BelfiusWeb
USABILITY EVALUATION

Manon Haulotte

TABLE OF CONTENTS

I. INTRODUCTION

The purpose of this document is to **highlight the core usability issues on BelfiusWeb** (BWNT), in order to identify and prioritize opportunities for improvements, support informed decisions and ultimately optimize the overall user experience.

The conclusions drawn here are the result of an usability evaluation, during which the interface and flows were confronted to a sets of recognized usability standards and best practices. More specifically, BelfiusWeb was evaluated against the EEP Design Principles[1] and the Nielsen Norman's 10 Heuristics for User Interface Design[2]. The analysis was based on a cognitive walkthrough of BelfiusWeb, screen by screen, from the perspective of a user trying to accomplish tasks. This allowed to **focus on pure usability**, without taking into account previous knowledge of the product, of internal discussions, of business goals and of technical constraints. Whenever a flow was inefficient or an interaction disrupted the completion of a task, the screen was captured and annotated. Similar issues were later grouped, assessed against existing research (uLab reports, analytics, users' complains,...), then prioritized. You'll find the analysis of those key findings in section II.

While this exercice offers a picture of the overall system usability an a quick way to figure out what to prioritize, **such evaluation does not, and never intends to, replace actual user research and testing.** It's also important to acknowledge its limitations, without proper qualitative and quantitative data nor access to users, and with limited time and resources. The recommendations made in this document should therefore not be thought of as prescriptive, but are meant as a starting point for further investigation, like a backlog of actionable and prioritized insights for future usability efforts.

The evaluation was performed on the GTU environment, between December 4th, 2019 and January 3rd, 2020, using actorID NEEFT, and only includes desktop screens, starting after login.

[1] Set of collaboratively and internally created rules of what makes good design for EEP products - see appendix I
[2] Best-known industry conventions for usability, developed in 1994, used as reference for uLab's reports as well - see appendix II

II. KEY FINDINGS

The following insights are the result of a usability evaluation, during which each screen on BelfiusWeb was confronted against a set of heuristics, principles and best practices, as well as our current knowledge of users' needs. The uncovered issues were grouped into larger patterns, which were then prioritized based on:

1. how often they occurred (prioritizing the more repetitive)
2. how destructive they were for the completion of users' goals (prioritizing the more critical)
3. how they impacted the product as a whole (prioritizing the foundational issues over the granular ones)
4. whether or not they were confirmed by our current sources of user insights (usability testing, analytics,...).

The findings are presented here in order of priority (meaning that dedicating effort to the first items on the list will create a more positive impact on the user experience as a whole). For each of them, the problem is first explained and illustrated, then high-level suggestions for improvements are made. Please keep in mind that those recommendations should not be implemented as is, nor should they even be the main focus of this document. Rather, they are there to shine a different light on the problem, helping to better frame it by exploring alternatives in similar situations, and are to be thought of as avenues for exploration, only once the problem is thoroughly defined, researched and understood.

1. Know your Users

Every single one of the issues identified in the following pages stem from the same cause: a lack of first-hand knowledge and contextual understanding of what users need, how they work, what their frustrations are, what makes sense for them.

Referring to the simple principle that users will be satisfied with a product if the reality of the experience matches their expectations of it, it's obviously crucial to first get a good grasp of what those expectations (and needs, contexts, workflows,...) are. To quote the first EEP design principle: "*If we don't know who we're designing for, we cannot design the right thing, or the thing right*". This is even more essential due the inherent complexities of

designing a professional tool such as BelfiusWeb: the fact that end users are not always involved in the buying decision but will end up using the platform repeatedly in order to achieve very specific tasks, that the same tool should enable a wide variety of profiles to all be efficient and successful in their role, or that disruptive flows and frustrating interactions cost time and have a tangible negative impact on people's working life.

Having said that, efforts are ongoing to bridge that gap, to source ideas in unmet needs, to test assumptions with users and to gain a richer, more contextual and empathic understanding of the people using BelfiusWeb (EEP design principles, more systematic usability testings, monthly analytics reports, research on corporate users, and more), which makes it unnecessary to further expand on this point. Simply remember that none of the usability issues presented here can be properly solved without researching users, including analytics, testing regularly with end users and making time for continuous improvement.

2. Improve Findability

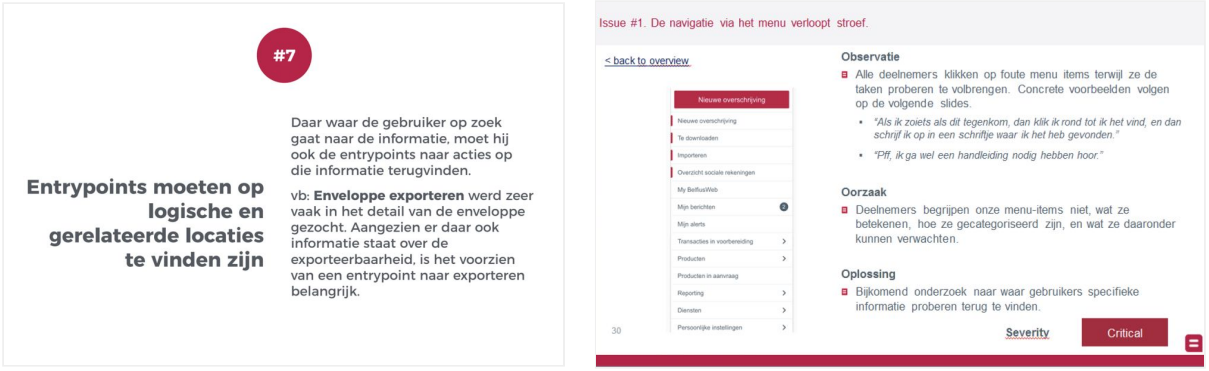
> "*Findability precedes usability. In the alphabet and on the Web. You can't use what you can't find*." - Peter Morville[3]

Findability is one of those words that people use *kind of* knowing what it means but without being able to explain what it actually entails. It is indeed a complex and interdisciplinary concept, so for the purpose of this document, I'll limit my definition of findability to: **the ability of a user to locate and interact with the information and functionalities contained within a website**.

Bridging the gap between users and the content and services they're looking for is obviously central in facilitating and enhancing their overall experience. The most innovative feature isn't worth much if it cannot be found, and getting lost online, searching for something specific, causes a lot of frustration. Unfortunately, all sources of user feedback we currently have point to a critical findability issue on BelfiusWeb. Which makes sense: it's a large platform, used by different profiles, whose tasks and needs are not always known, and built by multiple feature teams. So, let's have an in-depth look at what's not going well and what

[3] Peter Morville is considered to be one of the founding father of information architecture, and known for having popularized the concept of findability. He co-authored the best-selling book *"Information Architecture for the World Wide Web*" in 1998, which was used as a key resource when writing this section.

could be improved by exploring how the information is **organised** (information architecture), **connected** (primary navigation) and **contextualized** (local navigation) on BelfiusWeb.



A PlayCo study (May 2017, left) and a uLab report (June 2019, right), confirming unaddressed findability issues.

2.1. Information Architecture

Key concepts

Information Architecture (IA) forms the **skeleton of a website**. It's the structure that organizes where content and functionalities live, defines how pages relate to one another and labels them. By using categories and wordings that are familiar to the end user, a good IA helps users understand where they are, what is there and what's around. Just like finding a book in a library, thanks to the categorization by literary genre then by authors' names, a well-organized IA is invisible but allows for intuitive navigation and improved usability.

"*Information architecture starts with people and the reason they come to your site or use your app: they have an information need. It's important that you understand those needs and behaviors, and shape your designs to correspond accordingly. There is no goal more important to designing information architecture than to satisfy peoples' needs.*[4]"

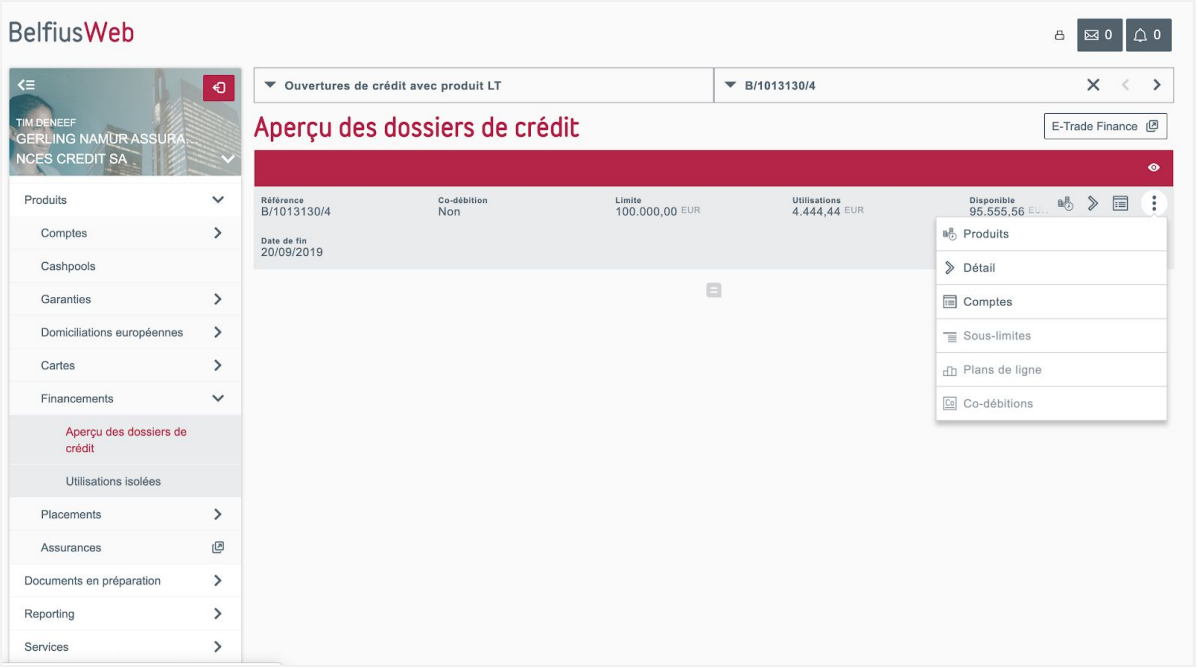
What's wrong?

- **Unbalanced breadth and depth:** On BelfiusWeb, users are often forced to scan a long list of entry points before locating the one they're looking for (=

[4] Information Architecture for the World Wide Web by Louis Rosenfeld, Peter Morville, O'Reilly Media, Inc, February 1998

breadth), then to click through more than two or three levels (and sometimes up to seven) before reaching the page they need (= depth), causing frustration as well as a feeling of heaviness and slowness.

- **The IA reflects the organisation's internal structure, rather than users' workflows**. Creating new pages and features without a good understanding of both how they fit with the rest of the website and how they will ultimately be found hinders findability and causes fragmented user journeys.
- **Unfamiliar and confusing labeling:** Entrypoints labels should speak the same language as the end users, helping them quickly identify those they already know, understand those they don't and know what to expect after they click on it. User testings have uncovered that the labels on BelfiusWeb often create confusion. For example: What's "My BelfiusWeb"? What's the difference between "Services" and "Products"? Between "Transactions being prepared" and "Documents being prepared"? Why are my accounts under products? What's under reporting?
- **Hidden pages:** as a consequence of the three previous points, content and functionalities on BelfiusWeb can end up buried and be hard to find for users.
- **Over reliance on quick fixes:** To compensate this, users are offered "Favourites". While shortcuts can be a useful part of a well-organised IA, they cannot fix a shaky foundation. And they should definitely not be the only entry point to key pages (e.g.: signing area).



Having had to scan a long menu and clicked through 4 levels already (Products > Financing > Credit Overview > LT product (group selector)), with unclear labels, users needing to make a straight loan withdrawal will still have to navigate two levels deeper before starting the flow (Products > Create a withdrawal, hidden behind three dots)

How to improve?

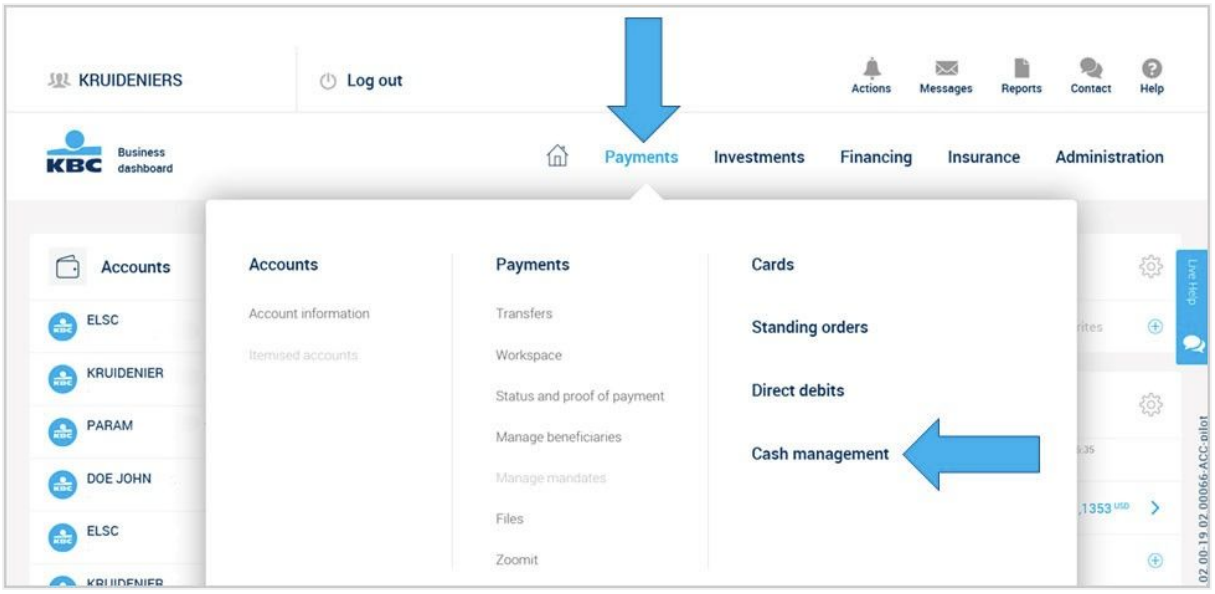
Because an IA cannot be restructured without input from domain experts and from users, I'll focus my recommendations on processes and approaches.

- **Evolving towards a task-based information architecture:** Users come to BelfiusWeb to perform a task (make a payment, sign a transaction, import a file, check a balance,...). Some of which are repetitive, some periodic. Some of those tasks always go together, therefore need to be linked. Some tasks always require the same information to be completed. Some will be performed primarily by a specific group of users, etc.

  It would therefore be possible to anticipate a number of high-priority tasks that users will want to perform on BelfiusWeb, as well as what they need to complete those tasks and how they relate to one another. This makes it the perfect situation to consider a task-oriented information architecture[5]. Task-based structures group pages not based on the function they offer to the end user. *" [They] use plain language words rather than company jargon.*

[5] Information Architecture for the World Wide Web by Louis Rosenfeld, Peter Morville, O'Reilly Media, Inc, February 1998

They provide clear paths to content, grouping similar material irrespective of which department or team is responsible for the content.[6]"



This is a good, and close, example of a (partial) task-based IA. While on BelfiusWeb, users have to find the product before the task (Products > Accounts > Payment), here the task comes first (Payments > products) Note how the mega menu allows users to skip a level or two to reach the page they need.

To implement such structure, a clear understanding of the tasks behind the content is needed. [7] This requires two things:

1. A good handle on the tasks available on BelfiusWeb, using a content audit and/or an updated sitemap in order to list the primary, secondary and related tasks of each page.
2. A better visibility on users' mental models and workflows. Top task analysis, card sorting and research will help creating a list of primary, secondary and related tasks per user group, as well as where they expect to find them and what language is meaningful to them.

[6] http://ephraimjf.com/2017/05/understanding-task-based-intranet-navigation-really-means/
[7] https://mattwatson.codes/articles/managing-large-website-using-task-based-navigation/
https://docs.microsoft.com/en-us/previous-versions/ms997506(v=msdn.10)

2.2. Primary Navigation: Menu

Key concepts

Navigation patterns are the visual manifestation of the information architecture, the elements that users can actually see and interact with in order to reach those pages. Primary navigation, more specifically, helps users understand where they are and allow them to move back and forth through different areas within a website. It's therefore accessible on every page of the site. On BWNT, the primary navigation includes the left menu and the icons in the top bar.

What's wrong?

- **Disorganised menu:** As a consequence of the IA problems previously explained, the entry points are not organised in a way that coordinates with user tasks, can have unfamiliar labels and are sometimes hidden, making it difficult for users to quickly scan the menu and find what they need.
- **Interaction cost**: It takes too many interactions for users to see the page they need. When opening a submenu item, they should directly be shown the page they clicked on, before having to click on the 2nd or 3rd level (i.e.: user clicks on "Product", then "Investment": she should already be redirected to the first page of the Investment menu, without having to select it)
- **Layout**: The menu eats up a lot of horizontal screen space, which is particularly inconvenient on BelfiusWeb, where that real estate is often needed to show large amounts of information (e.g.: tables).
- **Design:** There's too little contrast between the first and second level of the menu. The picture doesn't add any value and takes up unnecessary space. The direction of the arrows are confusing: if it points to the right, it indicates that clicking on it will take me to a new page, not open a submenu.
- **Favourites:** A button should be used to start a flow, not to direct to a page. Besides, the same action shouldn't potentially be shown in two different ways visually ("top" and "regular" favourites offer the same list of entry points but have a different design). Lastly, the absence of visual cues to modify the favourites limits the discoverability of the option, as confirmed in the analytics.

How to improve?

The redesign of the primary menu should account for the local navigation patterns. Suggestions can therefore be found at the end of the next section.

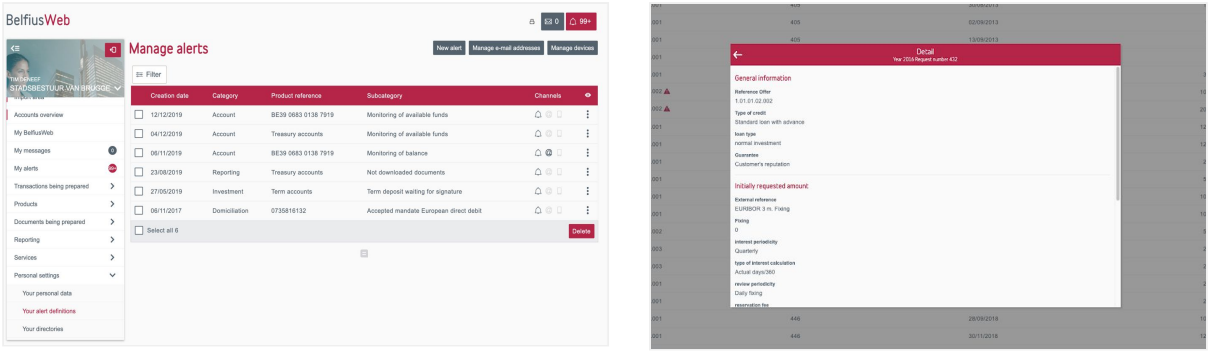
2.3. Local and contextual navigation

Key concepts

Built on top of the primary navigation, local and contextual patterns give flexibility to a system, enabling users to explore the immediate area by indicating what's related to the page they're currently on. On BWNT, they can be: the group and product selectors, the tabs, the overflow buttons and the regular buttons used as navigation.

What's wrong?

- **Lack of contextual navigation**: Contextual navigation cues improve findability and discoverability and provide extra value to users (especially expert and repeating users) by anticipating their needs, allowing them to bypass the primary navigation and offering direct links to pages related to their current task. On BelfiusWeb, however, users are usually forced to navigate in a very linear way, often having to click back through several layers of navigation before finding what they're looking for.



A good example of contextual navigation: On the left, the user is offered links to manage her email addresses and devices, anticipating what she might need to complete the current task. On the right, however, a user wanting to sign a transaction from the detail screen will first have to go back through to the whole list. Simply adding a "Sign" button at the bottom of the modal would alleviate the problem.

- **Inability to go back to the previous page using the browser**
- **Multiplication of navigation patterns on one page:** It's common to find group selectors, buttons, tabs, dropdowns and overflow buttons on one single

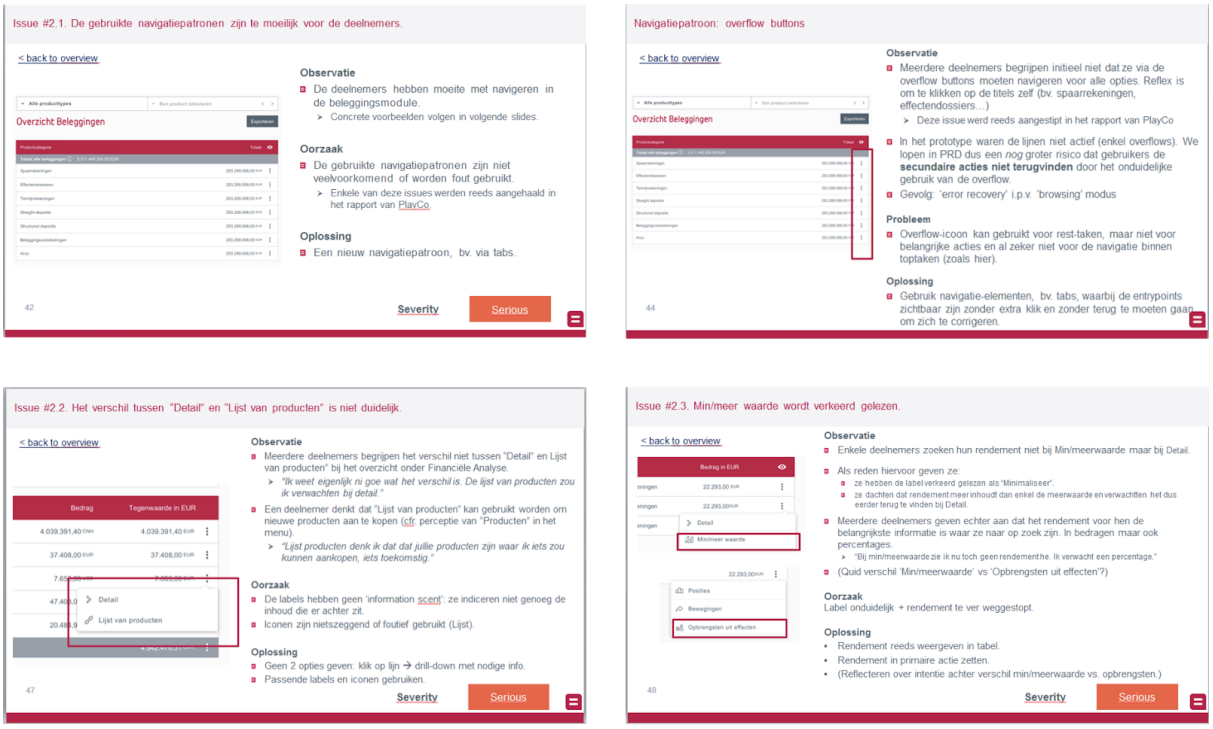
screen on BWNT, without clear rules of which elements does what, leading to confusion and a sense of heaviness.

- **Suboptimal local navigation patterns:** According to the theory of information foraging, if a page is going to be hard to find, it has to provide high value for people to put in the effort of searching for it. Unfortunately, as mentioned previously, it's all too frequent for functionalities on BWNT to be buried 5-7 layers deep.

  While this issue is rooted in an unstructured IA, it manifests in the use of navigation elements that make things difficult to locate:

  - **Group and product selectors** are confusing and considered a heavy navigation pattern. They clutter the screen, they hide information, they look like dropdowns (which are associated with filtering) but open a modal (which interrupts users and forces them to change focus)
  - **Buttons** are typically associated with actions, they should be used to start a flow, not to navigate between pages.
  - **Overflow buttons** (3 dots) should never be used as a navigation system. They are not suggestive, they don't have an information scent (which helps users to know what to expect when clicking on an element) and they hide available functionalities (especially with the current design that redirects users to the first option when clicking on the line). Key actions should be evident for users. This requires a proper prioritization and a thorough understanding of the user journeys: "*The crux of the issue is that overflow menus let designers, engineers, and product people off the hook from making tough choices. Instead of prioritizing, we just sweep complexity under the rug and pretend that it doesn't exist. Software with overflow menus is often still hard to use, even if the complexity is now beneath the surface.*[8]"

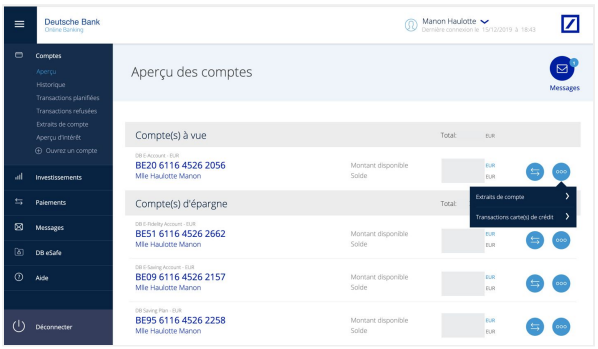
[8] https://www.freecodecamp.org/news/stop-the-overuse-of-overflow-menus-5caa4b54e843/



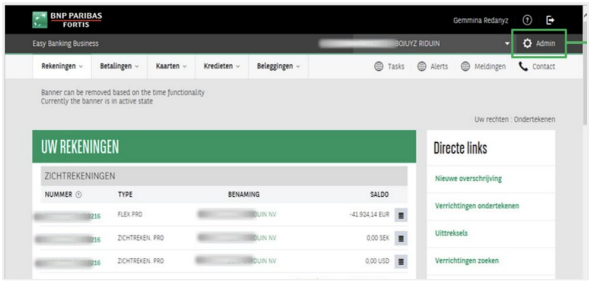
uLab report (Invest, June 2019), pointing out key issues with the local navigation patterns.

How to improve?

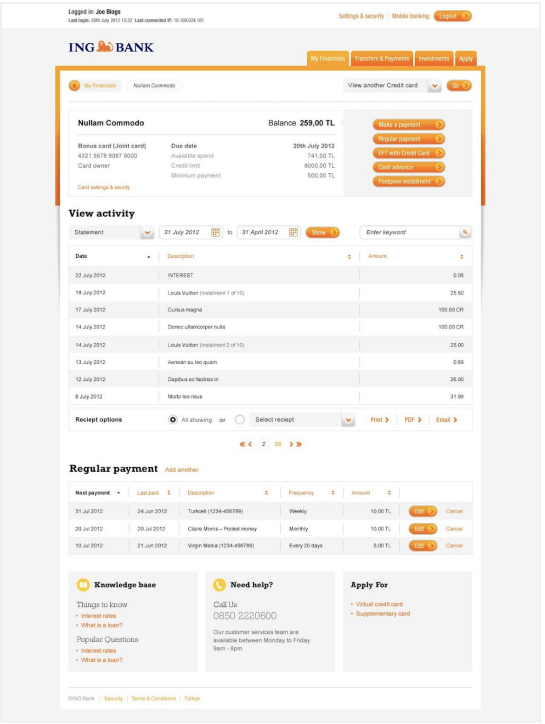
Please note that there's little sense in making changes to the navigation without first analysing the information architecture. Primary, local and contextual navigations should be integrated in a way they complement each other, without monopolizing too much screen space or overwhelming users. **Starting with user journeys** to inform navigation is the only way to design a good system that helps users understand which direction to take (rather than forcing them to bounce back and forth between pages). Lastly, **formalizing the navigation rules** and defining the use cases of each pattern, learning from best practices and user feedback, ensures consistency and reduces clutter. On the next page, you'll find inspirational examples of other banks' navigation systems. While they're not necessarily perfect, they contain interesting ideas and can help better understand what could be different on BWNT.



Primary navigation: The menu sticks to the left, leaving space for the important content. Thanks to the different colour treatment, it has enough visual weight, meaning it doesn't compete for the users' attention nor adds clutter. It's organized neatly, in clear sections, with icons adding visual cues. Clicking on a section will automatically redirect to the first page of that section. While the whole menu remains accessible at all time, each section is treated as a sort of portal, allowing users to focus on the task at hand, with the rest fading in the background. This is observable both through the design (slightly different colour once the item is selected and expanded) and the fact that each section has an overview page, like a mini dashboard giving access to all functionalities and key information related to that specific menu item. The contrast in the menu is clear enough to let me know where I am. Note how one key flow (Open an account) was selected to be accessible directly from the menu, and with a simple but effective visual cue that helps users predict the behaviour of that item. Lastly, on the top bar, users can access their settings and other tasks that are not directly related to the key uses of the platform, therefore don't need to be visible at all time. **Local and contextual navigation**: The left screen presents a list of accounts. Subtitles show all the different types of accounts on one page. One key action (Transfer) is highlighted. When clicking on the line or the account number, the history page opens. The history is accessible from the menu as well and each tab of that page is accessible from the overflow button. Meaning that no page or action is ever hidden. Note how the overflow button has more visual weight as well. On the history page (right screen), tabs are used, which are one of the strongest form of navigation, that allows to declutter the screen and reduce cognitive load while making it easy to switch between related pages and functionalities. They also leverage the power of accordions on that page: rather than on a modal or a new page, the details of each transaction is easily accessible on the screen by expanding the line. A key action (print) is also offered. Lastly, users can switch accounts at any point, without having to go back, thanks to the dropdown on the top.



On the left: The menu is on top, leaving the full horizontal space for the content. It's organised in clear sections. A drop down allows the user to switch accounts. Alerts and tasks are highlighted. On the page, there's contextual navigation with direct links to related tasks.



On the right: The menu doesn't take up a lot of space, and separates the platform in task-based sections. Note how the placeholder of the dropdown on top clarifies its purpose. Quick links make top tasks easily accessible. While the page is long (scrolling is still the fastest navigation pattern), it's well-organised and nothing is hidden (the filters are on page, there are clear buttons in the table,...). Related pages are accessible at the bottom.

3. Minimize Interruptions and Interaction Cost

> "Good design, when it's done well, becomes invisible. It's only when it's done poorly that we notice it. Think of it like a room's air conditioning. We only notice it when it's too hot, too cold, making too much noise, or the unit is dripping on us. Yet, if the air conditioning is perfect, nobody says anything and we focus, instead, on the task at hand." - Jared Spool

Evidently, users would rather interact with a tool without interruptions, and product teams would prefer to create flows that can be completed without eliciting any confusion or doubts. In reality, however, things are not always as easy and intuitive as we'd like them to be: users often have to wait for pages to load, scroll and look around to find entry points, switch attention to a popup window, consume mental energy to remember information, read through long lists, etc. This is called **interaction cost**: "*the sum of efforts - mental and physical - that users must deploy in interacting with a site in order to reach their goals*[9] ". Let's explore what this looks like on BelfiusWeb.

3.1. Modal flows

Key concepts

A screen can only be either modal or non-modal. While non-modal screens allow users to navigate back and forth between different pages and functionalities on a website, modal screens (dialogs, pop ups, overlays,...) will temporarily block access to the navigation and main window, while keeping it visible, until an action is completed on the modal window.

Because modal screens force people to focus on a single task before pursuing, they can be useful when needing to grab users' full attention. And because they don't take users away from the current page, they allow the presentation of relevant information with minimum context shift.

However, because modals are interrupting by nature, appearing suddenly, demanding an action from users and preventing them from interacting with the

[9] https://www.nngroup.com/articles/interaction-cost-definition/

background content, they can be a source of frustration when used in the wrong context. And because of their limited space, reduced flexibility and lack of accessibility, they create unnecessary limitations for product teams. For those reasons, design standards all recommend to limit the use of modality to situations where they have an obvious added value:

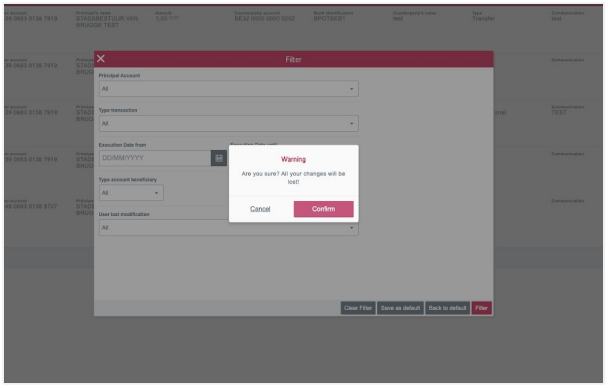
"*Dialogs are purposefully interruptive, so they should be used sparingly*" — Google's Material Design guidelines[10]

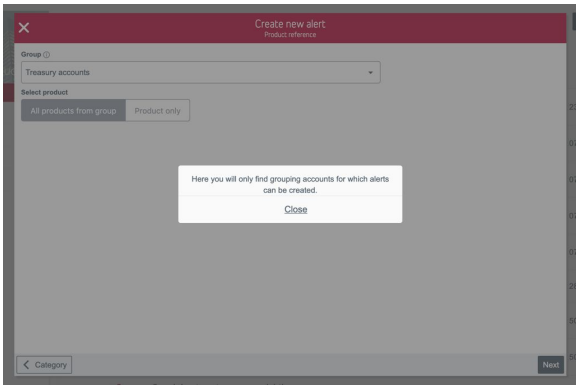
"*A modal experience takes people out of their current context and requires an action to dismiss, so it's essential to use it only when it provides a clear benefit.*" — Apple's Human Interface guidelines[11]

"*When it comes to modal dialogs, consider this: no one likes to be interrupted, but if you must, make sure it's worth the cost*." — Nielsen Norman Group[12]

What's wrong?

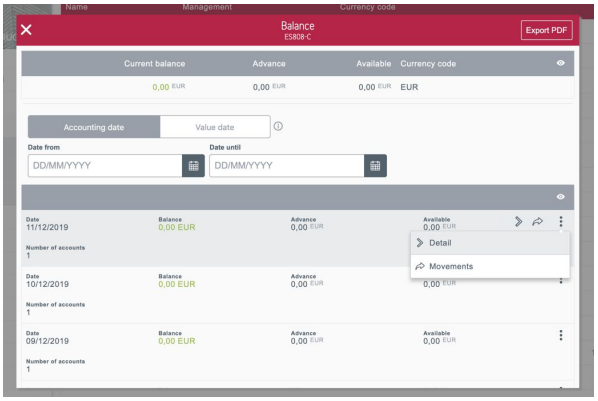
- **Multi-steps flows in modals:** On BelfiusWeb, nearly all flows are in modals. Making them the default creates a multitude of issues that could easily be avoided:
    - **Modals on top of modals:** It's very common to see a modal on top of another modal on BWNT. This makes the website look disorganised, creates nested flows and overwhelms users, making them lose track of where they are, forcing them to change focus multiple times and asking them to spend effort to close them.

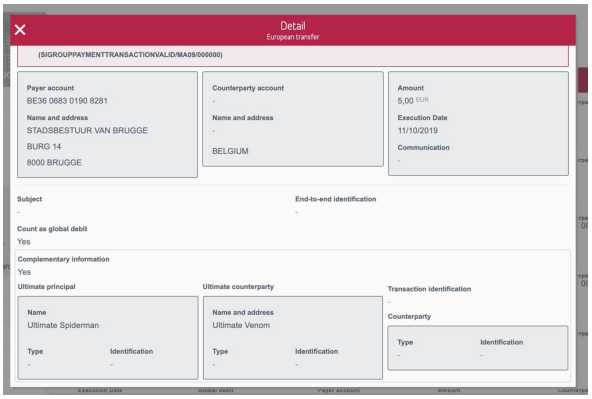




[10] https://material.io/components/dialogs/#usage
[11] https://developer.apple.com/design/human-interface-guidelines/ios/app-architecture/modality/
[12] https://www.nngroup.com/articles/modal-nonmodal-dialog/

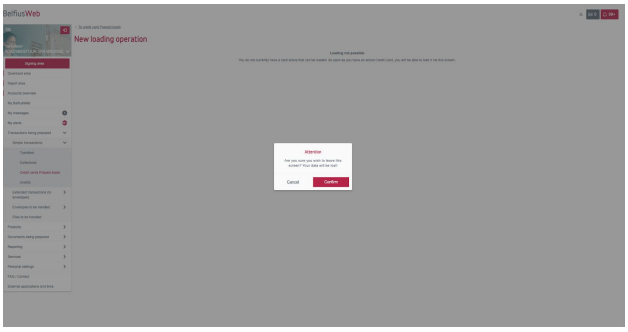
- **Visual complexity and clutter:** Modals lack flexibility and space, making it hard to fit complex content, such a tables.

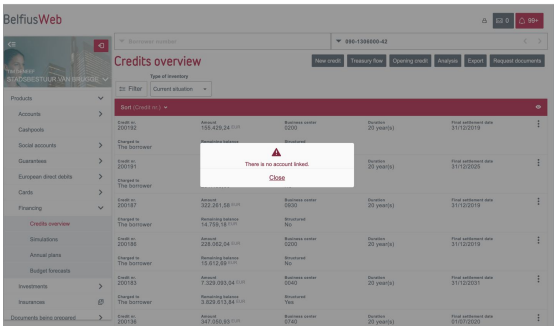




- **No progress indicators:** It's important to help users retrace their steps by letting them know what step they completed and what to expect next through a progress bar, or even simple dots.

- **Overuse of modals, in the wrong context:** On BelfiusWeb, modals often ask users to confirm non-destructive and routine actions, which overemphasize the transition, interrupts users flow, and creates annoyance, especially for regular users. They are also used to display errors and empty states, for filters, for details, for help,... making them lose the impact they could have if used correctly and in the right context.





Left: Confirming to leave an empty screen. Right: This could've been displayed in a less disruptive way.

- **Design:** The titles lack visibility, the buttons don't have clear labels.
- **Interactions:** Lack of smooth transitions (e.g.: When selecting a tile, users should go directly to the next step without having to click on "next")

How to improve?

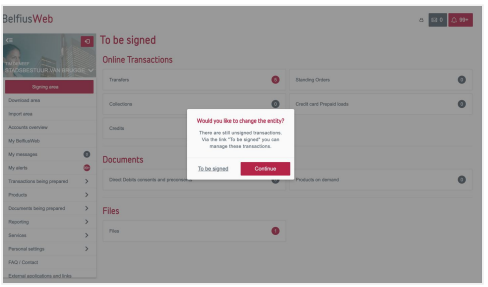
First, analyse each task flow to understand if the interruption is warranted or not, ideally relying on user insights and best practices.

- **When to consider modal screens**: when it has a clear added value.
    - To deliver crucial information at the appropriate time
    - To confirm a destructive action that cannot be undone (although “undo” should be offered when possible)
    - For self contained processes, with a clear start and end point and related to the main task (e.g.: a user is creating an alert and needs to quickly update the email address she’ll use to receive the alert)
    - Keep them simple, short and narrowly focused, with clear titles, buttons and call to actions.
    - Make sure the modal is screen reader- and keyboard accessible.
    - If it’s a routine action, let repetitive users bypass the modal (“do not ask me again”)

- **When to consider non-modal screens:** for everything else.
    - If an interaction requires more than 1 or 2 steps or substeps
    - If a scrollbar is necessary in the modal to show all the content
    - If there’s a need to launch a new modal on top of a modal
    - If a table or another complex visual element needs to be included
    - Keep error, success or warning messages on the page, and any action that can easily be reversed or won’t have critical consequences (leaving the transfer page, switching entities,...).

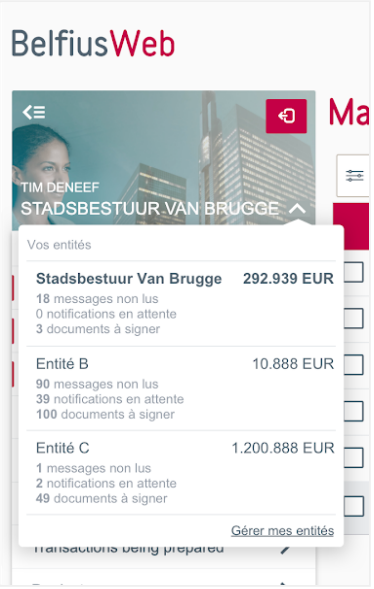
- **Explore alternatives:** Because of the limitation of modals, less-intrusive ways to show additional content are becoming more and more common: non modal overlays, accordions and collapsible sections, dropdowns, on-page warnings, drafts, “undo” buttons,...

An example of a modal being misused is the entity switch on BWNT. Currently, it interrupts users twice, first



to confirm the switch, then to select the entity.

An alternative could be a simple dropdown. In a non-obstructive way, it gives users the information they need, let them know if there's anything to pay attention to and empowers them to take the right action for them, knowing they can easily switch back if they change their mind.



3.2. Error handling and prevention,

3.3. Providing help in context

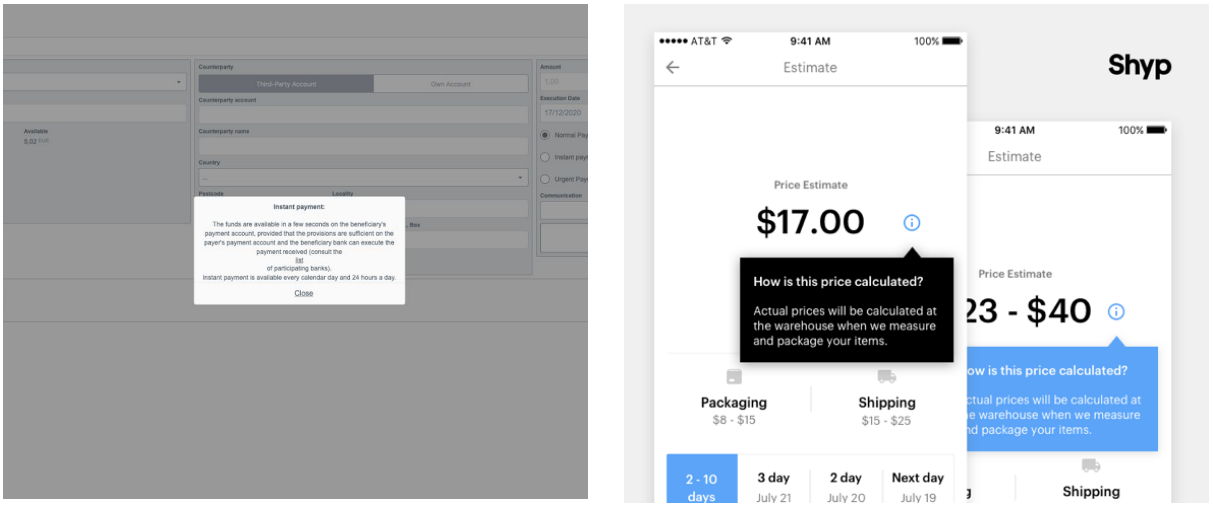
Key Concepts

While it's true that "*a user interface is like a joke; if you have to explain it, it's not that good*" (Martin Leblanc), this doesn't mean that a good interface shouldn't offer help. Quite the opposite, actually: a self-explanatory design will guide users as they're

completing their task, providing the necessary assistance exactly when it's needed and in a way they understand it, allowing them to get unstuck and move on without interrupting the flow to check an external help page or call the support center. A bad user experience uses help as a crutch for poor findability and confusing interfaces. A good user experience anticipates problems before users have to solve them. This ability to provide help when, where and how a user needs it is known as contextual and embedded help. It requires a strong knowledge of users' pain points and a continuous collaboration with both web analytics and support teams, to understand where users have the most difficulties. Currently, there are three patterns used to provide help on BelfiusWeb, presented here from the most to the least contextual.

What's wrong and how to improve?

- **Popup tips (vs Tool tips):** The most common form of embodied help, a tip is a message triggered by the user that provides additional information about the element it's attached to (e.g.: an "i" icon next to a label). Unlike a user manual, they're concise and narrowly-focused and contain the smallest possible chunk of information needed to clarify something. They can be very useful to provide extra information to first-time or less-experienced users, for example. If the information is essential to complete the task, however, it should be on the screen, not in a tip. Also, while popup tips appear in a modal hence unnecessarily disrupt the flow, add visual clutter, force users to shift focus and to manually close the pop up, tooltips display the same information on the screen after hovering over or clicking on the icon. They are less disruptive and therefore recommended here.

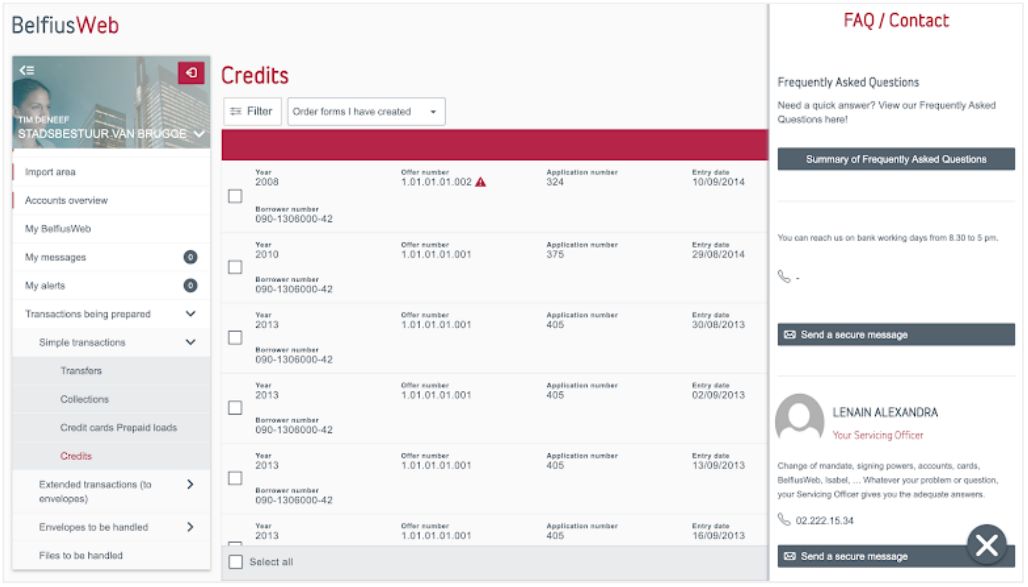


Left: the popup tip is not concise enough, might contain important information that should be available on the

screen, especially since it redirects users to another page.. **Right:** the tooltip doesn't disrupt the task, anticipates users' questions, has one clear focus (bolded question) and provides a concise answer in user-friendly terms.

- **Help on page:** A new pattern on BelfiusWeb is to offer access to the FAQ directly on the page, via an icon on the bottom right corner (not yet available on all pages). This is a great idea, but it could be made more contextual. Rather than simply providing an entry point to the FAQ or to the “send a new message” flow (which will both take users out of the flow), it would be helpful to only show a pre-filtered list of questions (with their answers), relevant to the task at hand.

  Ideas to explore: embedded chats (relevant to the page, e.g. connect with someone who can help complete a form), interactive walkthroughts and guided tours (triggered by users, see DB)



APPENDIX 1: EEP Design Principles

1. **We are driven by users insights:** If we don't know who our users are and what they need, we cannot design the right thing, nor design the thing right. That's why every decision we make are led by actual user insights, not assumptions or personal preferences. We do research. We observe and talk to users. We analyze data. We test early and often. And we aim for continuous improvements as soon as we release.

2. **We take care of all our users:** EEP users are diverse, in terms of their role, sector, level of experience,… Knowing this, we aim to provide value for each and every one of our users, rather than creating a one-size-fits-all solution. We therefore not only do the work necessary to ensure our designs are accessible, intuitive and relevant for everyone, we also provide personalised solutions that improve our users' individual workflows whenever possible.

3. **We help our users work better**: Time and effort saved by our users is more important than our time and effort. Because our tools are used mainly to execute professional tasks, we strive to respect our users' time and facilitate their job by streamlining and optimizing workflows, by limiting distractions, guesswork, unnecessary actions and information, and by making our products as easy to learn and intuitive to use as possible.

4. **We are a trustworthy partner to our users:** We increase confidence and trust by making our tools self-explanatory, by proactivaly anticipating our users' needs, by keeping them in the know of any issue or relevant information, by actively reducing frustration, uncertainty and stress and by providing support whenever needed.

5. **We optimize for large data sets:** Because of the nature of our users' activites, we often have to deal with high volumes of data, which can be slow to process or complex to display. While aiming for a good balance between performance and design when possible, we need to ensure that users are always able to vizualize all the information they require (no more, no less) at each moment

APPENDIX 2: Nielsen Norman's 10 Heuristics for User Interface Design

https://www.nngroup.com/articles/ten-usability-heuristics/

1. **Visibility of system status:** The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

2. **Match between system and the real world:** The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

3. **User control and freedom:** Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

4. **Consistency and standards:** Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

5. **Error prevention:** Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

6. **Recognition rather than recall:** Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

7. **Flexibility and efficiency of use:** Accelerators — unseen by the novice user — may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

8. **Aesthetic and minimalist design:** Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

9. **Help users recognize, diagnose, and recover from errors:** Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

10. **Help and documentation:** Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

RESOURCES

Existing user research:

- uLab report, Invest, June 2019
- PlayCo study, May 2017
- Web analytics (?)

On findability, information architecture, navigation:

- Information Architecture for the World Wide Web by Louis Rosenfeld, Peter Morville, O'Reilly Media, Inc, February 1998
- https://www.invisionapp.com/inside-design/information-architecture/
- Kevin Larson and Mary Czerwinski, Microsoft Research, "Web Page Design: Implications of Memory, Structure and Scent for Information Retrieval"
- https://www.nngroup.com/articles/menu-design/

On tables and large data sets:

- https://www.uxmatters.com/mt/archives/2006/08/refining-data-tables.php

On modals:

- https://www.nngroup.com/articles/popups/
- https://uxplanet.org/modality-the-one-ux-concept-you-need-to-understand-when-designing-intuitive-user-interfaces-e5e941c7acb1
- https://www.nngroup.com/articles/modal-nonmodal-dialog/

- https://www.nomensa.com/blog/2014/how-improve-accessibility-overlay-windows-part-1
- https://www.smashingmagazine.com/2014/09/making-modal-windows-better-for-everyone/
- https://medium.com/pulsar/modern-enterprise-ui-design-part-2-modal-dialogs-2ccd3cc33c92

On error prevention:

- https://www.nngroup.com/articles/user-mistakes/?lm=confirmation-dialog&pt=article

On providing help:

- https://pronovix.com/blog/overview-context-sensitive-and-embedded-help-formats
- https://www.nngroup.com/articles/faqs-deliver-value/