ARCADIS gen

—

UIXA guidelines

—

Arcadis Gen User Interface, User
Experience and User Accessibility
guidelines and best practices

14 May 2020

Contacts

Raymond Delno
UX engineer

m +31 (0)6 270606056

e raymond.delno@arcadisgen.com

Arcadis Gen.

Contents

1 Introduction

This document intends to give guidelines when creating products or solutions to maximize the customer and user experience. This is divided into three main chapters:

1. Customer experience (CX)
2. User Interface (UI)
3. User experience (UX)

Customer experience is overall experience the customer has when interacting with us; when having the first initial meetings, when agreeing on and setting up contracts, when using our products or solutions, when calling our helpdesk and when receiving our invoices.
This document focusses on the part of CX when creating a new product or solution. Help-desk or other customer engagements are out of scope of this document.

User interface is the look and feel of our products or solutions; a recognizable 'look & feel' that ensures a consistent and enjoyable environment that the user interacts with. Although the users can be the same as the customer often the users are a subset of the customer employees, specifically in large businesses.

User experience is the way the User Interface works. Is it arranged logically? Does it work as the users expects? Is it intuitive? When our User Interface does not need any explanation and the user can accomplish his/her goal, our created User Experience has succeeded.

2 Customer experience (CX)

What makes a user value the experience they have when using our projects? What do we need to do to make the user experience a <u>good</u> experience?

2.1 Design process

The process of creating a good user experience begins with 9 steps (designed by the British Design Council):
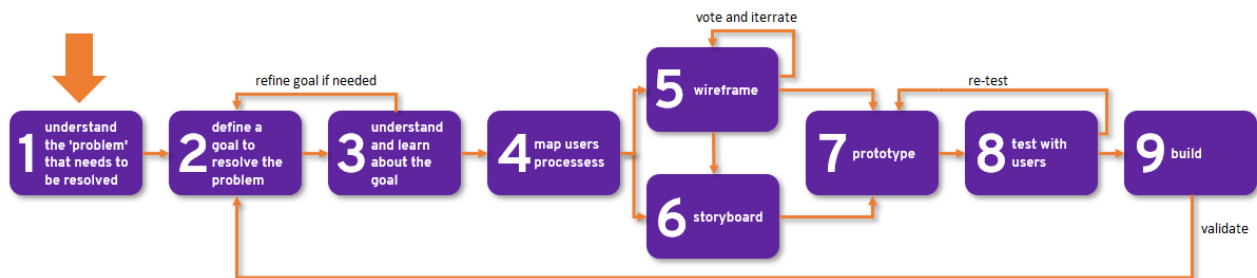
2.1.1 Understand what needs to be resolved

The process starts with defining what the actual problem your trying to solve is. This applies to both an existing product and for a brand new to be developed product. Try to formulate multiple statements that define the problem. For example:

- The user has multiple assets. Maintaining these is difficult and differ from asset to asset due to storage in different systems. Creating reports is therefore not easy and consistent,
- Users often go for grocery shopping at the same time, causing congestion at times. Spreading of shopping times would help. But how do you 'force' clients to come at different times?

Important is to understand that the team-members that are set to solve this problem (to build the product) probably will not be the key users. using this knowledge, find and understand the real issue. Investigation is key here!

2.1.2 Form a team goal

When the statements have been defined, collaborate with the product team to come with a single team goal that serves the purpose of solving the client's problem, request or need. Purpose of this is to thoroughly understand the problem.

2.1.3 Understand the goal

After forming the goal, learn about it! Research and learn by asking the user what they experience when they encounter the actual problem. Ask what they try to accomplish and how they do this at this moment? Ask what they like about this process and what they dislike? And often a very helpful question to gain insight is what the user would change in the process if they had a magic wand? It forces into thinking without boundaries. And keeps the process fun!

Important when interviewing the user is to do this in the user's natural environment. And let the user do the talking, let the user finish his/her sentences. Observe!

When the interviews have taken place and the team has learned, match the learned against the goal. And redefine the goal if needed!

2.1.4 Process mapping

The next step is to understand which players are involved, what steps they follow and what their goals are. The previous step, where the team goal was formed, might help in understanding what players are involved.

Setting up a schema with players (users), their steps they follow and their end goal. There can be mutual steps and/or goals for different players.

This schema can then be used to match against the team's goal, defined previously.

2.1.5 Wireframe

The wireframing process involves drafting rough sketches of how the product should look like. Do this quick and dirty, no need to come up with a smooth final solution. It can help to fold a paper in 8 equal squares and draw a sketch in each frame (crazy 8's technique).

Each team member can do this individually. After this the team members present their own ideas to the group after which the team can vote on other's ideas.

Based on the idea that got the most votes, sketch a final idea.

2.1.6 Storyboard

To create a storyboard, combine the process mapping and the wireframes. This outlines the user's workflow and shows the frontend of the product. Basically, the storyboard brings the process mapping and the wireframes to a coherent 'flow' that should result in the defined team goal.

2.1.7 Prototype

Using the storyboard and wireframes, build a prototype. Creating the prototype, it's not about what tools to use. It's about the process and how the product would look like. It should reflect the steps for all users from the process map and follow the chosen wireframe. Building a prototype should be a matter of days, not weeks (depending on the size/complexity of the product).

2.1.8 Test

Test the prototype with real users! This is important to receive the correct feedback. While testing do not instruct user how to do what they're supposed to do. Let them fail if the prototype is not clear. Take notes! And ask questions like:

- What do they like (about the prototype)?
- What do they dislike?
- Are there any issues/problems?
    - Why did a certain flow work?
    - Why didn't a certain flow work?
- What would they like to see improved?
- Does the design and flow meet their needs?

]

Test again

After testing, analyse what worked and what did not work. Learn from it and revise the design. After this, test again with the users.

Once the team and the users are happy about the process, front-end and end-results, build the product!

It can happen that the prototype did not solve the client's problem or meet their expectations; it completely missed its goal. Although this might feel like a major failure, it's likely to have cost less than having built the entire product. And the team learned again about the user's expectations, it's pain points and what not to build. Start over again and use the learned to build a matching product.

2.1.9 Build

Once the prototype has been approved by the user, build the actual product. During this phase, keep having user tests to validate the built steps. The user has been involved in the creation process from the beginning, they have invested in the product. They agree with the prototype and thus like the product to be build. Gather their feedback in the final stage at specific build milestones to keep on track.

2.2 User experience by a good user interface

The above 9 steps force the team into truly understand what needs to be built to solve the user's problem and meet their demands. How the user experiences this, is defined by the final end result *and* by interacting with the device that serves this end result (often, but not limited to, a computer). This interaction between human and (computer) device is the user interface. The user interface supports a good (or bad) user experience.

The human-computer interface is not limited to buttons, text and a like on the screen. It consists of all input/output devices the user uses. Like the monitor (visual output), the mouse (tactile input), tactile feedback (tactile output), sound (audible output). A perfectly designed user interface will ultimately fail if, for instance, the monitor is set too dark (user has problems seeing what's on the screen), the mouse does not work (user cannot input their commands) or sound is distorted or absent.

When developing products for clients, we cannot control their hardware (fix their mouse or adjust their screen brightness). We can control what is on the screen and how the user interacts with it. This is the user interface we need to design and build. The interface that supports the user's experience.
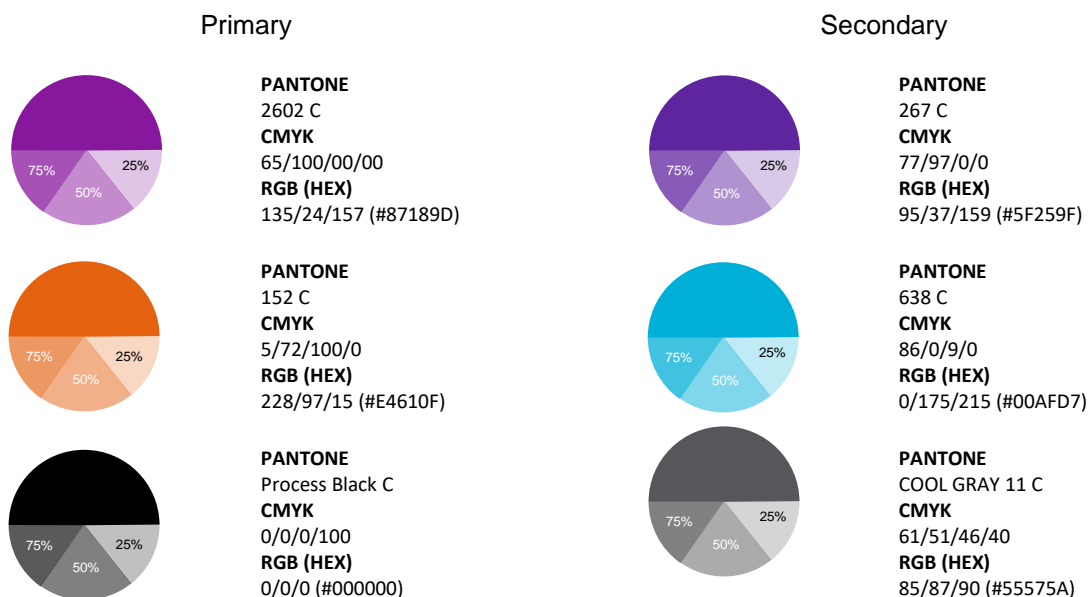
3 User interface (UI)

The user interface is the connection between the user and the product or solution. Though a product or solution often is much more than its front-end, from a user perspective, the user interface *is* the product. It's what he/she see and interacts with. Therefor special care needs to be taken to the user interface, it cannot be done -in between- the product development!

A recognizable user interface starts with using our brand features as colours, shapes, fonts etc. It's about how our products look. Our brand should be reflected in this look.

3.1   Colours

The default colours to be used within the Arcadis Gen brand are divided in primary colours and secondary colours.

Primary   Secondary

PANTONE   **PANTONE**
2602 C   267 C
CMYK   **CMYK**
65/100/00/00   77/97/0/0
RGB (HEX)   **RGB (HEX)**
135/24/157 (#87189D)   95/37/159 (#5F259F)

PANTONE   **PANTONE**
152 C   638 C
CMYK   **CMYK**
5/72/100/0   86/0/9/0
RGB (HEX)   **RGB (HEX)**
228/97/15 (#E4610F)   0/175/215 (#00AFD7)

PANTONE   **PANTONE**
Process Black C   COOL GRAY 11 C
CMYK   **CMYK**
0/0/0/100   61/51/46/40
RGB (HEX)   **RGB (HEX)**
0/0/0 (#000000)   85/87/90 (#55575A)

See also the brand guidelines: 'Arcadis_Gen_Brand_Guidelines_v6a.pdf'

Next to the above colours, Arcadis Gen products use often RAG statuses. Its therefor obvious to add these colours to the set too:
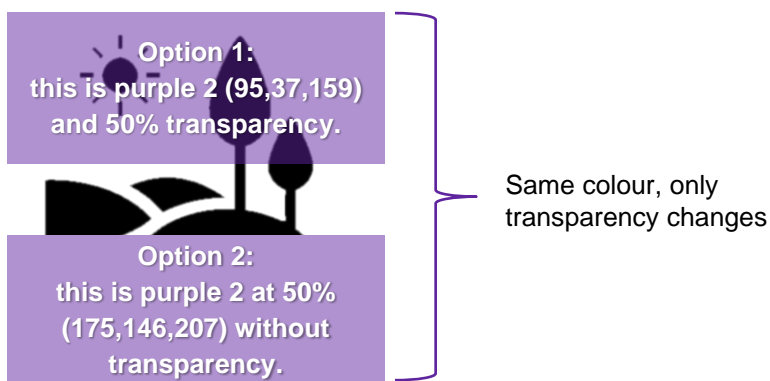
PANTONE   **PANTONE**
3517 C   2423 C
CMYK   **CMYK**
0/100/89/24   100/0/89/33
RGB (HEX)   **RGB (HEX)**
193/0/22(#C10016)   0/170/19 (#00AA13)

The table below can be used for easier copy& paste of the colour values.

GEN colors

		RGB	transparency with background **white**			transparency with background **black**			CMYK
		100%	25%	50%	75%	25%	50%	75%	100%
primary	purple 1	135,24,157 #87189D	165,82,182 #A552B6	195,139,206 #C38BCE	225,197,230 #E1C5E6	101, 18, 118 #651276	101, 18, 118 #651276	101, 18, 118 #220627	65/100/0/0
primary	orange	228,97,15 #E4610F	235,137,75 #EB894B	241,176,135 #F1B087	248,215,195 #F8D7C3	171,73,11 #AB490B	114,49,8 #723108	57,24,4 #391804	5/72/100/0
primary	black	0,0,0 #000000	64,64,64 #404040	127,127,127 #7F7F7F	191,191,191 #BFBFBF	0/0/0 #000000	0/0/0 #000000	0/0/0 #000000	0/0/0/100

		RGB	transparency with background **white**			transparency with background **black**			CMYK
		100%	25%	50%	75%	25%	50%	75%	100%
secondary	purple 2	95,37,159 #5F259F	135,92,183 #875CB7	175,146,207 #AF92CF	215,200,231 #D7C8E7	71, 28, 119 #471C77	48, 19, 80 #301350	24, 9, 40 #180928	77/97/0/0
secondary	blue	0,175,215 #00AFD7	64,195,225 #40C3E1	127,215,235 #7FD7EB	191,235,245 #BFEBF5	0, 131, 161 #0083A1	0, 88, 108 #00586C	0, 44, 54 #002C36	86/0/9/0
secondary	gray	85,87,90 #55575A	128,129,131 #808183	170,171,172 #AAABAC	213,213,214 #D5D5D6	65, 65, 67 #414143	44, 44, 45 #2C2C2D	22, 22, 23 #161617	61/51/46/40

		RGB	transparency with background **white**			transparency with background **black**			CMYK
		100%	25%	50%	75%	25%	50%	75%	100%
additional	red	193,0,22 #C10016	209,64,80 #D14050	224,127,138 #E07F8A	239,191,197 #EFBFC5	145, 0, 16 #910010	97, 0, 11 #61000B	48, 0, 6 #300006	0/100/89/24
additional	green	0,170,19 #00AA13	64,191,78 #40BF4E	127,212,137 #7FD489	191,234,196 #BFEAC4	0, 127, 14 #007F0E	0, 85, 10 #00550A	0, 43, 5 #002B05	100/0/89/33

When using the partially transparent colours, there are two options:

1. use the 100% colour <u>and</u> set the transparency of the colour,
2. don't use transparency and use the 75%, 50% or 25% colour.

Option 1:
this is purple 2 (95,37,159) and 50% transparency.

Option 2:
this is purple 2 at 50% (175,146,207) without transparency.

Same colour, only transparency changes

3.2   Logo & background

From the Brand style guidelines (*Arcadis_Gen_Brand_Guidelines_v6a.pdf*), this is how our logo should be shown:

Logo – How to use it

The logo must always be clearly represented with good levels of contrast. Where possible use the preferred option, but the white out version is recommended for use over the biosphere.

When used within text the name must appear as: Arcadis Gen.

Preferred first choice

Reverse choice

Mono version

White out version

Any variations in this are not allowed.

Logo – How not to use it

The colour and lock up of the logo mustn't be changed or distorted as shown.

11

7

3.3   Fonts

The Arcadis Gen brand uses, for digital copy, three fonts:

Amiko

- Amiko regular is used in all headlines

Overpass

- Overpass Light is for all body copy
- Overpass Bold for all call to actions (everything the user can click on that triggers an action)

Arial

For our day to day usage (e.g. news items in blogs) Arial is used.

- Arial regular for headlines and body copy
- Arial bold for subheads and small subheads

Headline
Subheader

Small subheader

Body copy cae aturerrorum raturios sum hilitem que niet doluptat undaniat. Liquos doluptur audandaest, nonsequiatem isim dolenitatem adit vitent venist ut aut que doles por si seri volupit voluptati dolo et ut eatur sanis et quate voloreprat ipsamet hilloribus expel exceruptam, omnistrum, conectem re ipsaerere quos sed mi, sunt.

3.4 Login screens

Arcadis Gen has several products that each require users to log in. These login screens all differ which will not help a user recognize our products. An overview:
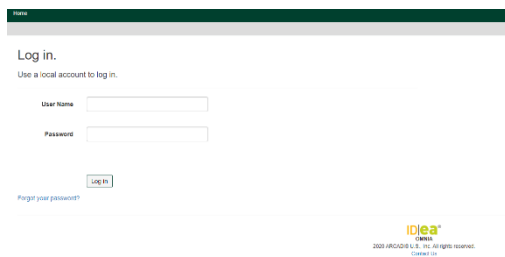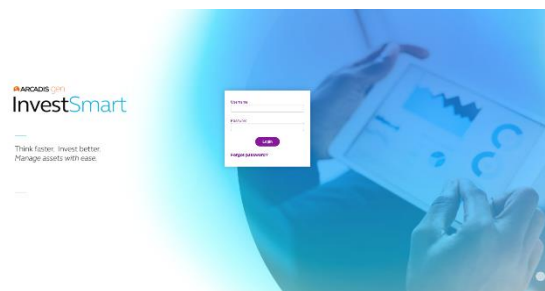
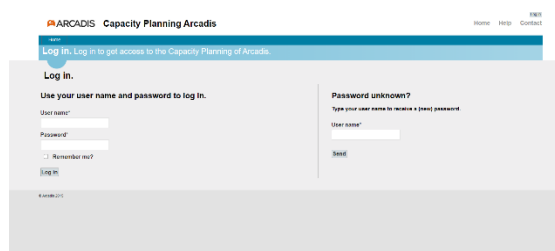ForeSight
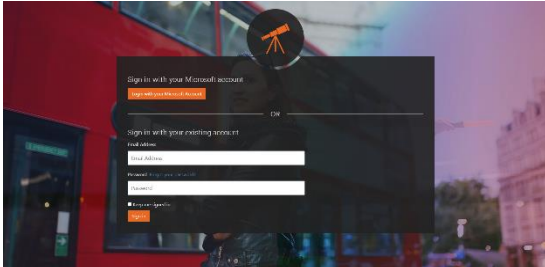
SmartDD
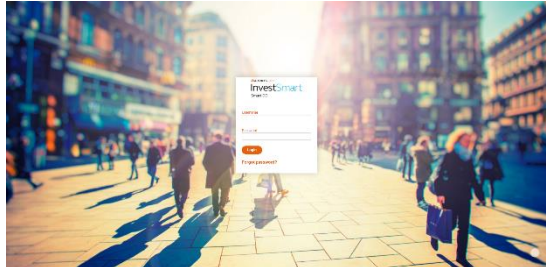
Omnia

InvestSmart

VidasNET

Capacity Planning

Next to this variety of login screens, some very similar ones exist too. This also will not help in recognition of our products:
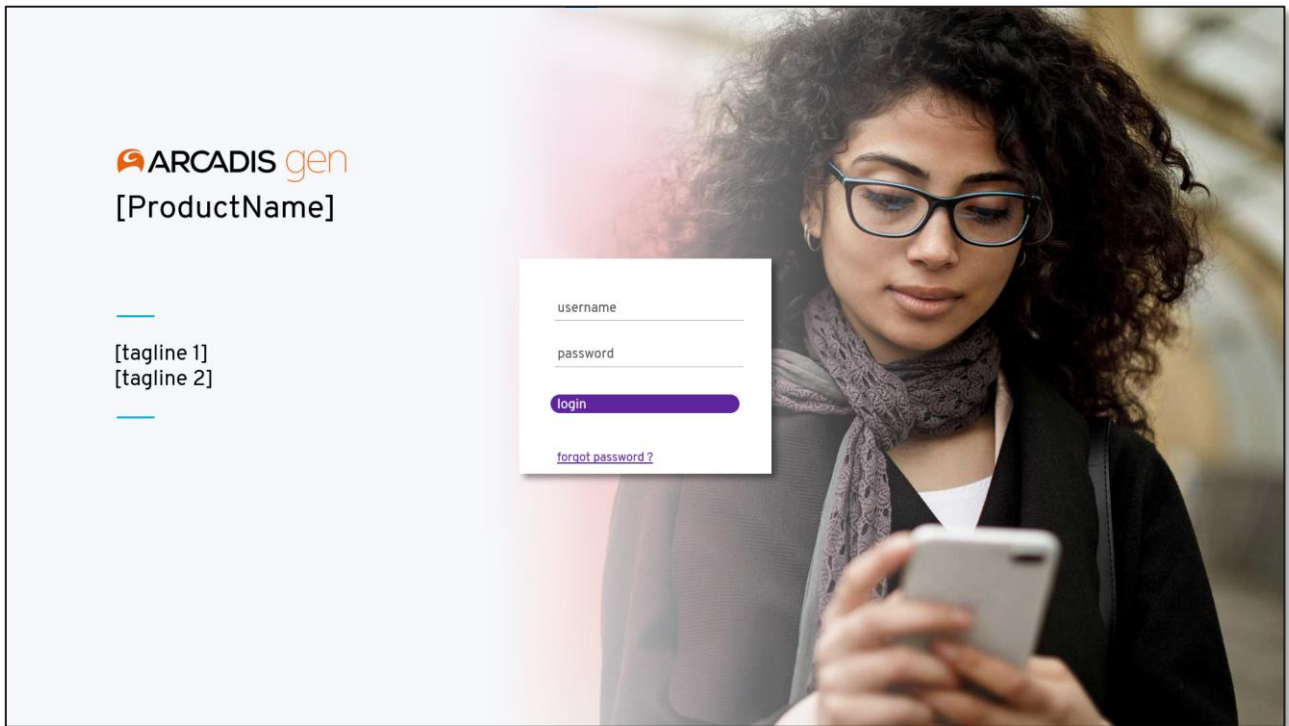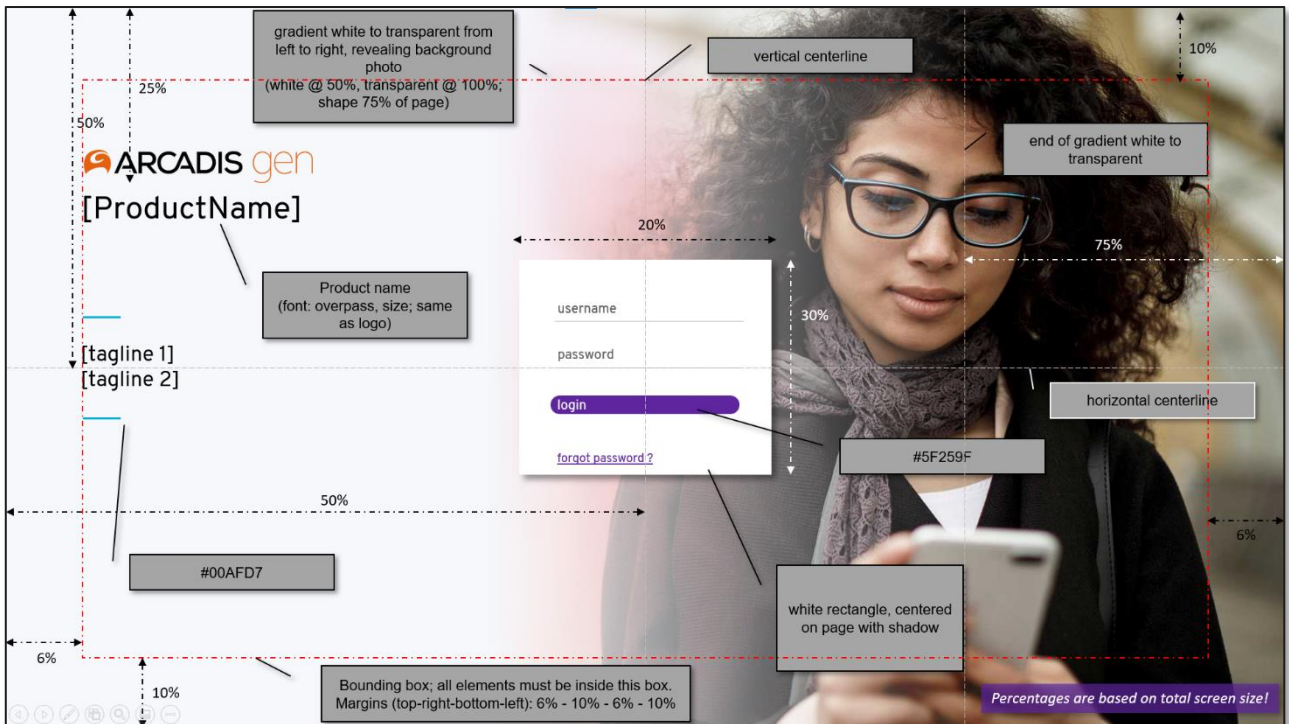
SmartDD

CAB request

In order to have consistency throughout our products, a recognizable login-screen is paramount. Such a screen can still allow for app-specific naming and branding. A suggestion:


The above screen follows some basic rules to have the same 'look & feel' of our products login pages. Relative sizes are shown here. Screen width and height are set as 100%.
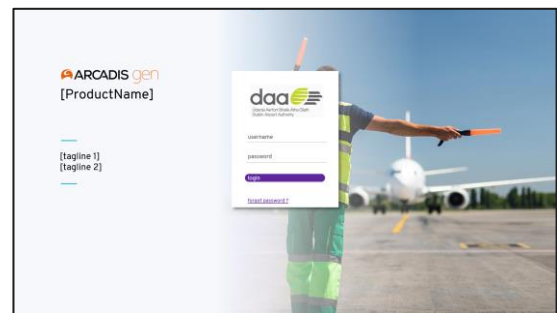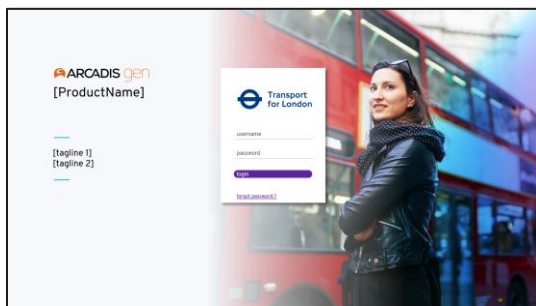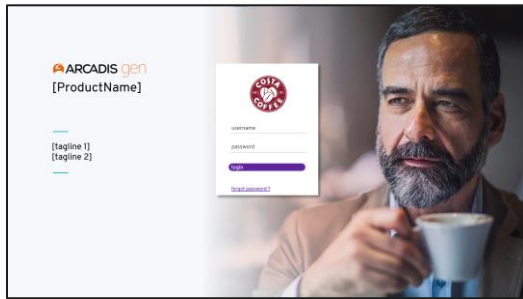
When including a client logo, this logo is to be placed centre-aligned above the log-in area at maximum 15% of the screen size.
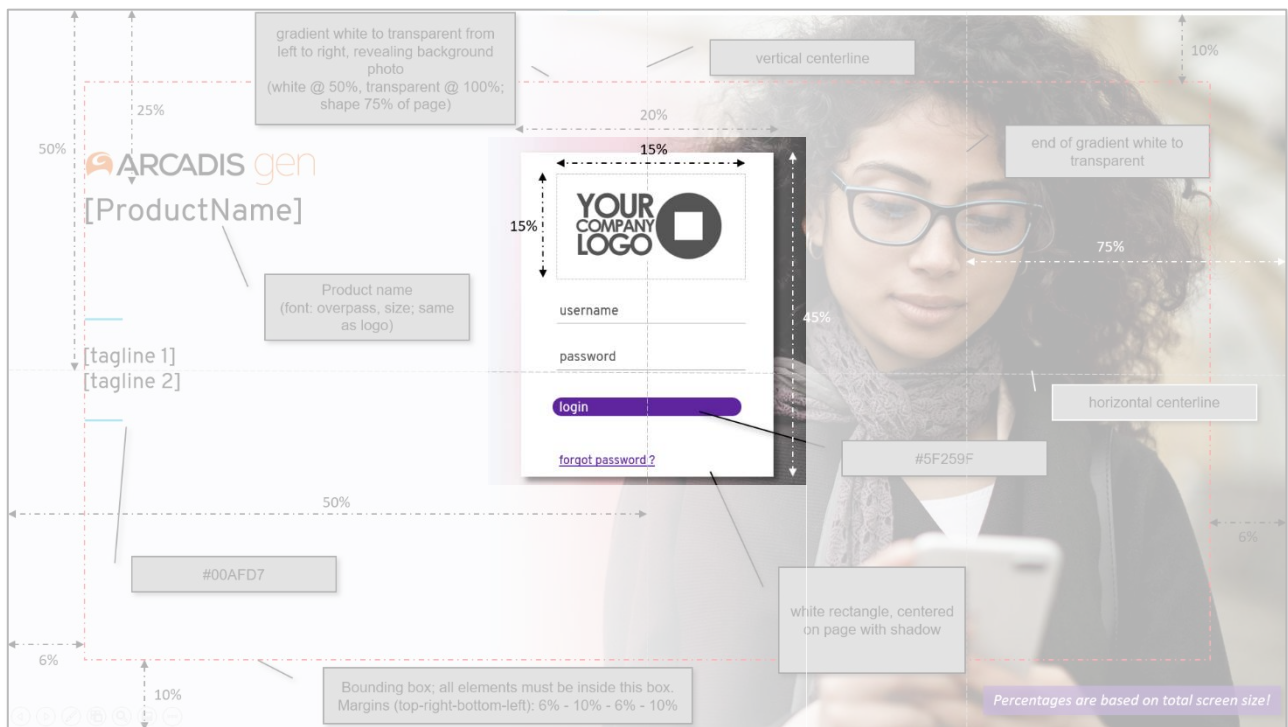
For the background photo's, a selection from eight brand-photos can be made, resulting in these login-screens:
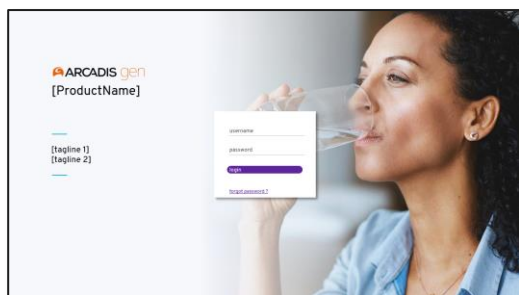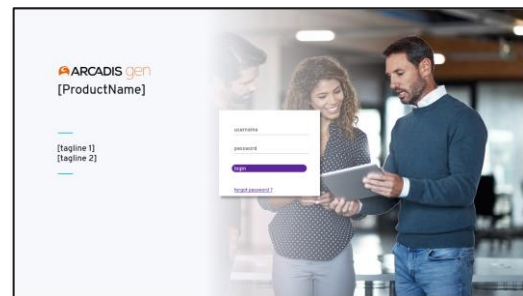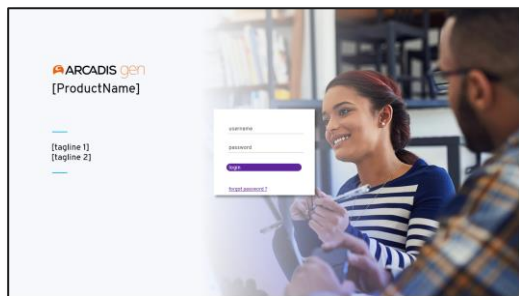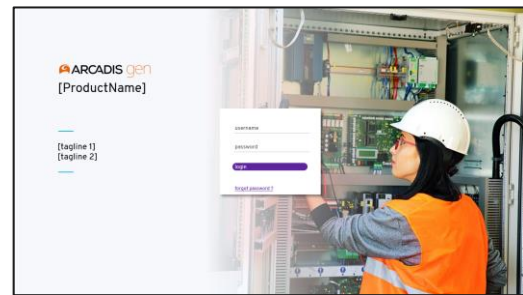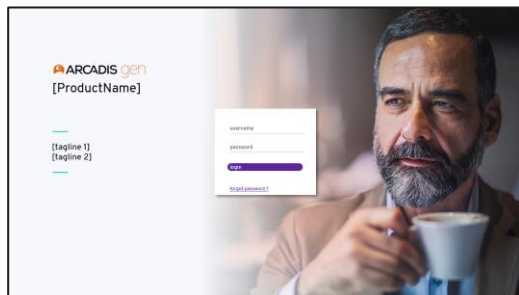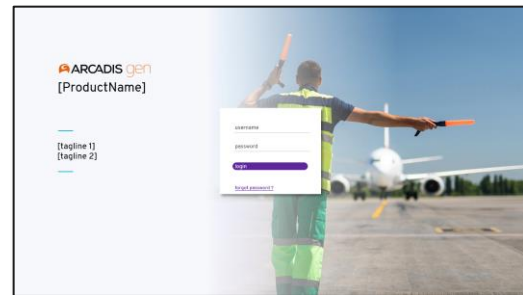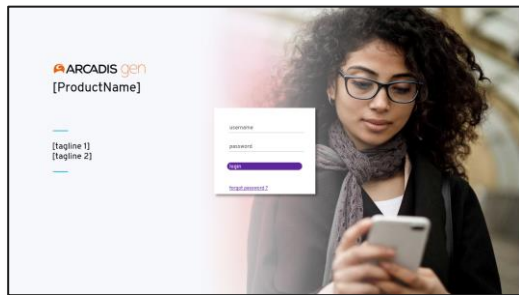
…and including the usage of a client brand logo:

3.5 Control element styles

Controls are (one of the) the building blocks of an application, whether this is a web-application, a window forms application, phone app or other. Controls typically serve the (visual) interaction between application and users. These come in many different formats and usages, like buttons, labels, text-entry fields, selection-boxes etc.

For a recognizable style in our Gen products, the following style overview act as a guideline. It is strongly suggested to adhere to this style. Technical limitations can force to deviate from this.

3.5.1 Buttons

- buttons are fully rounded to the left and right,
- text in buttons is aligned to the left, starting at the beginning (or end) of the radius of the left edge (see below),
- button version 1 uses Arcadis Gen secondary purple background (RGB 95,37,159),
- button version 2 uses Arcadis Gen secondary grey at 50% background (RGB 85,87,90),
- button version 1 and 2 uses white text in font Overpass, style bold,
- icons can be added to buttons in pure white.

3.5.2 Checkboxes

- common rule for checkboxes: always square (opposed to option boxes which are round)

3.5.3 Option boxes

- common rule for option boxes: always round (opposed to checkboxes which are square)

3.5.4 Text entry fields

- text input controls (textboxes) have a label on top of them,
- text input controls have only a single line below the text entry field (exception is a search bar),
- text input controls have a text-hint example (not the same text as the label but an actual text that could be entered, following the text-entry rules),

3.5.5 Dropdown boxes
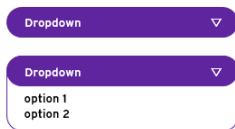
3.5.6 Progress bar

- progress bars show the percentage *in* the progress bar aligned to the left,
- when progress bar itself does not cover the percentage text, the percentage text is shown in black,
- when progress bar covers the percentage, the percentage text is shown in white.

3.5.7 Label

YOUR ADDRESS

- Font: Overpass

3.5.8 Slider

- when slider moves, the text on the right shows the current value,
- slider minimum, halfway and maximum values shown in regular,
- slider actual value shown in bold.

3.5.9 Fonts

Amiko
(headlines)

Headline

Overpass
(body copy)

Body copy lorum hilitem que niet doluptat undaniat. Liquos doluptur audandaest, nonsequiatem isim dolenitatem adit vitent venist ut aut que doles por si seri volupit voluptati dolo et ut eatur sanis et quate voloreprat ipsamet hilloribus expel exceruptam, omnistrum, conectem re ipsaerere quos sed mi, sunt.

3.5.10 Search field

```
Search for ...                                    🔍
```

3.5.11 Tables

Tables have been designed for maximum clarity and understanding. Yet, as Arcadis Gen is part of Arcadis, it also has references to the Arcadis brand style, like the square column titles.

ID	fist name	last name	amount	status	street	number	postal code	city	country
1001	Elon	Musk	€ 38,213	closed	BlockNine Street	1	1234AB	Arnhem	the Netherlands
1002	Boris	Johnson	€ 94,375	closed	Downing Street	10	SW1A 2AA	London	United Kingdom
1003	Joe	Biden	€ 33,214	open	Pennsylvania Avenue	1600	20500	Washington	United States of America
1004	Walt	Disney	€ 91,026	open	Hollywood Boulevard	6381	90028-6331	Los Angeles	United States of America
1005	John	Doe	€ 73,430	unknown	Franklin Street	70	SA 5000	Adelaide	Australia
1006	Arcadis Gen	London	€ 57,109	closed	Canada Square	25	E14 5LQ	London	United Kingdom
1007	Arcadis Gen	Arnhem	€ 56,976	closed	Beaulieustraat	22	6814 DV	Arnhem	the Netherlands

Row spacing 3x font

background columns in Secondary purple

ID	fist name	last name	amount	status	street	number	postal code	city	country
1001	Elon	Musk	€ 38,213	closed	BlockNine Street	1	1234AB	Arnhem	the Netherlands
1002	Boris	Johnson	€ 94,375	closed	Downing Street	10	SW1A 2AA	London	United Kingdom
1003	Joe	Biden	€ 33,214	open	Pennsylvania Avenue	1600	20500	Washington	United States of America
1004	Walt	Disney	€ 91,026	open	Hollywood Boulevard	6381	90028-6331	Los Angeles	United States of America
1005	John	Doe	€ 73,430	unknown	Franklin Street	70	SA 5000	Adelaide	Australia
1006	Arcadis Gen	London	€ 57,109	closed	Canada Square	25	E14 5LQ	London	United Kingdom
1007	Arcadis Gen	Arnhem	€ 56,976	closed	Beaulieustraat	22	6814 DV	Arnhem	the Netherlands

highlighted row to indicate selection
(secondary purple alpha: 0.1)

(note: all styles above are work in progress and as such subject to change)

4 User experience (UX)

Next to how our products look in terms of colours, fonts etc. it's important to be aware of how the user interface is organized. Questions like 'where to place a button?' or 'in what order to place option?' need to be answered. A well-organized interface ensures a user's flawless 'workflow' without any additional instruction. There are certain 'rules' to follow to maximize the user experience.

4.1 Active and disabled buttons

Buttons can be active or non-active (disabled). In order to clearly show the button's status, use the correct text colour.
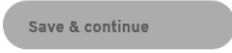
	❌ *not as good*	✅ *better*	
disabled button	Save & continue	Save & continue	Change the buttons text-colour to a colour close to its background colour, to emphasize the button is not active. Make sure the text remains readable.
enabled button	Save & continue	Save & continue	Instead of changing the button size, change the buttons back color and fore colour to emphasize the button is active.

4.2   Button text

Buttons always call for action, requiring the user to click on them to initiate a task. While some actions are very obvious, like a 'cancel' button cancels the initiated action, other might not be.

| ✅ not as good	✅ better	
End	Save & close	Do not use a to generic text like 'end' as it might not be clear what it ends. Be specific and short.
Current mode is A	Switch to mode B	Show what clicking will initiate, not what it has initiated.

As a guideline, the button-text should…

- explain what happens when it's clicked upon, e.g. always show the *next* step it initiates, not what it has initiated after the previous click,
- be clear, detailed and short in what it does.
- Avoid additional text like 'are you sure you want to delete this?' Just 'delete' with a yes and no button will suffice. It's short and to the point.
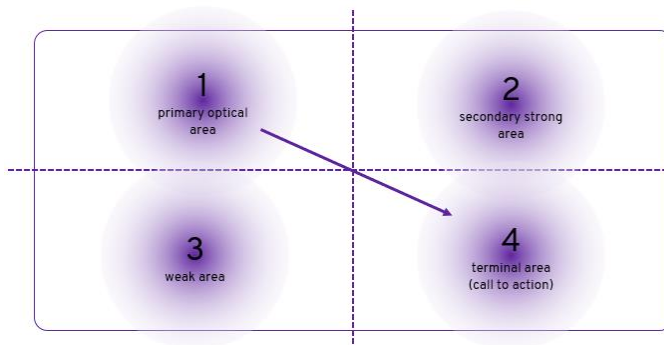
4.3   Button placement

Traditionally button placement is bottom right or 'primary right' as often called. This work well in most cases and seen in many instances, like the windows operation system itself.
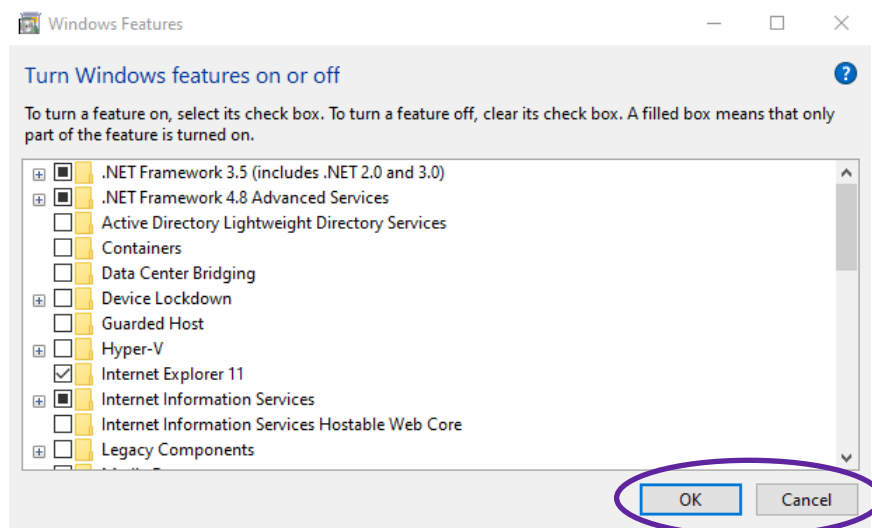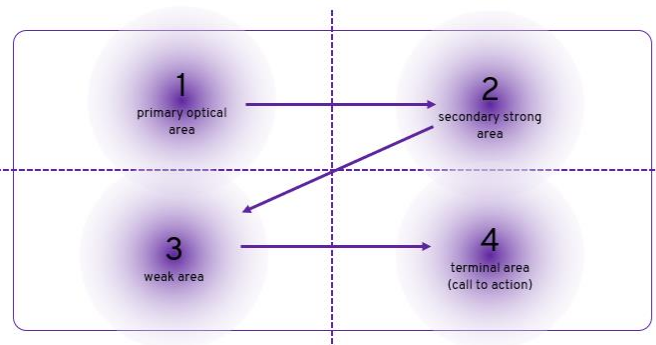
This placement follows the common attention flow, for western culture users, a Z-pattern that flows from top left to bottom right (extended Gutenberg diagram).

Note that the Z-pattern is particular applicable for users that have a genuine interest in the content of the page. Users (again western culture) that are in search for specific information often follow a F-shaped pattern (not discussed here).

Assuming that our customers have consciously chosen our products for a specific purpose we can also assume that our users are indeed interested in using our products. Hence the use of the Z-patterns is valid.

Still, deviations of the former can still be useful. For instance, when using a centre-aligned form, a central placed button would work very well:

Or, even left aligned buttons (primary left) can work too:


It always comes down to the reading pattern the user probably will follow and thus depends on the layout of the page or form.

4.4   Button recognition

A 'call to action' is a UI element that invites the user to interact with it, in order to initiate subsequent action(s). The most common form of such an element is a button. The most common form of interaction with this button is to click on the element.
For a good user experience, it's mandatory that the user (instantly) recognizes the 'call for action' element. The button has proven fit for this purpose, though this does not mean that everything, that can be interacted with, should be in the form of a button. But often it does help.

As an example, the following (sub-section) of a web-app:
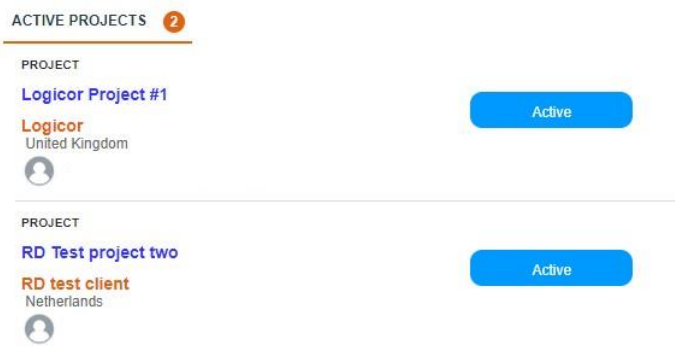
In the above example, the 'Active' 'button' isn't actually clickable, it's merely an indicator that the project (shown at the left in blue text) is active while the project name and the client name below, *are* clickable. This might not be the best implementation for an intuitive user experience.

The above can easily be improved though by modifying the 'call for action' elements and the use of icons:
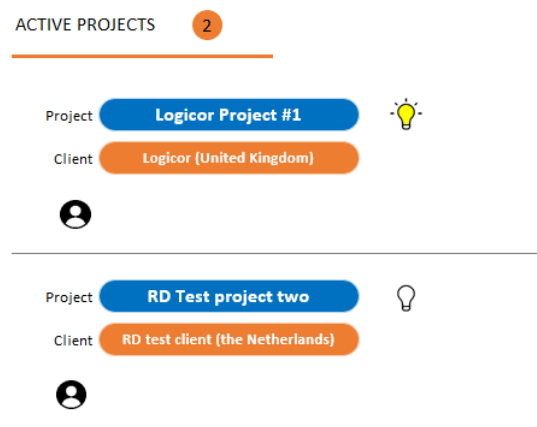
Now the actional elements are more recognizable (and match the style of the other clickable elements of this specific webapp). The active/inactive indicators convey the same message (bulb on is active project ,bulb off is inactive project). As a second benefit, the icon saves precious screen space.

4.5   Button order

If we consider our products are mainly targeted for left to right readers and we practice the 'place right' guideline (see chapter 'button placement'), the rule of thumb for button placement is: the main action button to the right, a cancel button to the left of the main action button and in case of a third button, this is place to the left of the cancel button.
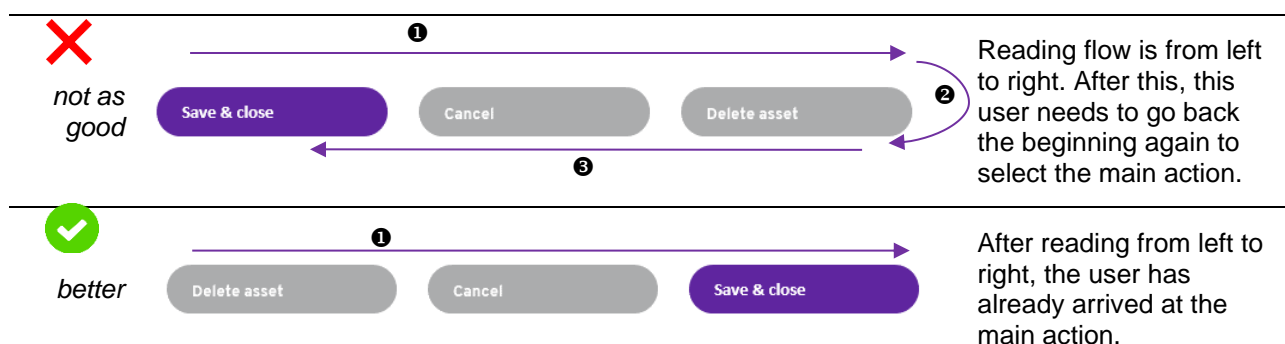
As an example, the user is editing an asset. After editing the user has the following choices:

DELETE ASSET	CANCEL	SAVE & CLOSE

Read from left to right, the user first sees 'delete asset, then 'cancel' and as last 'save & close'. Because 'save & close' is the main action, the most common action the user will perform when editing an asset, this action is at the end of the 'read'-line. The user does not have to go back to another part of the screen to find this main-action button.

not as good ❌

Reading flow is from left to right. After this, this user needs to go back the beginning again to select the main action.

better ✅

After reading from left to right, the user has already arrived at the main action.

As a rule of thumb, try to make the main-action button as easy as accessible as possible preventing additional going back to the left again.

4.5.1   Exception

The above might not work properly for in-line actions. As an example, the approval of timesheets of employees by a manager:

✅ *not as good*   Employee 1 → Reject → Approve

The first action the manager reads is to reject the emplo Which is commonly not the idea of timesheets.

✅ *better*   Employee 1 → Approve → Reject

The first action is the main action as, most of the time a r to approve the users timesheet.
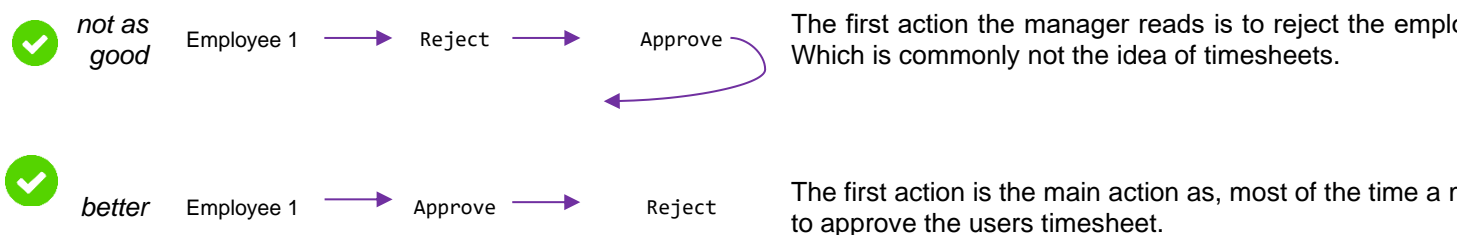
4.6   Are you sure? vs undo option

When the user is about to perform an important step, it's common practice to ask the user for confirmation to make sure the user understands what the button that was clicked means. There are two options for this:
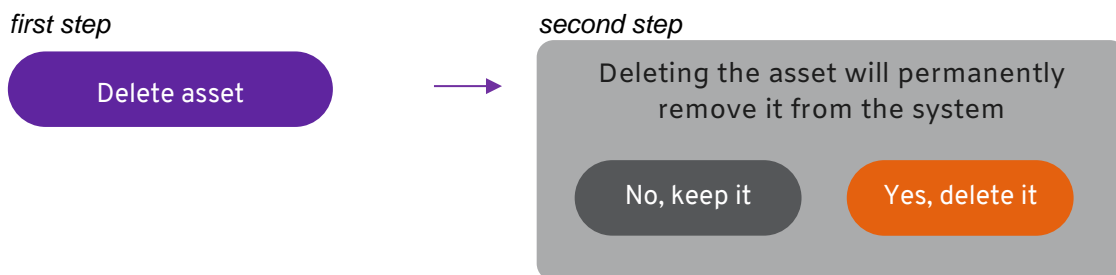
1. Show a confirmation dialog prior to the execution of the action
2. Show an undo option after the execution of the action.

When to use the confirmation and when to add an undo option afterwards? The answer is quite simple: when the action is truly irreversible, like deleting an item with no option to restore it, the user should be asked in advance if they understand what is about to happen.
When the action *is* reversible, for example the action to move an item to the recycle in, the user can be presented with an undo option *after* the action.

The advantage of an undo option, when applicable, is the reduced need the user need to focus on what is presented on the screen (and one less click). When the user exactly knows what they're doing, there's no need to overcomplicate the process. The undo option can just be ignored by the user.

Example of confirmation *prior* to the actual action execution:

first step   *second step*

Delete asset   →   Deleting the asset will permanently remove it from the system

No, keep it   Yes, delete it

Example of undo option *after* the action execution:

first step   *second step (this could be a popup that disappears after a few seconds)*

Move asset to bin   →   Asset moved to bin.   ✖

Undo
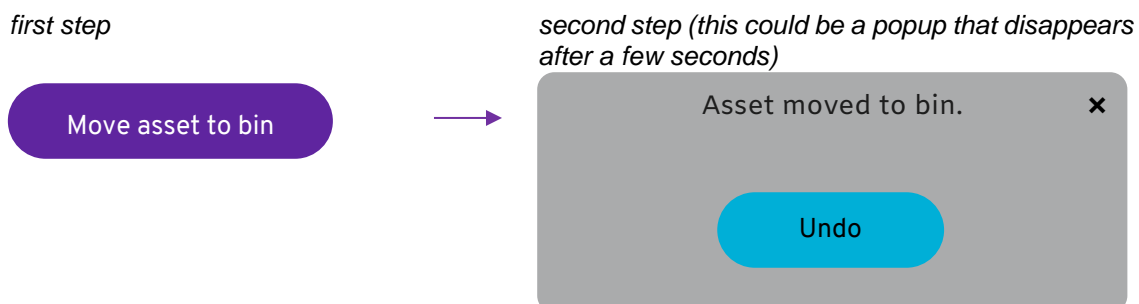
5   Graphs

In our daily work, products and solutions, we collect data. A lot of data. With this data we perform calculations, analysis and combine data with other data. The result of these actions are new insights, new conclusions, new relationships and,or new understandings. Graphs can help in these processes.

This requires a proper presentation of this data. Plain numerical data might not always provide a good understanding of the data. Hence, we use graphs and charts. For clarity, a chart is the overarching term to present data in the form of graphs, diagrams or table. As tables present data in their plain (alpha)numerical form, there is little to modify how this is presented. It's the diagrams and graphs that needs understanding on how to use them and when.

Graphs specifically refer to charts that plots data along two dimensions like x- and y-axes. The known values are plotted on the x-axis, the measured values on the y-axis.

It is important to keep graphs simple. Overloading the viewer with information will likely leave them with more confusion rather than insights.

5.1   Variations in graphs

The most common used types of graphs are:

- bars graphs,
- line graphs,
- pie charts,
- histogram,
- pictogram charts,
- area graphs (which is basically a filled line graph),
- spider graphs (or radar charts),
- flow graph,
- Gantt charts,
- scatter plots,
- tree diagrams,
- sunburst charts.

the use of the name 'chart' in the list above is basically incorrect as explained above but commonly accepted for this type of graph.

5.2   Line graph versus bar graph

To know when to use a line graph or when a bar graphs, one needs to understand the difference between continuous or discontinuous data.

Continuous data is measured and can represent any value on a continuous scale: height, weight and time are all examples of continuous data.

Discontinuous data is not measured but counted: numbers of employees in a company or cars in a traffic jam are examples of discontinuous data.

Line graphs can only be used for continuous data because there's a link between the data points on the x-axis (e.g. the next moment in time). An example:

The bar graph on the right is using discontinuous data on the x-axis. Product A has no continuous relation with product B and C.
The goal of the graph, having insight in sales over time is still visible as well as the difference between sales for each product.
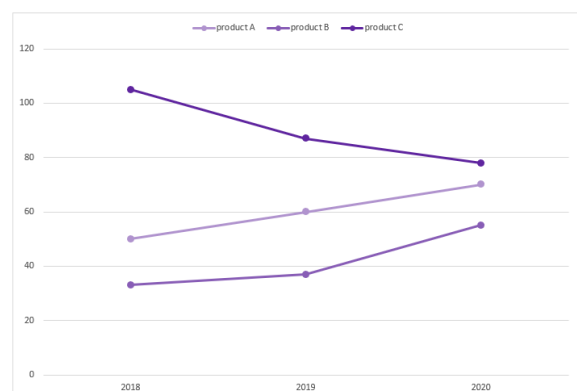
Hence a bar graph works best here.
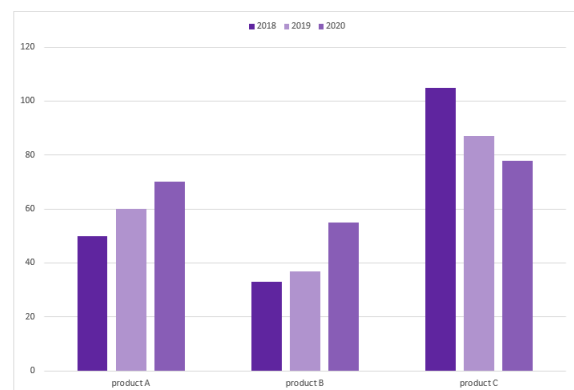
When using the time (years) on the x-axis, the line-graph can be used as there's a continuous relation between the years.
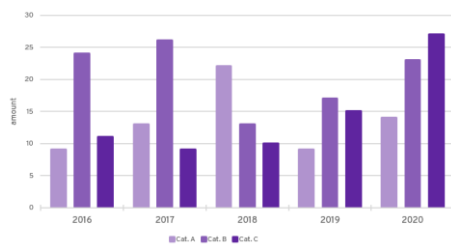
In this graph, the sales per product is a little more difficult to understand; the viewer needs to read the legend to understand which line represents which product.

The goal of the graph, to understand the trend in sales is still visible, just as is with the bar-graph.
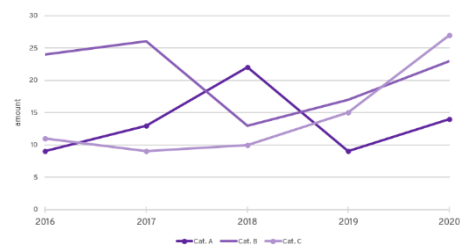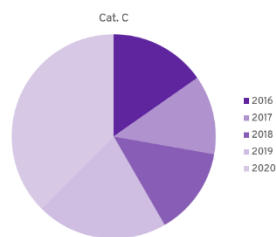
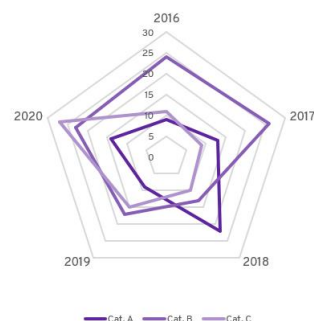|	
Bar graph	Bar graphs are useful when to compare quantities of different types of categories. They show discontinuous data on the x-axis.
Line graph	Line graphs are often used to display a trend in time. They plot continuous data on the x-axis.
Pie chart	Pie charts are used to compare values in relation to their summed total. For this to work, all measured values must be in the same unit of measurement. In such a chart, the difference between the actual values is more important than the actual values itself. When using a pi-chart, limit the number of segments to approximately six. Otherwise, the chart quickly renders unreadable.
Table	The use of a table is recommended when the exact numbers are important to show. And when multiple units of measure are involved.
Spider / radar chart	Spider or radar chart are often used to visually compare three or more variables. It allows to compare categories and at the same time discover trends in the data.

Table shown:

ID	fist name	last name	amount
1001	Elon	Musk	€ 22,489
1002	Boris	Johnson	€ 83,294
1003	Joe	Biden	€ 28,853
1004	Walt	Disney	€ 44,792
1005	John	Doe	€ 98,998
1006	Arcadis Gen	London	€ 41,003
1007	Arcadis Gen	Arnhem	€ 57,195
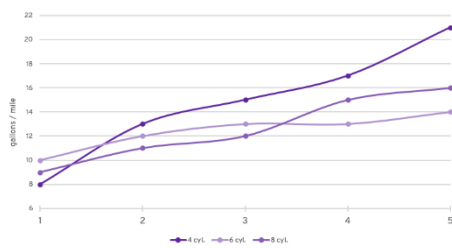
Scatter plot		Scatter plots (or scatter grams) have two axes, each representing a set of data. In this example the fuel efficiency compared to the number of cylinders. The graph shows the efficiency for 4-, 6- and 8-cylinder cars per gear.
Gantt chart		Gantt chart are very useful showing schedules. Each coloured bar varies in length, depending on the time some work takes.

For all tables, always use the Arcadis Gen defined brand colours. Variations in transparency/lightness are allowed. See paragraph 'Colours'.
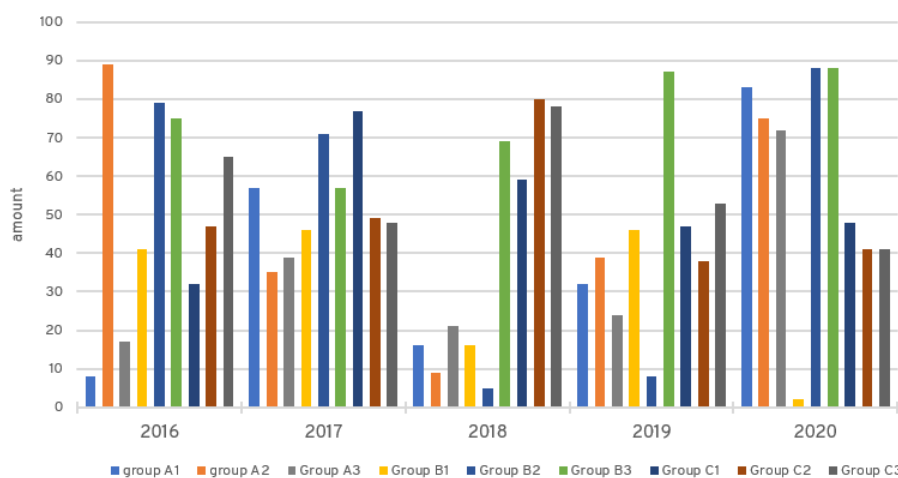
5.3   Graphical elements

When using graphs, always take into consideration the graphical elements that make up the graph. These graphical elements are:

- shape
- colour
- grayscale

- saturation
- size
- texture

- direction
- grain
- volume (3D)

- transparency
- focus
- shadow

The combination of different graphical element can enhance the readability of graphs.

For instance, when using more than 6 to 7 colours in a graph, most viewers find it difficult to separate all colours from each other. Making them mix up colours or 'forgetting' what a specific colour means. If required to display more than 6 to 7 entities, consider splitting these up into multiple graphs. Or try to combine groups and use another graphical element, like saturation.

Without use of multiple graphic elements: only colour is used. This makes interpreting the different groups A1 to C3 difficult (the colour for group A3 is similar to the colour of group C3).
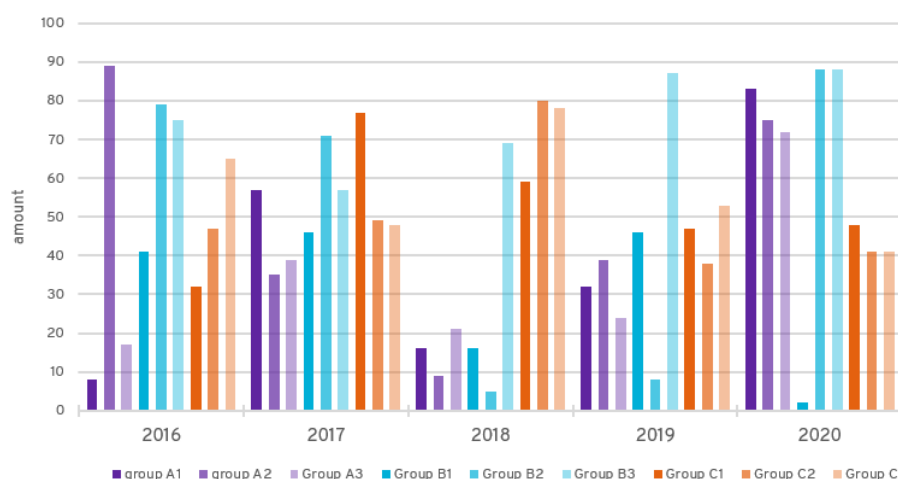
With the use of multiple graphical elements, in this case colour *and* transparency. Now the individual groups all have their own colour while the sub-groups A to C vary in transparency.

[add topic about accessibility > contrast and patterns]

6 User accessibility

For most people our products are readily usable, without the need for any additional aids. Unfortunately, this might not be the case for some users having impairments to their:

- Vision – people who are blind, partially sighted or colour blind,
- Hearing – deaf or hard of hearing,
- Mobility – people having difficulty moving the mouse or using the keyboard,
- Thinking and understanding – dyslexia, autism or learning difficulties.

There are guidelines set up how to maximize the usability for these people; the Web Content Accessibility Guidelines, or WCAG.

Link to the website: https:,,www.gov.uk,service-manual,helping-people-to-use-your-service,understanding-wcag.

The WCAG is based on 4 design principles of which each consist of multiple guides:

Perceivable	Operable	Understandable	Robust
provide text alternatives ('alt text') for non-text content	make sure everything works for keyboard-only users	use plain English	use valid HTML so user agents, including assistive technologies, can accurately interpret and parse content
provide transcripts for audio and video	let people play, pause and stop any moving content	keep sentences short	
provide captions for video	not use blinking or flashing content - or let the user disable animations	not use words and phrases that people won't recognise - or provide an explanation if you can't avoid it	make sure your code lets assistive technologies know what every user interface component is for, what state it's currently in and if it changes
make sure content is structured logically and can be navigated and read by a screen reader - this also helps if stylesheets are disabled	provide a 'skip to content' link use descriptive titles for pages and frames	explain all abbreviations and acronyms, unless they are well known and in common use - for example UK, EU, VAT	
use the proper mark-up for every feature (for example, forms and data tables), so the relationships between content are defined properly	make sure users can move through content in a way that makes sense	make it clear what language the content is written in, and indicate if this changes	make sure important status messages or modal dialogs are marked up in a way that informs user of their presence and purpose, and lets them interact with them using their assistive technology
not use colour as the only way to explain or distinguish something	use descriptive links so users know where a link will take them, or what downloadable linked content is	make sure features look consistent and behave in predictable ways	let the user return to what they were doing after they've interacted with the status message or modal input
use text colours that show up clearly against the background colour	use meaningful headings and labels, making sure that any accessible labels match or closely resemble the label you're using in the interface	make sure all form fields have visible and meaningful labels - and that they're marked up properly	
make sure every feature can be used when text size is increased by 200% and that content reflows to a single column when it's increased by 400%	make it easy for keyboard users to see the item their keyboard or assistive technology is currently focused on - this is known as 'active focus'	make it easy for people to identify and correct errors in forms - you can find best practice for form design in the GOV.UK Design System	
not use images of text			
make sure your service is responsive - for example to the user's device, page orientation and font size they like to use	only use things like mouse events or dynamic interactions (like swiping or pinching) when they're strictly necessary - or let the user disable them and interact with the interface in a different way		
make sure your service works well with assistive technologies - for example, important messages are marked up in a way that the screen readers know they're important	make it easy for users to disable and change shortcut keys		

The next paragraphs explain some best practices for these 4 main design principles.

6.1 Perceivable

Information and user interface components must be presentable to users in ways they can perceive.

6.1.1 Testing for perceivability

To test an online app or website for its perceivability, one of the aspects is that text should not be represented as images. To test this, turn off the displaying of images.

In Google Chrome this can be done by
1. click on the three dots in the top-right corner of the browser,
2. select 'settings',
3. navigate to 'Privacy and security' section and select 'Site settings',
4. scroll to 'Content' section and select 'Image',
5. click on this setting and disable the 'Show all (recommended)' option (changing it into 'do not show any images').

Once images are not shown, a placeholder-text is often shown, depending on the browser used and website-settings. Make sure the placeholder, when shown, does not cover other important parts of the screen.
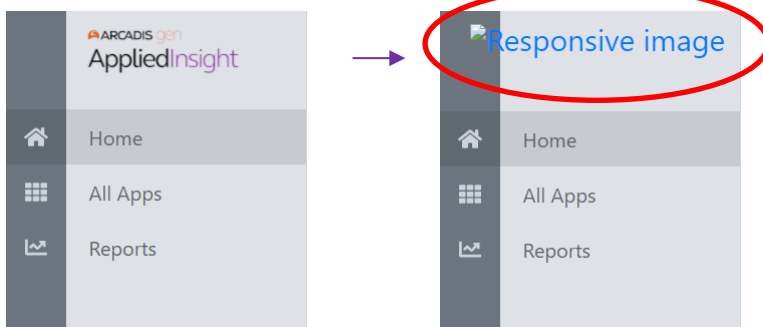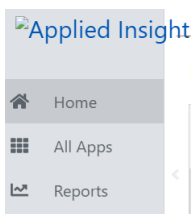

There are two options to cover for images that are not shown:

A
show a sensible alternative text

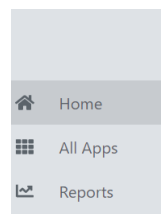to do this, use the 'alt' tag in the image-element. For example:

```
<img src=".\logo\brand-logo.jpg"
alt="Applied Insight">
```


B
do not show anything (including the 'broken-link' image)

add an empty alt-tag to the image-element.

```
<img src=".\logo\brand-logo.jpg" alt="">
```


Tips for a good alternative text

Describe the contents of the image but keep the text short. Keep in mind that the alt-tag text can have a maximum of 100 characters! Do not use keywords but a normal sentence. It's also not needed to include 'image of' or 'picture of' in the alternative text.

6.1.2 Line , character spacing

…

6.1.3 Colour-contrast

The WCAG 2,0 1.4.3 describes a minimum colour-contrast of 4.5 to 1 between foreground and background elements. Contrast range from 1:1 (no contrast) to 21:1 (maximum contrast; black text on white background).

An example:

	Save	Save	Save
background RGB	95,37,159	0,175,215	85,87,90
HSL	190,159,98	136,255,108	153,7,88
Foreground RGB	255,255,255	255,255,255	255,255,255
HSL	170,0,255	170,0,255	170,0,255
contrast	9.3 : 1	2.6 : 1	7.2 : 1
	✔ accepted	✔ not accepted	✔ accepted

For more information: https:,,www.w3.org,TR,WCAG20,#contrast-ratiodef
Colour contrast ration calculator: https:,,www.msfw.com,Services,ContrastRatioCalculator

6.1.4 Use of colour as single means of conveying information

The use of only colour as the only indicator of a state is highly discouraged. As an example, the following (subsection) of a website is shown, with and without colour.

The coloured image clearly provide clues the 'upload files' step has completed by means of the green circle, line and background. Therefor the current step must be 'Modify' also indicated by the blue line and circle and the little triangle below the text.

In the 'uncoloured' right image the current status is only distinguishable by the triangle.
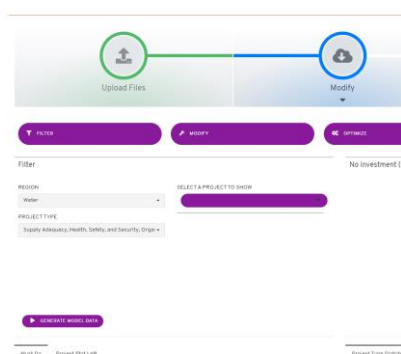
visual clues:
- green circle
- green line
- blue line
- blue circle
- triangle
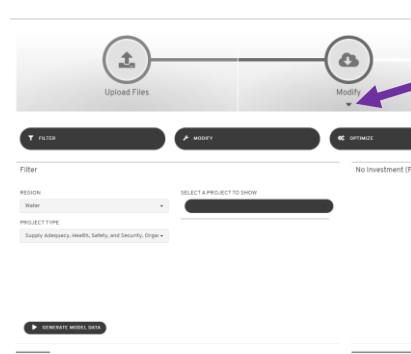
visual clue:
- triangle

6.2   Operable

User interface components and navigation must always be operable.

6.2.1   Alternative input controls

For all elements that a user can click or tap or drag or drop or slide or scroll, the user must also be able to navigate to that element and perform an equivalent action by using the keyboard.

This involves the user pressing the tab-key to step through each element to 'navigate' to. In the example below, the circles with numbers show the tab-stops and their order in which they're invoked.
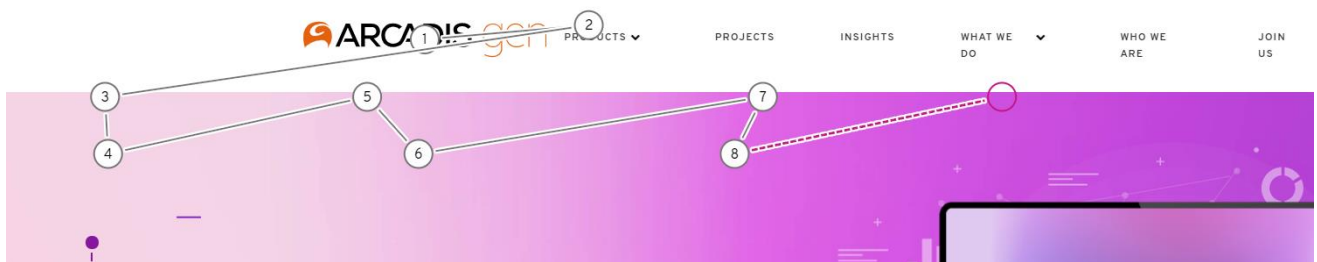
1. Logo
2. First menu item
3. Unknown
4. Unknown
5. Unknown
6. Unknown
7. Unknown; is this the second menu item?
8. Unknown

Users that step through the site using the tab-key quickly would get lost from step 3 on, not understanding what that step would bring when pressing enter (as alternative for a mouse-click).

As an example, a better implementation:

1. Username field
2. Password field
3. 'remember my login' field
4. Login button
5. Link to business page
6. Link to about page
7. Link to report a problem

Each tab-step is in a logical order without any unclear steps in between. Next to typing the username and password it requires only pressing the tab-key 4 times to login.

6.2.2 Screen zoom, reflow of text

For visually impaired users, the option to zoom into a webpage can help. This require the website to be designed that it will reflow the text within the width of the browser-window. It will prevent the user from scrolling horizontally back and forth. When developing the website, test this up to 400%

6.2.3 On screen flashes

Do not design content in a way that is known to cause seizures or physical reactions. This includes flashes and animations.

- Maximize flashes shown on the screen to 3 per second
- Implement the option to disable animations that are triggered by interaction (unless it is essential to convey the information in question)

6.2.4 Redundant links

Many links in a web page might have both a textual and an iconic representation of the same link. And although they may appear visually as a single link, they might have separate, identical links. Assistive technologies, like screen-readers would read aloud this (identical) link twice, confusing some users.

An example: the AppliedInsight platform had a link in the logo in the top left corner of the screen, redirecting the user to the page with the home menu selected.
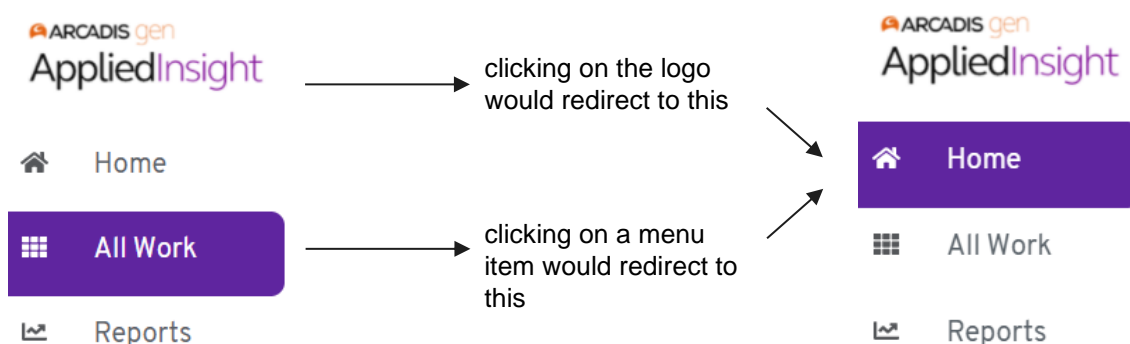
By removing the link in the logo, this will be clearer for users using screen readers, as only a single link brings them back to the home page.

6.2.5 Element without a label

Input elements, like text boxes, radio-buttons, or, as shown below, drag & drop areas, must have labels, indicating what the input element is for. When placing these elements in combination with images that explain the purpose of the element, this might be clear for the majority of the users. But visually impaired users might not be able to retrieve the explanatory message the image tries to convey.

It is therefore important to either add labels to input elements, or, when this is not an option, add aria-labels (see chapter 'Use of ARIA in HTML') to the source-code of the app or website. Such aria-labels tell screen assistive technologies what the element is for. Screen-readers (a form of screen assistive technology) will read out loud the text provided as label and as such verbally explain the purpose of the element.

Equally this applies to empty links. As an example, many websites show icons of social-media platforms they are active on. Clicking on these icons, brings the user to the companies' section of that social-media platform. Below an example of a part of the Arcadis Gen internet website:
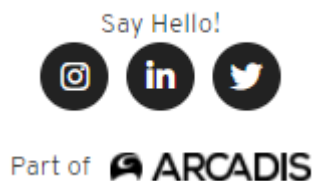
Though the social-media icons are clickable, they do not contain any text (the 'in' of the LinkedIn icon is an image, not text!). Screen-readers cannot process these icons and will skip them. Resulting in saying aloud:

'Say Hello. Part of'

or

'Say Hello. Part of Arcadis logo black'

(the Arcadis logo is an image too, with the alt-tag set to 'Arcadis logo black' and without having an aria-label)

6.2.6 Level headings

It is commonly known that structuring an app or website properly is important. Textual structuring is done by changing the text size. The technique for this is using <h1> to <h6> tags, where <h1> indicates the highest importance and <H6> the lowest.

Because headers and sub-headers will help visually impaired users to understand the website's structure, the screen-readers must be properly instructed.

This text

```
<h1>Introduction. …</h1>
<h4>Further….</h4>
<h6>Notes: …</h6>
```

Will not help users to grasp the structure even though the header tags are in numerical increasing order and properly closed. There's not relation between the <h1> and the <h4> and <h6> tag.

This would be a properly structured text:

```
<h1>Introduction
  <h2>
    <h3>
      <h4>Further…
        <h5>
          <h6>Notes: ….
          </h6>
        </h5>
      </h4>
    </h3>
  </h2>
<h/1>
```

Screen-readers will 'understand' the lower importance of the <h6> to it's 'parent' <h5> and <h4>. Which, in turn, the <h4> is known to be lower than <h3>, <h2> and <h1>.

6.3 Understandable

Information and the operation of user interface must be understandable. One of the aspects of this are abbreviations and acronyms. If used, and except they're well know, explain them. Well know and commonly used acronyms that need no explanation are for example UK, EU and VAT. Although VAT is typically used in English speaking countries and might not be understood by non-native English users.

6.3.1 Auto populate fields

…[ongoing work]

6.4   Robust

Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies.

6.5   Use of ARIA in HTML

When developing products for use on the internet that are used by humans, developers my consider the use of 'Accessible Rich Internet Applications' or ARIA-attributes.
For detailed information on how to implement ARIA, see this link: https:,,www.w3.org,TR,using-aria,

To investigate which elements in a browser are distinguished as navigation, presentation or other, the installation of the extension 'Accessibility Insights for Web' (Microsoft Edge) can be useful. Other browser brands will have similar extensions.

[ongoing work]

7 Changelog & approval

Version	Date	Changed by	Marketing approved by	Comments
0.1	2020-06-03	Raymond Delno		Initial setup of document
0.2	2020-06-05	Raymond Delno		Added chapters 'button text' and 'button order'. Updated login screen chapter
0.3	2020-06-15	Raymond Delno		Updated buttons to align with Gen branding. Added chapter 'control element style'
0.4	2020-06-18	Raymond Delno		Split document into UX and UI. Added UX high level text (work in progress)
0.5	2020-06-22	Raymond Delno		Added content for chapter 'button placement' , added sub-chapters for each 'control' including explanation text where applicable , replace login screen with mock-up
0.6	2020-06-23	Raymond Delno		Finished UX chapter , added user accessibility chapter
0.7	2020-06-30	Raymond Delno		Added chapter and placeholder for graphs styles
0.8	2020-07-05	Raymond Delno		Re-arranged chapter to be more logical (CX-UX-UI) , fixed typos , extended chapter graphs
0.9	2020-07-07	Raymond Delno		Improved login screen chapter
1.0	2020-07-09	Raymond Delno		Added red and green colour for RAG statuses
Version 1.0 is presented to the product owners at the PO call on 2020-09-07				
1.1	2020-07-23	Raymond Delno		Added the chapter 'line graph versus bar graph'
1.2	2020-10-27	Raymond Delno		Added chapter 'User Accessibility'
1.3	2020-10-28	Raymond Delno		Extended , re-arranged chapter 'User Accessibility'
1.4	2020-11-05	Raymond Delno		Added login-screen variant including client branding
1.5	2020-11-24	Raymond Delno		Added chapter 'Tables'
1.6	2020-12-01	Raymond Delno		Added margins to login-page design guide
1.7	2021-02-25	Raymond Delno		Extended colour chapter with all RGB and Hexadecimal values + note on transparency
1.8	2021-03-18	Raymond Delno		Expanded accessibility chapter
1.9	2021-04-30	Raymond Delno	Nicholas Hall	Added approval by marketing
2.0	2021-05-14	Raymond Delno	Nicholas Hall	Added chapter 'Logo & background' / changed document title
2.1	2021-05-17	Raymond Delno		Fixed some typos and made minor textual improvements