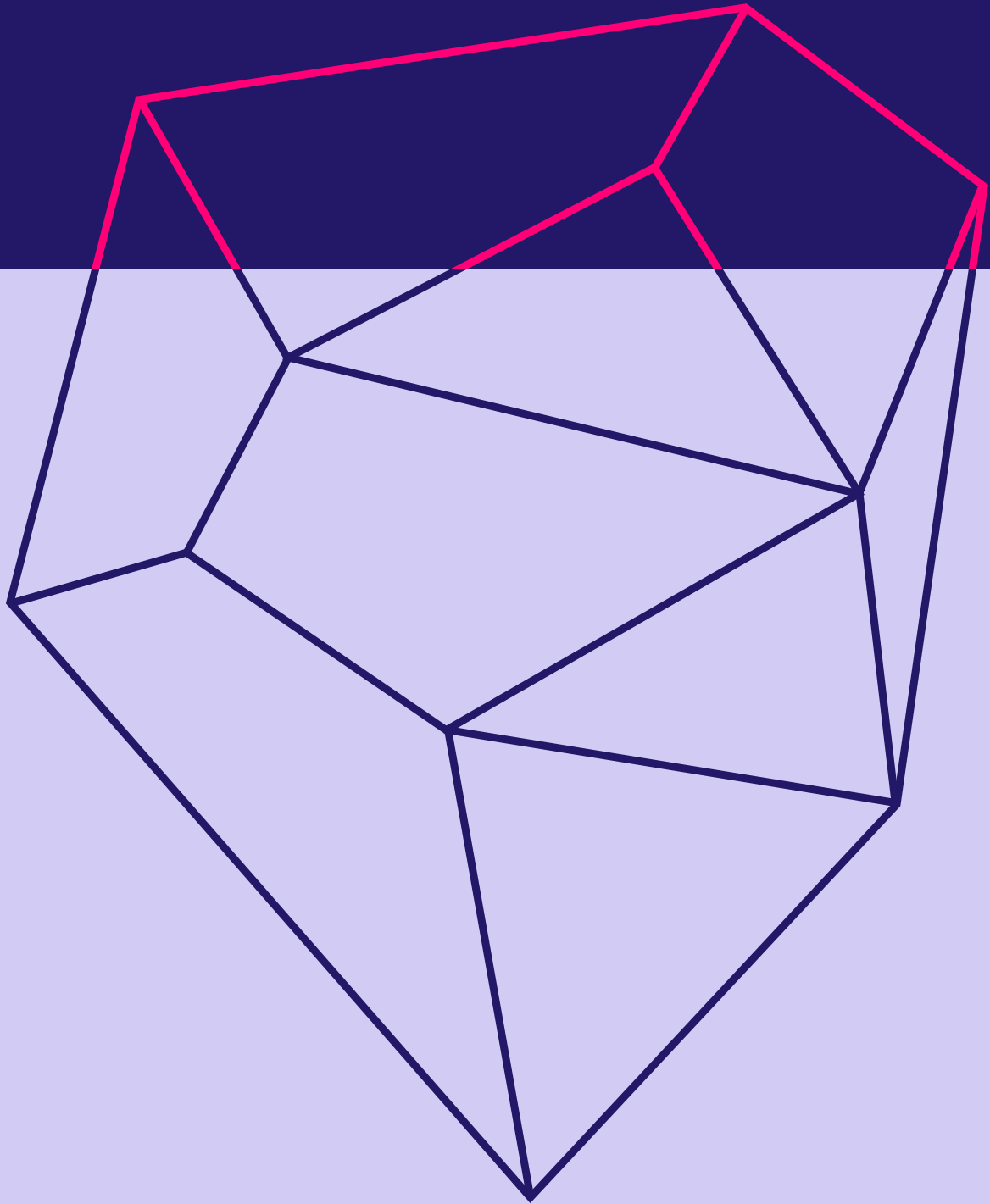


UX Guidelines

for Module Developers



Best Design Practices for a Successful User Experience

Summary

PrestaShop's Design Principles	3
Why this guide ?	4
1. Facilitate the first steps	5
Onboarding	
Login / Account creation	
2. Organize your interface	7
Menu and navigation	
Information architecture	
Tables	
Forms / Input fields	
3. Guide and help your users	12
Actions and buttons	
Contextual help / tooltips	
Confirmation popup	
Feedback	
Empty states	
4. Use the right words	16
Copywriting style	
PrestaShop glossary	
5. Consistency and standards	18
Fonts and Icons	
Text areas	

PrestaShop

Design Principles

Build for professionals

PrestaShop's final users are merchants. Interfaces, flows and interactions should serve their professional needs. Go straight to the point. Deliver help and information. User experience must be as efficient as possible. Users should achieve their goals without hesitation or difficulty.

Design for humans

Prestashop users are humans. Don't forget the emotional part of user experience. Prevent stress or frustration. Give feedbacks to any interaction. Use clear copy. Each problem should have a solution and a helpful alert message. Never forget the user's emotions.

Stay consistent

PrestaShop has different platforms and products and serves a huge community. Consistency is crucial to maintain user experience quality and stay scalable. Please respect our branding, style guides and standards.

Be universal

PrestaShop is international. Every human being, no matter the culture, language, connection or screen's quality, should be able to use our products and understand our interfaces. Avoid tone, expressions or interactions that are specific to only one kind of people. Be universal. Interfaces should be adaptive to any language, behavior or standard.

Why this guide?

This document is a set of good practices meant to help our contributors to create efficient and easy to use modules. It's also an effort to harmonize and improve the overall experience proposed by the PrestaShop ecosystem.

Considering the diversity of modules and contributors, building an exhaustive list of guidelines and cover all the use cases would have been a tough task. Hence, we decided to develop a certain number of points that are important to take in account to design user-friendly module's interfaces.

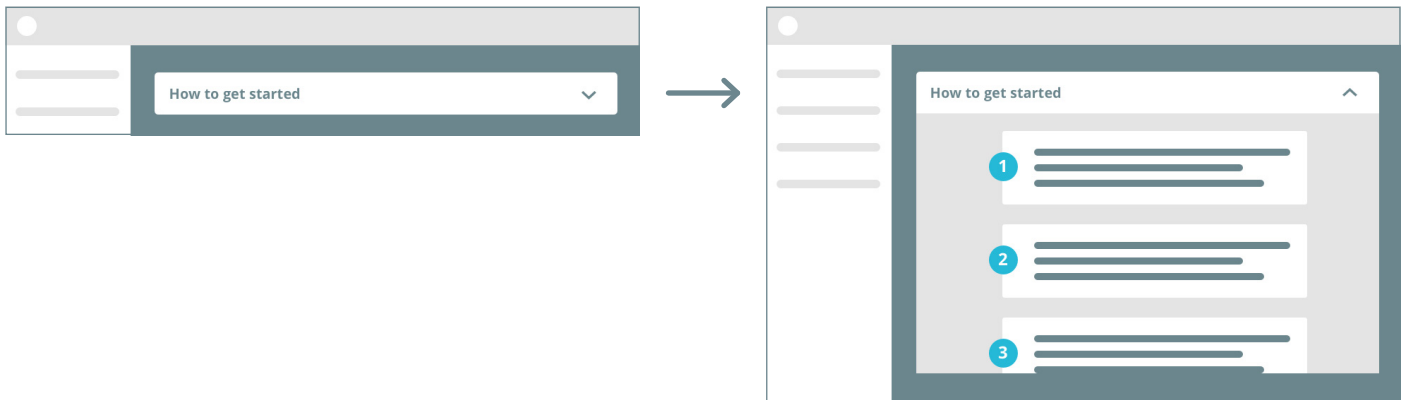
We need your help to continue to update these guidelines: don't hesitate to give us your feedback or submit your requests here: design@prestashop.com

Facilitate the first steps

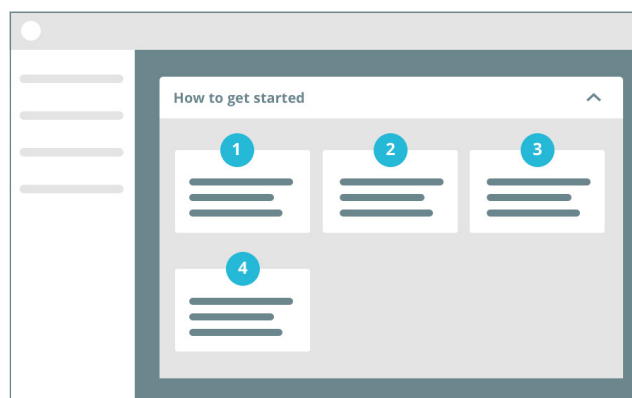
Onboarding

How to get started

Depending on the nature and complexity of your module, you may want to start with a short introduction to summarize the main steps / tasks to complete to use the module properly. This will help the user to understand quickly how it works and make the first approach easier.



There are different approaches to onboard your user, you could also use a video, or a popup:



Steppers

If the user has to go through several specific steps to configure the module, make sure that he is guided through these steps by using **steppers**.

Steppers can be horizontal or vertical depending on the number of steps, they display progress by dividing a sequence into multiple tasks.

Users should know what task they are completing and how many tasks they have got left to complete.



Login / Account creation

If your module requires a login or an account creation, be sure to make it the first visible task to complete. Keep the signup form as simple as possible, with the minimum number of fields.

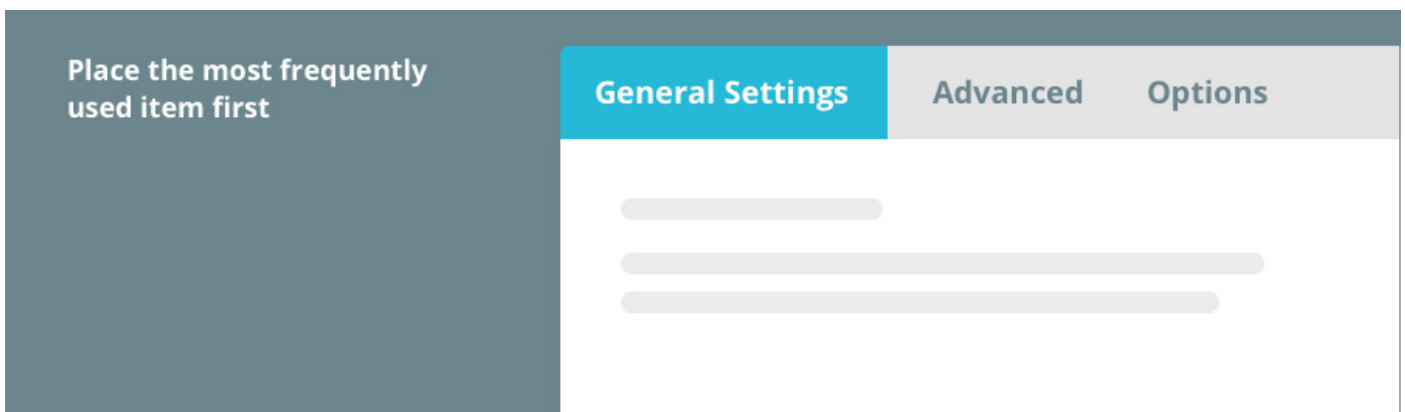
See the **Forms / Input fields section** for more best practices about designing forms.

2 Organize your interface

Menu and navigation

Menu ordering

If your module is meant to be used frequently, the most frequently used items should be placed first in the menu.



If your module is meant to be configured only one time before it runs on its own, the menu ordering should have a step-by-step approach in order to guide the user through the configuration process. You can consider numbering your menu items.

Have a step-by-step approach during configuration process

Step 1

Step 2

Step 3

Step 4

Menu items denomination should be as explicit as possible while remaining concise.

Menu design

Horizontal tabs on top of the page are the primary navigation items of your module's configuration interface.

If your menu has too many items to fit in the horizontal tabs, use a vertical menu.

Do not place your menu in a location that is unfamiliar to the user, and remember that a menu with too many items may be confusing:

try to minimize the number of items as much as possible.

Navigation

Users should always know where they are and how they got there.

Above a certain level of complexity, breadcrumbs are a good option to keep the user informed about his location within the module's hierarchy, just like on a website.

Information architecture

Try to limit the number of levels of information: the deeper a hierarchy becomes, the more likely users are to become disoriented.

Always give context to the user with pages that are properly titled and sub-titled. The optimal number of levels for typographic elements is 3: primary, secondary and tertiary.

Grid view & List view

Choose a layout that suits the type of content.



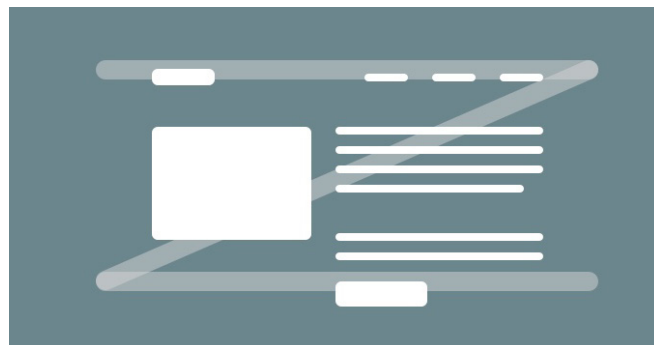
Prefer grid view for visual content. Ensure you display 5 or 7 images at the same time, on the same screen.



List view is appropriate for giving very detailed information.

Layout

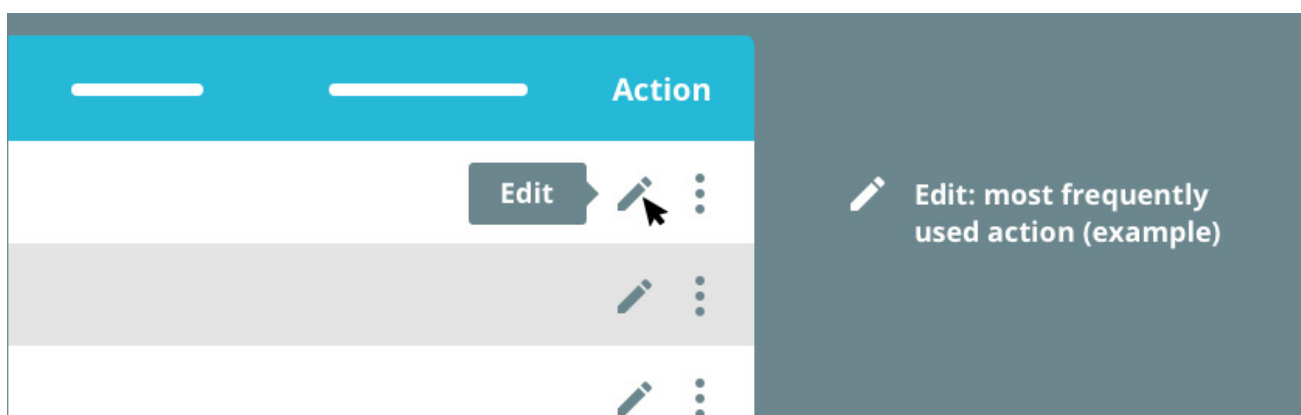
Information should be easy to scan. Try to use the F-shape or Z-shape to organize your interface. By giving structure to your content, users will find the needed information quickly.



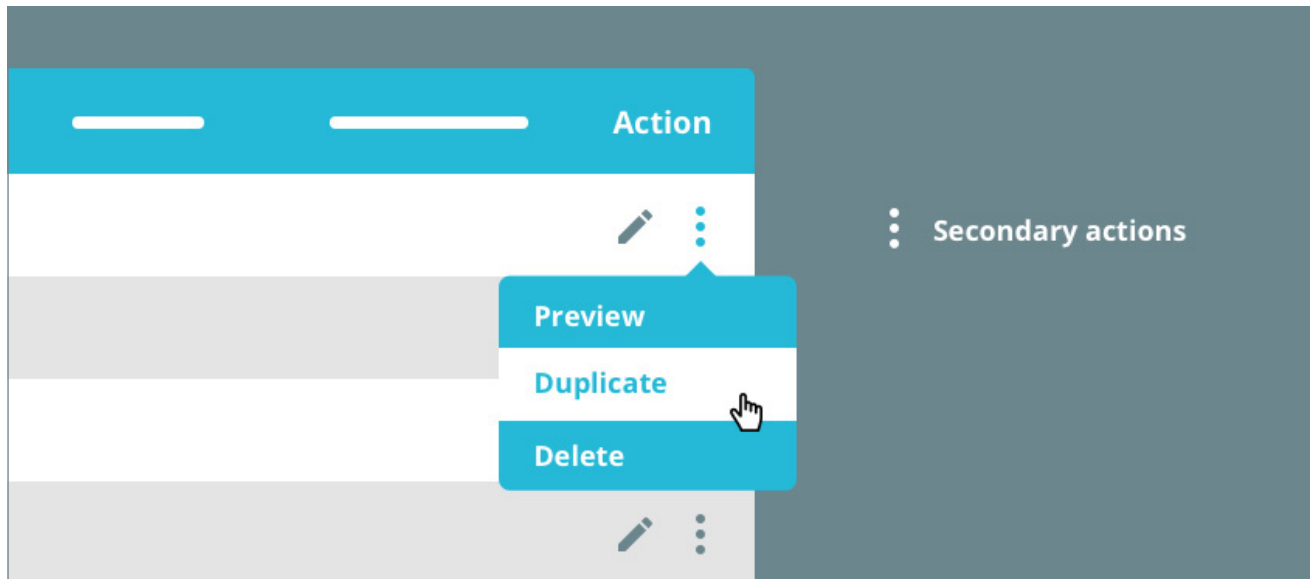
Group related informations together and use white space to create separations between different groups: white space will make your interface easier to read.

Tables

- Give the ability to filter any list of datas by making the **key columns** sortable.
- If your table allows more than 2 actions per line, identify the most frequently used action and make it an icon. Add a tooltip on hover to reduce ambiguity:



Gather the other actions under a single drop-down icon and name this column "Action"



- Provide a bulk-action functionality as it will help your users to finish their tasks quicker.

Forms / input fields

- Use short and accurate labels for each field. If the label is not explicit enough, add a Help tooltip.
- Use radio buttons when you have to display few options. Beyond 5 options, consider a dropdown list.
- Finish the form with a Validation button.
- Group together related fields to make the form more scannable.
- Generally, a form input field has 6 states: Default, Hover, Focus, Error, Success and Disabled.

3 Guide and help your users

Actions and buttons

Text on a button is always an imperative verb. It has to describe the action in the most concise and clear way.

Do not use more than 2 button sizes and 2 button colors. Your interface should have one contrast button color to differentiate the main and secondary actions.



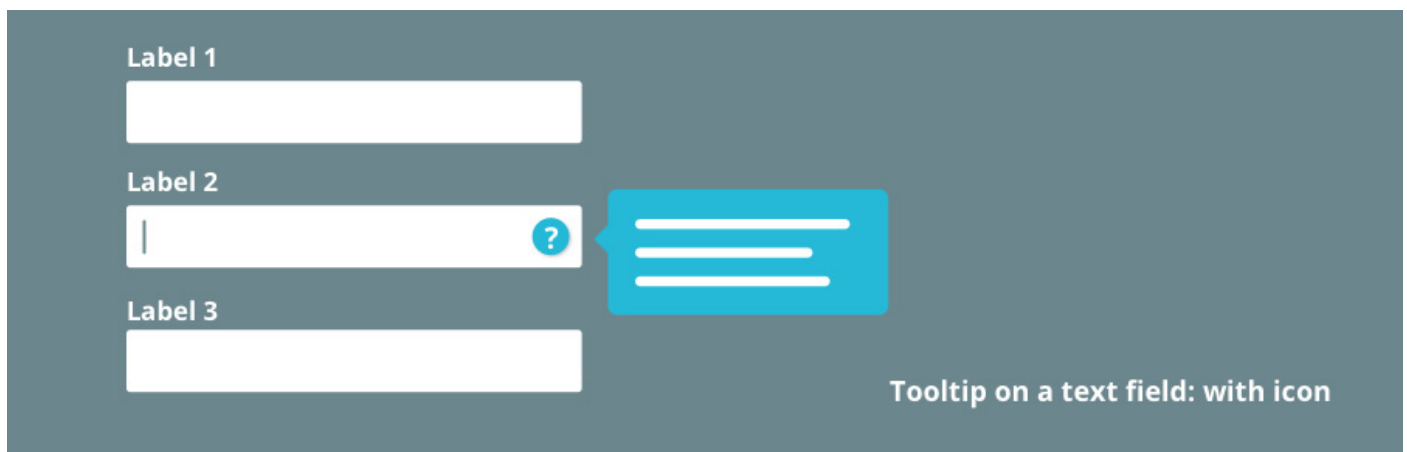
Primary action



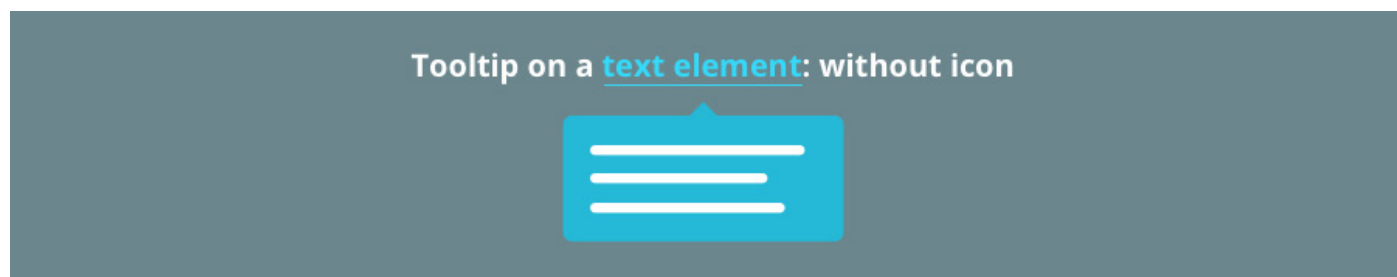
Secondary action

Contextual help / tooltips

In addition to the documentation you provide with your module, you can use contextual help to give the user the possibility to find answers to his questions at the moment he's more likely to need it, without having to leave his position in the interface.



Tooltips should be concise. Avoid writing long paragraphs as the users will be less likely to read it. Note that tooltips appear and disappear on hover, not on click. You can use them directly on a element or use an icon to make them more identifiable.



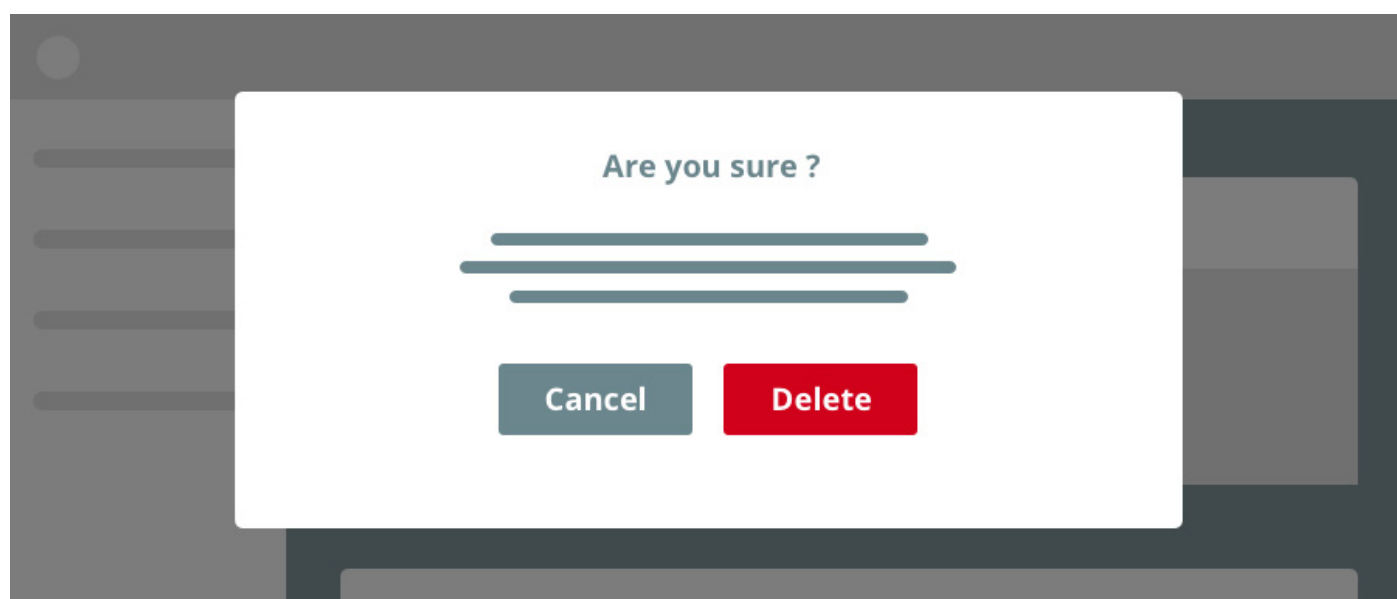
Try to identify the critical points, where the user is most likely to need information: too many tooltips on the same page may have the opposite effect and leave your user disoriented.

Confirmation popup

Prevent your users to accidentally make irreversible manipulations by using confirmation popups.

Keep in mind that popups are interruptive, they must be used only when users must confirm a **critical action**. Unnecessary confirmation popups are very annoying as they will only slow down your user in it's work.

Confirmation popups should inform user about the consequences of the action.



Note that a confirmation popup is followed by a dynamic feedback message to confirm that the action was successfully done.

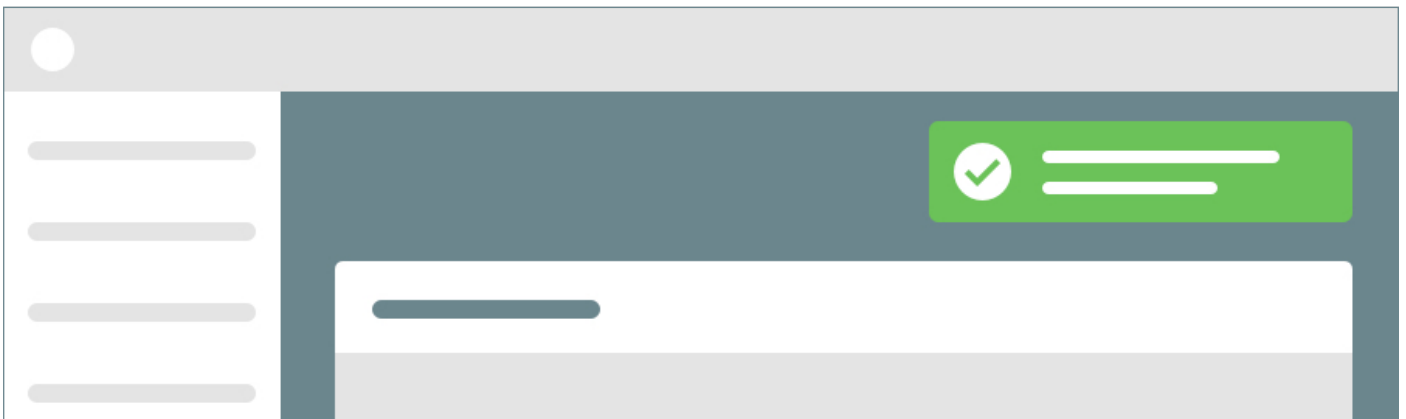
Feedback

Be sure to provide clear and contextual feedback so the user understands what went right or what went wrong.

A confirmation message should be given after the user successfully completed a task. An error message should give the information about what went wrong, and what to do to complete the task.

Feedback messages must be concise and go straight to the point.

Feedback messages are dynamic, they appear on the top right corner of the screen.

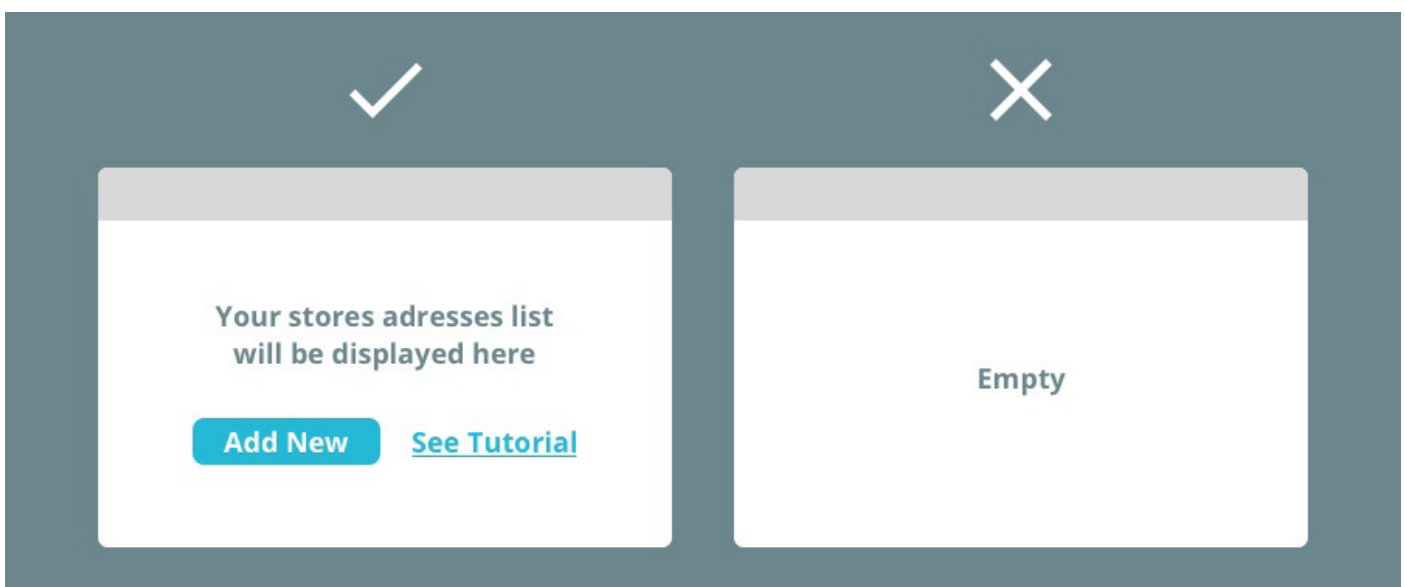


Empty states

Make sure not to leave your users in a stalemate. Instead, you can provide them clues to help them fill that screen.

- Give explanation to the merchant about what information he would find once that screen completed
- Enable user to fill that page by making available the appropriate function
- Provide more details about the process if necessary

Your empty states should never be empty.



4 Use the right words

Copywriting style

- Go straight to the point, be concise and use meaningful text.
- Consistency is key: ensure you use a specific label for the same function/action throughout your entire interface.
- Always stay positive and helpful, even in critical context.

PrestaShop glossary

Merchants are familiar with the denominations listed below. Ensure you use the suitable word in order to provide a consistent and fluid experience.

Add new ---	Act of extending something by creating a supplement
Configure	Set parameters
Disable	Turn off a function
Edit	Operate a change
Enable	Turn on a function
Reset	Restore the default settings
Save	Apply changes
Cancel	Abort an action
Duplicate	Create a copy
Preview	View a modification before it's saved
Filter by ---	Apply a specific filter on a list of items

Sort by ---	Generally used for numerical values with a meaningful order
Select all	Select all the items of a list
Upgrade	Move to the latest version
Generate	Create a specific file (eg: generate a .pdf, generate a .csv file)
Show all	Display a full list of items
Refresh	Update a list of items
Search	Look for a specific item
Bulk action	Apply an action to several selected items

5 Consistency and standards

Font and icons

Font

The font used in the PrestaShop back-office is [Open Sans](#). Keep the number of fonts to minimum.

Icons

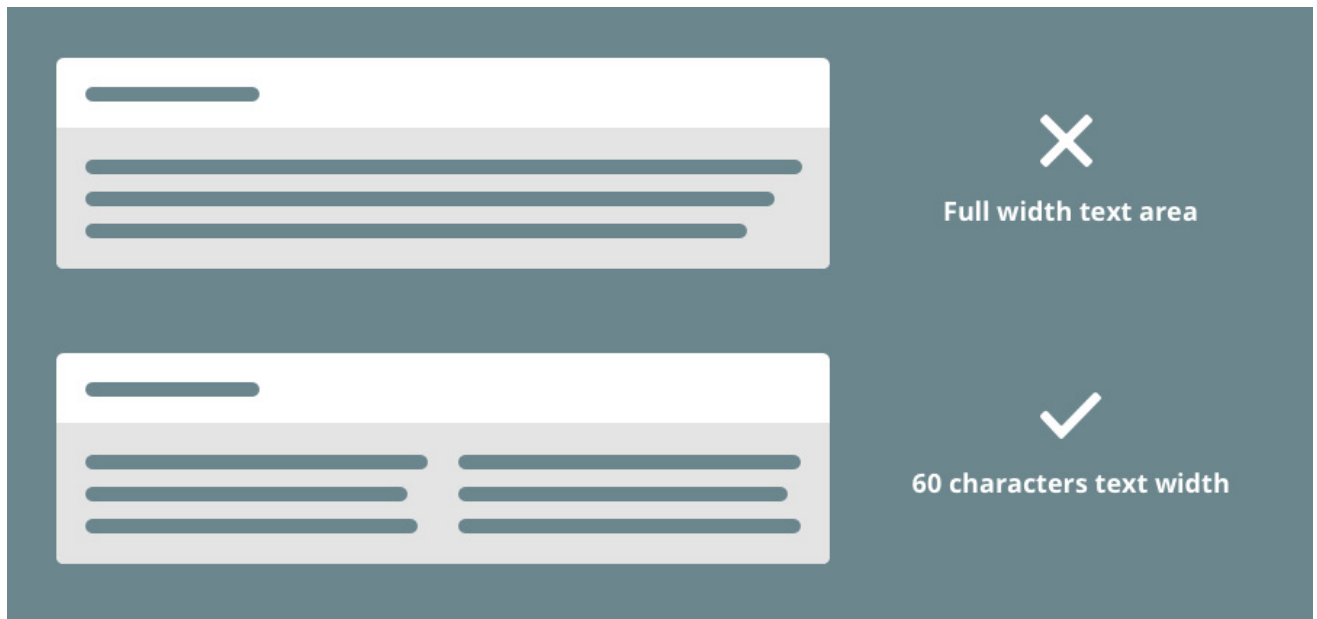
The icon font used in the PrestaShop back-office is [Google material icons](#).

Make sure to use the most universal possible icons.

Wherever possible, use labels with your icons to reduce ambiguity. Icons alone can be interpreted differently depending on the user's experience on other softwares and products.

Text areas

- Limit line length to approximately 60 characters per line, this is key to the readability of your text areas.



- Use comfortable spacing between lines: it is recommended to always set a line spacing 20% higher than the default one.
- Do not use too small character size. Below 12px, some users might have troubles to read your interface.
- Do not use all caps text
- Use proper contrast between font and background (don't hesitate to use a [color contrast checker](#) to see if your interface meets the WCAG standards).

Color choices

PrestaShop doesn't impose the use of the back-office color palette in the modules interfaces. However, the more your color choices will match the back-office environment, the more your users will enjoy a consistent and fluid experience.