the DS team.

As a final activity during the session, the DS team discussed the intentions and methods to be used in the prototype phase. This debate resulted in the decision that, indeed, three processes had to be prototyped and validated. Another decision was that the validation of the content of these diagrams and the location they would be placed in the documentation were two different things. Therefore, two prototypes and at least two validation sessions would need to happen.

3.4 Prototyping

The decisions and possible paths discussed during the ideation were used as a base for the prototyping phase. Prototypes offer a tactile representation of the solution without the need of having it implemented (Gibbons, 2016). Even with limited coverage of use cases, they can provide valuable insights by observing users interacting with them. These learnings can help anticipate issues and identify points that can be improved. As decided in the validation with the DS team, two prototypes were built for validating the proposed solutions:

3.4.1 Prototype A

This prototype intends to validate the content, steps and general understanding of the processes. Therefore, the three process diagrams were added to a virtual artboard as a preparation for the validation phase (Fig 31).

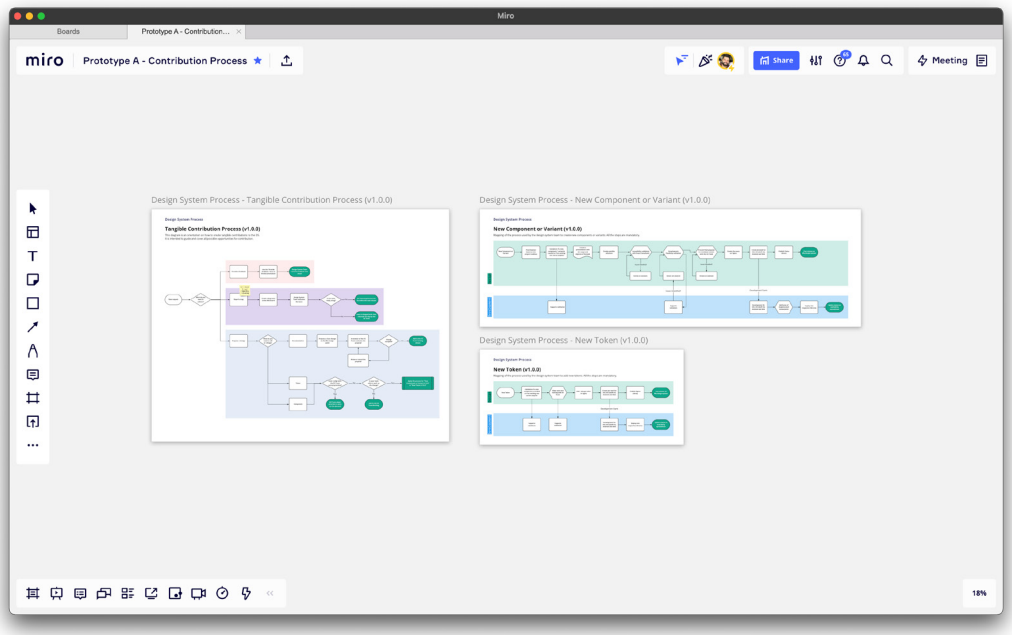

Figure 31 - Prototype A: Process prepared for feedback in a digital artboard tool.

The decision was to use the same tool where the iterations took place. By doing this, the final solution could be exported as an image asset and added to the documentation once finished. This approach can also benefit future iterations by keeping a source that can be quickly iterated with new findings. This prototype artboard contained the following diagrams:

- **Tangible contribution process:** A new process to support designers from product teams to identify how they can tangibly contribute to the company DS.
- **New Component or Variant:** A DS team process to create or change components and variants.
- **New Token:** A DS team process to create or change tokens.

The work on this prototype also revealed that the three diagrams could be connected, reducing the number of similar steps. For example, depending on the terminator that users would land on the tangible contribution one, they would be directed to one of the other two to continue their process.

3.4.2 Prototype B

Currently, all the HDS documentation is hosted in a third party platform called Zero Height. This product allows DS knowledge sharing via the creation of documentation using different types of media. So to avoid fragmented access to DS related content, the same tool was used to create the second prototype (Fig 32). This prototype objective is to validate the solutions intended placement and content within the HDS documentation. The usage of this tool allowed the creation of a high fidelity prototype that can hopefully provide valuable insights and learnings. Also, to avoid misunderstandings, the documentation area with the prototype was hidden under a password. So no one that was not part of the validation phase would have access to its content.

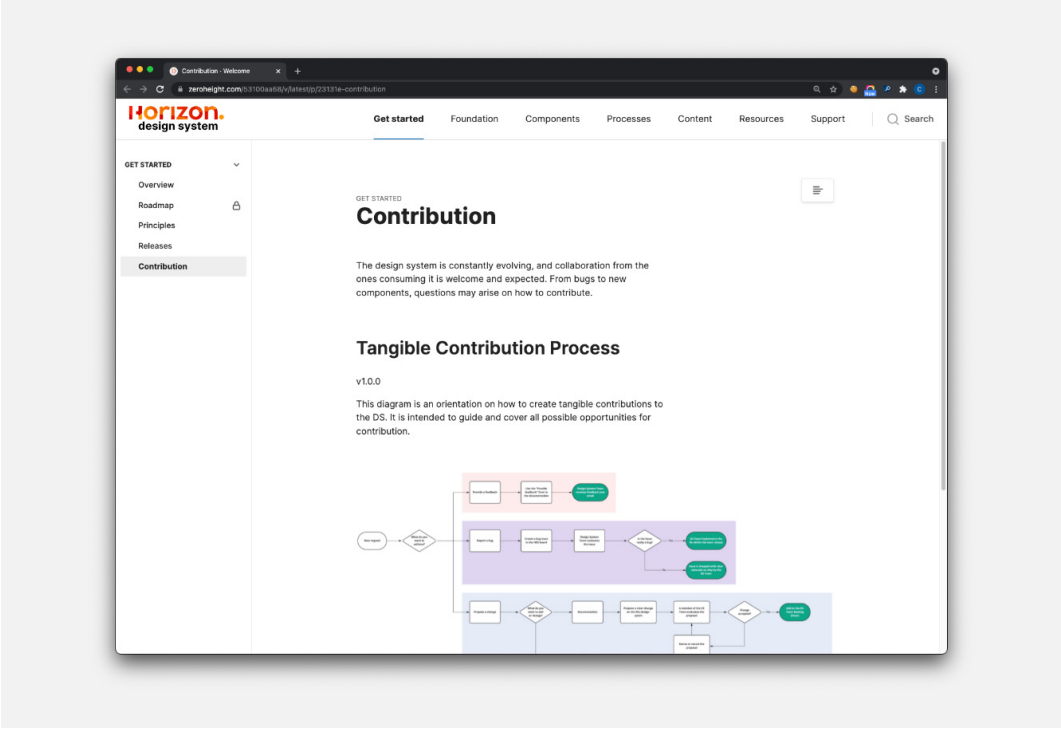

Figure 32 - Prototype B: Diagram processes in the HDS documentation.

Luckily no new tools or methods had to be adopted for prototyping the solutions since they were already available in the company's toolset. This not only accelerated its conception but also allowed these tools to be used at their maximum capabilities, based on the knowledge acquired from the previous usages. Finally, the two prototypes were reviewed again in a session by the DS team and judged to be ready for the validation phase.

3.5 Validation

The validation phase took place over the spam of 10 days. They were scheduled in a way where the learnings obtained in a session could be applied before the following session, allowing for a new iteration to be created considering all insights and learnings obtained. This was an important feature, considering that the validation phase was composed of three validation sessions.

3.5.1 First validation

Stimuli: Prototype A.
Method: Remote virtual workshop with facilitation (30 min).
Participants: Eight participants, split between 4 product designers, 2 researchers, 1 lead product designer and 1 head of design.

Participants were invited to evaluate each of the processes one by one. The session started with an individual evaluation for each process, followed by the recommendation to add post-its to the steps that were unclear or needed discussion (Fig 33). Once completed, all these points were discussed as a team, and action points were created for each one. At the end of the session, the team was asked if this solution could help them be less dependent on the DS team, and all agreed that it heavily contributes to this objective.

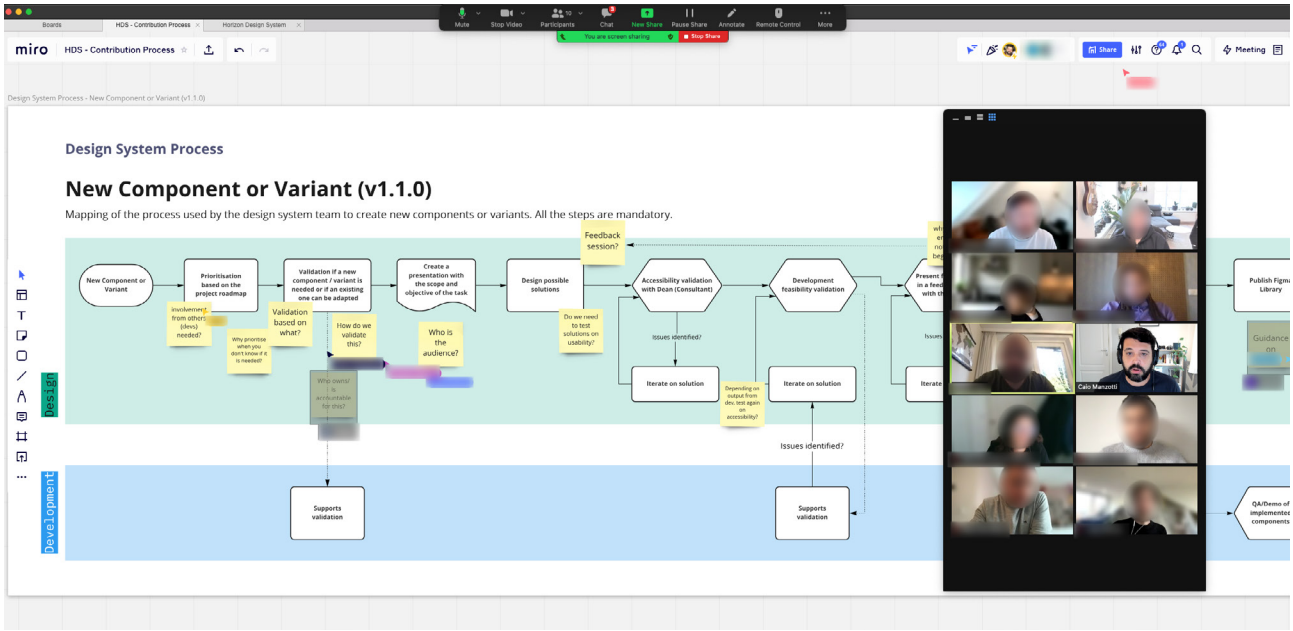

Figure 33 - Remote virtual session for the first validation.

The session was very successful, generating multiple insights and more than 20 action points. The most critical identified issues were related to the fact that these processes were made using the internal naming conventions that only the DS team could relate. The main action points and learnings were:

- Improve clarity about the audience of the process by adding the practice to the name of the process, such as "Tangible cOntribution process for designers".
- There is a need to add a new lane that shows the stages where designers share the

responsibility with other practices, such as development.

- The feedback flow needs to be removed since it does not allow two-way collaboration. This goes against the intention of only offering actionable and tangible contributions.
- The section related to bugs needs improvement. The participants pointed out that the primary source of bugs was not being mentioned in the process diagram.

At the end of the session, the participants were asked if this solution could help them be less dependent on the DS team, and all agreed that it heavily contributes to this objective. Also, by fixing the identified issues, the solution would be ready for the second validation. Participants were then invited to continue their collaboration after the session. So a channel for feedback was created that allowed them to re-evaluate the iterations even if they were not invited for the subsequent validation sessions.

3.5.2 Second validation

Stimuli: Prototype B.
Method: Remote virtual workshop with facilitation (20 min).
Participants: Five participants, split between 4 product designers and 1 lead product designer.

The session started with a demo using the interactive prototype to show where the process diagrams would be placed within the documentation. Participants were then able to check the content by themselves. This was followed by the possibility of adding comments regarding the possible improvements (Fig 34).

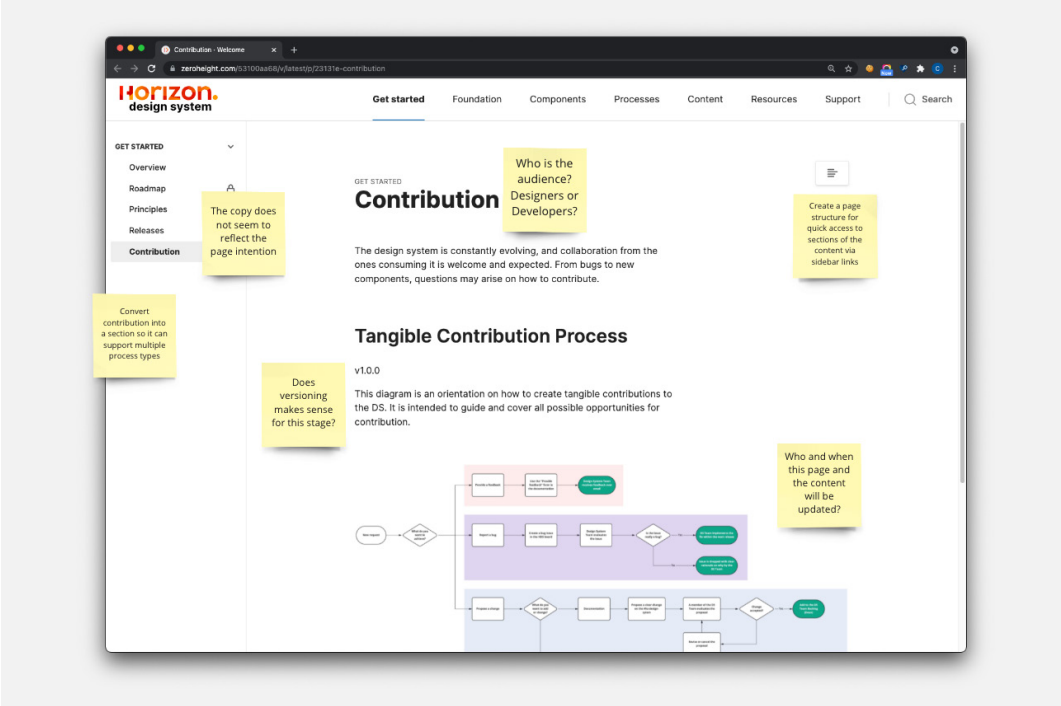

Figure 34 - Feedback being added during the second validation session.

These comments also led to a facilitated discussion and the compilation of 12 action points for improvement. The main action points from the second section were the following:

- The placement works well considering the navigation structure of the documentation, but the page name is not reflecting well the content of its pages.
- Instead of a page, the content should be treated as a section, where the future processes for other practices besides design can be added as well.
- Corner cases can be explained with the text guidelines. There is no need to add all cases into the diagrams. Otherwise, it can get too complex and extensive.
- Participants decided that the prototype would be ready for the last validation session if all the action points were addressed.

3.5.3 Third validation

Stimuli: Prototype B.
Method: Remote/In-person interview and usability test (30 min).
Participants: Two participants, both product designers.
Scenario: A hypothetical situation where the participant had to perform a specific change in a DS colour token.

The third and last round was planned as the in-depth validation of the process diagrams. The objective was to simulate a hypothetical situation to evaluate how the participants would navigate the possible flows. The task was designed to make usage of two of three diagram flows, therefore also validating if the connection between more than one diagram was clear (Fig 35).

Figure 35 - Interviewee suggesting changes during the third validation.

The two interviews were successful in revealing insights not captured before. The main ones are:

- The confidence regarding the process in the hypothetical situation was moderate to high. To elevate this confidence, more detailed information on how to perform each task should be added to the text content on the page. Also, tutorials in video could help designers be more

confident for critical parts of the flow, such as when publishing a new version of a DS library.

- The autonomy while using the process was perceived as high, but some of the copy needs to be improved, such as mentioning a direct DS team contact for each lane of the flow.
- It is sometimes challenging for designers to evaluate if they should wait for the DS team or work on the changes themselves. An indication of the average time for each step could be provided, facilitating the decision making.

No blockers were identified in this last validation. The identified issues became action points to be fixed before the implementation, but some need further investigation. Therefore, they were flagged as potential changes for future iterations.

3.6 Implementation

The last phase of the Design Thinking framework intends to materialise the solution, which is achieved by implementing the research results. The learnings and insights collected during the previous phases indicate that the proposed solution works as intended and can be implemented. Also, since the improvements were made as the validation progressed, the effort to put a final version live within the HDS documentation was relatively low (Fig 36).

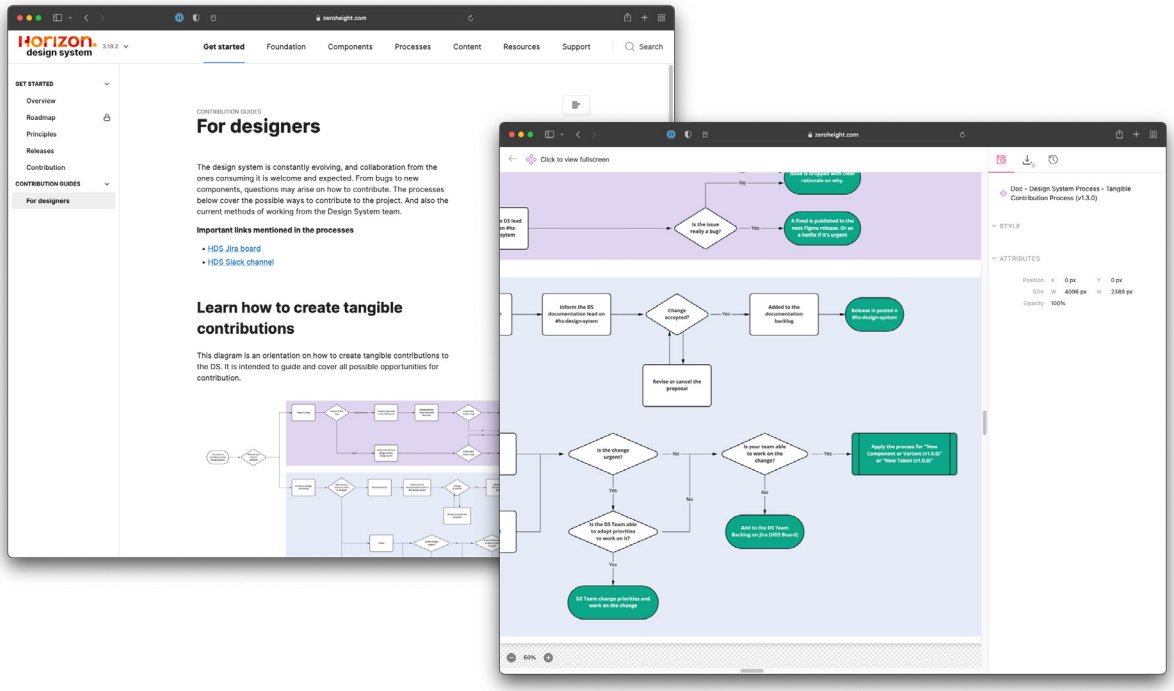

Figure 36 - Implemented solution within the HDS documentation.

The process diagrams built in the digital artboard tools were converted into image assets. Furthermore, the prototype created for the navigation was updated with the latest findings and published as an official documentation release. Finally, an announcement was made in a company-wide communication channel via Slack, allowing participants to interact by providing feedback or asking questions about the newly released documentation. The end of this last phase does not mean, however, that the problem is completely solved. Instead, it means that an iteration cycle has been completed, and that there is an immense opportunity to expand and improve the implemented solution.

4. Interviews with experts in the field

Stimuli: Semi-structured interview.
Method: Remote interview (45min).
Participants: Two participants, an independent DS and accessibility consultant and a senior product designer from a global online retail company. Both with years of experience dealing with DS and related processes.

The objective of this section was to have the processes evaluated by experts in the fields. The idea was not to have them look into the details of the processes but to validate if the approach as a whole would work. The conversations (Fig 37) revealed that some essential components for the project's success were missing. However, the overall approach was correct so that these issues could be worked on in future iterations.

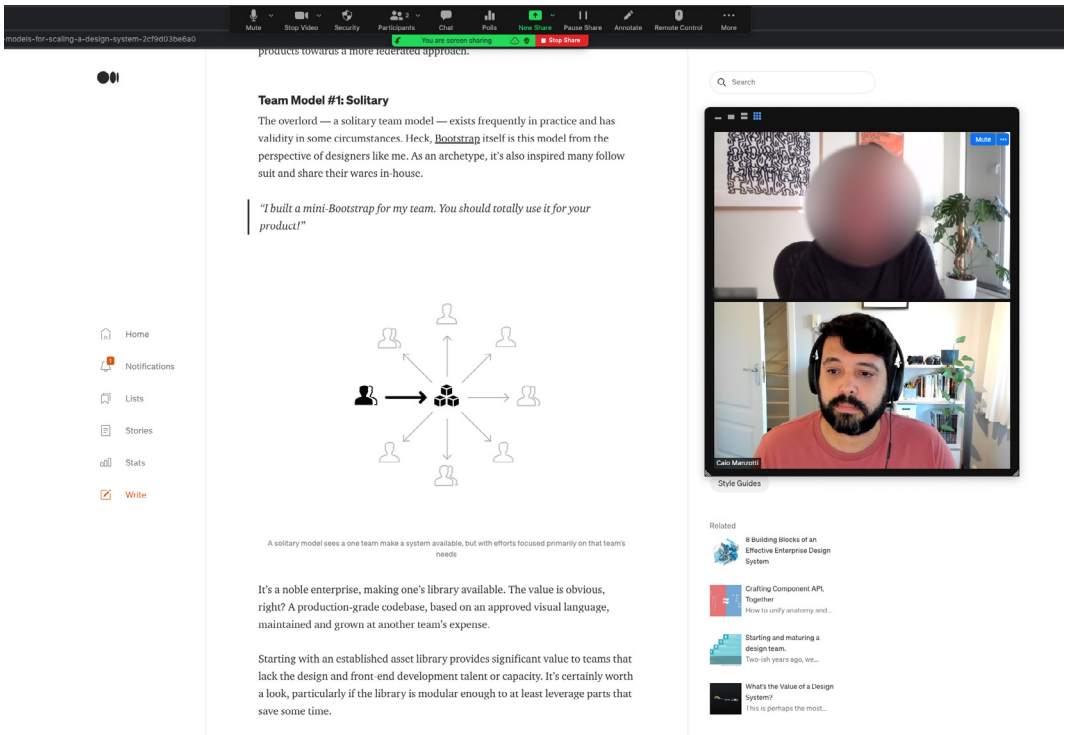

Figure 37 - Discussion about DS team decentralisation with an expert in the field.

Some of the main highlights and learnings from these conversations were:

General insights

- Decentralising a DS team with the help of a mapped process is indeed common for large organisations. However, a core team of DS individuals should be maintained to support contributors and plan the strategy of a DS.
- There should be clear agreements in place regarding the responsibility and accountability for applying these processes.
- The adoption of such a solution will only happen if designers feel they have ownership of it. The DS team should not act as police to validate its usage. They should instead be portrayed as supporters and facilitators for the ones applying it.

- This type of documentation is beneficial for new joiners in the company. The HDS and its content should be part of the onboarding process, so these new people know how to use and contribute to it early on.
- A successful DS depends on multiple people building and using it. It benefits from diverse feedback and collaboration.

Insights regarding the solution

- Accessibility should not be an isolated step since it is not a feature. To be truly accessible, designers have to consider it in every design decision. This includes even the prioritisation before deciding what DS component needs to be built.
- The brand visual principles for the company and its products were left behind in all three process diagrams. They should be incorporated at the beginning of all processes, helping decision making regarding visual aspects of the created DS elements.
- The lack of confidence from designers is expected once a solution like this is in place. This tends to improve once they learn how to navigate the processes. The DS team should enforce a series of learning by doing agreements, helping these designers be more self-sufficient and more confident with these tasks.

The learnings obtained from interviewing these specialists were surprising. It shows that exposing this type of idea to external stakeholders can benefit the generation of insights. The identified opportunities will be included in the recommendation for the next steps.

5. Next Steps

While the implemented solution is a solid take on the problem, it is just the start, and more stakeholders need to be included in the strategy of decentralising the DS team and mapping its processes. Also, the chosen solution is still relatively static, depending on the user interest to happen. Therefore, new approaches to turn it into more actionable content need to be explored.

This research and its results will now be presented to its principal sponsors, the company leadership team. This meeting is yet to be scheduled but will still happen in 2021. Besides the results, a series of recommended actions for the following quarters will also be presented, with the main suggested points being:

Q1/2022

- Implement the learnings identified from the interviews with specialists, especially those related to accessibility and brand principles.
- Keep a constant and continuous channel with early adopters of these processes. So unforeseen cases and circumstances can be mapped and integrated into the existing flows.
- Start the evaluation of potential new process diagrams. A strong candidate is the one that covers the same processes but for developers.

Q2/2022

- Keep a constant check with stakeholders on the challenges of using the proposed process. The DS team should start moving to a more strategic and federated approach by doubling down the efforts on this type of initiative.
- As learned from the literature review, to be effective, feedback needs to be in context. Solutions such as plugins for Figma, or automated bots on Slack can reduce the friction to find information by making its content more dynamic.

6. Conclusion

6.1 Contribution

Over the last weeks, the problem in the workspace continued, proving that the identified issues were valid and the prediction that they would persist was correct. Luckily, this project had the necessary company support and resources to explore possible solutions. The research, interview and constant validation with users demonstrated that the original strategy of not supporting external contribution was the leading cause of the problem.

Thanks to the framework applied, the team is confident that the implemented solution had its risks mitigated. This preliminary study concludes that the mapped process diagrams will serve its objective of guiding designers into creating tangible contributions for the HDS. The implemented solution can change the way designers from product teams collaborate with the DS team, by creating opportunities for them to work together. In addition, while exploring solutions, the DS team was also able to map its process, resulting in a more organised and transparent way of working.

As a commitment to this new and more decentralised way of working, a new principle focused on collaboration will be added to the HDS principles. The "Open to collaboration" principle reinforces the need for tangible collaboration for the project's success.

6.2 IRP Reflection

To better structure my reflection, I've used the Gibbs Reflective Cycle (Gibbs, 1988). In addition, I've incorporated a routine to write bi-weekly about my experience with this research project. The reflection below is the last iteration of this process, showing how the project impacted me:

Description
The opportunity to work on the Industry Research Project came at a perfect time for me. I had a challenging problem in the workspace, but none of the previous modules provided the necessary time and depth. With the support and trust of my manager, teammates and many other stakeholders, this work is now done. This project opened opportunities for collaboration and discussion on topics that were on hold for too long. And its results will hopefully support a new way of working for my team and me.

Feelings
I've been thinking about the theme for this paper for quite some time. Although, to be honest, I believe it has been since the course started. So when it was time to start, I felt thrilled to be able to dedicate time to dive in-depth into a specific topic. In addition, my day-to-day work projects don't tend to take this long, and I was curious to learn what would happen if this time was available. Now that it is finished, I feel fulfilled by creating a piece of work that makes me proud and will help me progress in my career.

Evaluation

The research result showed that the process that supported my team and me to be successful became a blocker. But the research showed that it indicates that we are maturing and growing. It also showed that adaptation will be essential to scale from now on, so we need to incorporate a way to keep improving the way we work.

Analysis

The research, especially the literature review, provided valuable insights on ways to solve our problem. We could have solved the problem in the workspace earlier if we had dedicated more time looking for references and even empathising with users. We need to diversify the way we collect insights for projects. Looking not only into competitors but also for the knowledge provided by parallel fields and academia.

Conclusion

After this project, I've become not only a better professional but also a better person. The learnings obtained about the topic and ways of working will support all my future endeavours. I now understand that some cases cannot be rushed or solved in a week's sprint. Otherwise, we may lose essential insights on moving fast. This project would have failed or would have been shallow if it had been done in less time. Not that every project needs to last six or more months, but we need to create space for projects of different sizes.

Action Plan

From today ahead, I will always make sure that I incorporate two essential learnings from this experience. First, I will keep learning how to improve the way I research and apply critical thinking. Second, I will dedicate time and space to build up trust with my peers. Otherwise, goals become extremely hard to achieve.

7. References

Adevinta (2021) Adevinta Brands, Adevinta. Available at: https://www.adevinta.com/brands/ (Accessed: 1 October 2021).

Alexander, C., Ishikawa, S. and Silverstein, M. (1977) A pattern language: towns, buildings, construction. New York: Oxford University Press.

Aronowitz, K. (2018) Designers Finally Have A Seat At The Table. Now What?, Fast Company. Available at: https://www.fastcompany.com/90156186/designers-finally-have-a-seat-at-the-table-now-what (Accessed: 28 October 2021).

Bedingfield, W. (2019) 'The rise and fall of Flash, the annoying plugin that shaped the modern web', Wired UK, 18 September. Available at: https://www.wired.co.uk/article/history-of-macromedia-flash (Accessed: 8 November 2021).

Benlian, A., Koufaris, M. and Hess, T. (2011) 'Service Quality in Software-as-a-Service: Developing the SaaS-Qual Measure and Examining Its Role in Usage Continuance', Journal of Management Information Systems, 28(3), pp. 85–126.

Berkun, S. (2020) How Design Makes the World. Available at: https://www.overdrive.com/search?q=43940B75-6BB0-4F7D-AEC4-EE6F840AC974 (Accessed: 3 October 2021).

Biswas, D. (2019) Architecture and UX Design: the obvious parallels, Medium. Available at: https://uxdesign.cc/architecture-and-ux-design-dacefa2a6478 (Accessed: 9 November 2021).

Buley, L. (2017) 'External consultancy recommendations for Design Maturity'. Unpublished.

Buley, L. (2019) Design Maturity Model by InVision: The New Design Frontier, Design Maturity Model by InVision: The New Design Frontier. Available at: https://www.invisionapp.com/design-better/design-maturity-model/ (Accessed: 27 October 2020).

Chen, M. (2020) The Zendesk Triple Diamond process, Medium. Available at: https://medium.com/zendesk-creative-blog/the-zendesk-triple-diamond-process-fd857a11c179 (Accessed: 1 November 2020).

Collopy, F. (2019) 'Why the Failure of Systems Thinking Should Inform the Future of Design Thinking (06.07.09)', Design Issues, 35(2), pp. 97–100. doi:10.1162/desi_a_00538.

Curtis, N. (2015) 'Team Models for Scaling a Design System', EightShapes, 13 October. Available at: https://medium.com/eightshapes-llc/team-models-for-scaling-a-design-system-2cf9d03be6a0 (Accessed: 7 October 2021).

Dawson, M. et al. (2010) 'Integrating Software Assurance into the Software Development Life Cycle (SDLC)', Journal of Information Systems Technology and Planning, 3, pp. 49–53.

Donovan, R. (2020) Design Delight from Disability - 2020 Annual Report: The Global Economics of Disability | Return on Disability. Available at: https://www.rod-group.com/content/rod-research/edit-research-design-delight-disability-2020-annual-report-global-economics (Accessed: 18 October 2021).

European Commission (2019) European accessibility act. Available at: https://ec.europa.eu/social/main.jsp?catId=1202 (Accessed: 6 November 2021).

Ferreira, J., Noble, J. and Biddle, R. (2007) 'Agile Development Iterations and UI Design', in Agile 2007 (AGILE 2007). Agile 2007 (AGILE 2007), pp. 50–58. doi:10.1109/AGILE.2007.8.

Frost, B. (2016) Atomic design. Pittsburgh, Pennsylvania: Brad Frost.

Garrett, J.J. (2011) The elements of user experience: user-centered design for the Web and beyond. 2nd ed. Berkeley, CA: New Riders (Voices that matter).

Gibbons, S. (2016) Design Thinking 101, Nielsen Norman Group. Available at: https://www.nngroup.com/articles/design-thinking/ (Accessed: 12 November 2021).

Gibbs, G. (1988) Learning by doing: a guide to teaching and learning methods. London: FEU.

Gold, J. (2020) The Evolution of Tools, Airbnb.Design. Available at: https://airbnb.design/the-evolution-of-tools/ (Accessed: 5 November 2021).

Goswami, R. (2021) 'A better segmented control', Tap to Dismiss, 18 May. Available at: https://medium.com/tap-to-dismiss/a-better-segmented-control-9e662de2ef57 (Accessed: 10 November 2021).

Gurusamy, K., Srinivasaraghavan, N. and Adikari, S. (2016) 'An Integrated Framework for Design Thinking and Agile Methods for Digital Transformation', in Marcus, A. (ed.) Design, User Experience, and Usability: Design Thinking and Methods. Cham: Springer International Publishing (Lecture Notes in Computer Science), pp. 34–42. doi:10.1007/978-3-319-40409-7_4.

Harvard Business Review (2014) Giving effective feedback: check in regularly, handle convversations, bring out the best. Boston, Massachusetts: Harvard Business Review Press (20 minute manager series).

Hodson, B. (2017) Rediscovering Apple's Human Interface Guidelines from 1987, Medium. Available at: https://blog.prototypr.io/rediscovering-apples-human-interface-guidelines-1987-59731376b39e (Accessed: 8 November 2021).

Kalbag, L. (2017) Accessibility for everyone. New York: A Book Apart (A Book Apart, No 23).

Kholmatova, A. (2017) Design systems: a practical guide to creating design languages for digital products. Available at: https://catalogue.solent.ac.uk/openurl/44SSU_INST/44SSU_INST:VU1?u.ignore_date_coverage=true&rft.mms_id=990096960200204796 (Accessed: 14 November 2021).

Kuang, C. and Fabricant, R. (2019) User friendly: how the hidden rules of design are changing the way we live, work and play. London: WH Allen.

Lee, J. (2020) Let's talk about that GPT-3 AI tweet that shook designers to the core, Medium. Available at: https://uxdesign.cc/lets-talk-about-that-gpt-3-ai-tweet-that-shook-designers-to-the-core-d2b31ad3d63b (Accessed: 24 October 2021).

Lencioni, P. (2002) The five dysfunctions of a team: a leadership fable. 1st ed. San Francisco: Jossey-Bass.

Malone, E. (2017) 'A History of Patterns in User Experience Design', Medium, 31 March. Available at: https://erinkmalone.medium.com/a-history-of-patterns-in-user-experience-design-f21f7eaabb83 (Accessed: 1 October 2021).

Meadows, D.H. and Wright, D. (2008) Thinking in systems: a primer. White River Junction, Vt: Chelsea Green Pub.

Meyer, E. (2014) The culture map: breaking through the invisible boundaries of global business. New York: PublicAffairs.

Murphy, A. (2021) The Global 2000 2021, Forbes. Available at: https://www.forbes.com/lists/global2000/ (Accessed: 4 November 2021).

Park, W. et al. (2011) 'A factor combination approach to developing style guides for mobile phone user interface', International Journal of Industrial Ergonomics, 41(5), pp. 536–545. doi:10.1016/j.ergon.2011.04.002.

Perez Cruz, Y. (2019) Expressive design systems. New York: A Book Apart.

Santala, S. (2020) 'Optimising User Interface Layouts for Design System Compliance'. Available at: https://aaltodoc.aalto.fi:443/handle/123456789/45031 (Accessed: 5 August 2021).

Sparkbox (2020) The 2020 Design Systems Survey by Sparkbox. Available at: https://designsystemsurvey.seesparkbox.com/2020 (Accessed: 20 August 2021).

Sutton, S.M. (2000) The role of process in software start-up. Available at: https://ieeexplore-ieee-org.ezproxy.tees.ac.uk/document/854066 (Accessed: 28 October 2021).

Tanrim, B. (2021) Design system: LEGO kits syndrome. Available at: https://uxdesign.cc/design-system-lego-kits-syndrome-b5fe6a6ca53e (Accessed: 14 November 2021).

Tidwell, J., Brewer, C. and Valencia, A. (2020) Designing interfaces: patterns for effective interaction design. Third edition. Sebastopol, CA: O'Reilly.

Vesselov, S. and Davis, T. (2019) Building design systems: unify user experiences through a shared design language. New York: Apress®.

Wilkins, B. (2017) Sketching Interfaces, Airbnb.Design. Available at: https://airbnb.design/sketching-interfaces/ (Accessed: 13 November 2021).

A process for tangible Design System contributions

Caio Salles Manzotti