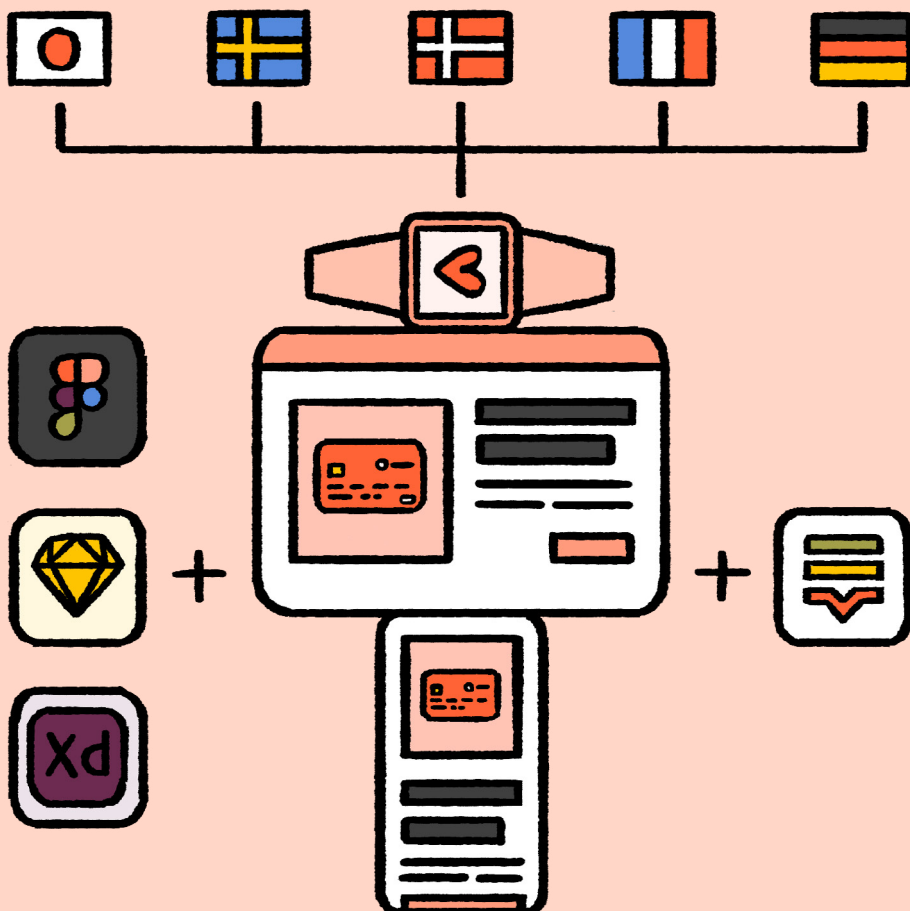


The complete guide to design-stage localization

“With design-stage localization, I’ve eliminated my operational workload and can now focus on performance and can now focus on performance to improve the overall experience for our customers.”

Romain Dahan
Product Manager



When you talk to fast-growing companies that are agile and rely on continuous development to some degree, you'll quickly realize that most haven't maintained processes and technologies to integrate localization into their product development workflow.

As a UX writer, designer, developer, product manager, marketer, or linguist, do you know how you can align as a team and build a unified localization workflow that starts at the design stage? And do you know how to implement it to build a better experience, improve your product for a global audience, and—ultimately—grow your business?

Design-stage localization can help with this. So, let's jump straight in.

We wrote this guide to give you:

- A complete introduction to design-stage localization
- Real examples of design-stage workflows
- A video guide to help you get set up
- A [checklist](#) to help you successfully implement localization at the design stage
- Real case studies and practical examples

This way, you can see for yourself how valuable design-stage localization is when it comes to continuously releasing localized products and growing your global presence.

What's Inside

1. Design-stage localization 101	4
2. Top design-stage challenges for localization teams	15
3. Design-stage case studies	19
4. A video guide to help you get setup	34

Chapter 1:

Design-stage localization 101

Design-stage localization is a powerful way to continuously release fully localized products like mobile apps, web apps, and games. It allows for the creation of designs suitable for multiple languages and bridges the gap between the designers, developers, content designers, and linguists working on localization.

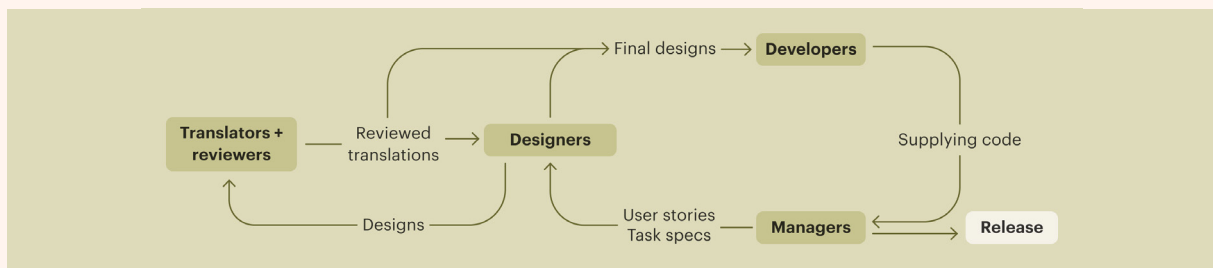
What is design-stage localization?

Design-stage localization entails starting the localization process along-side the concepting phase in product development. The process goes like this:

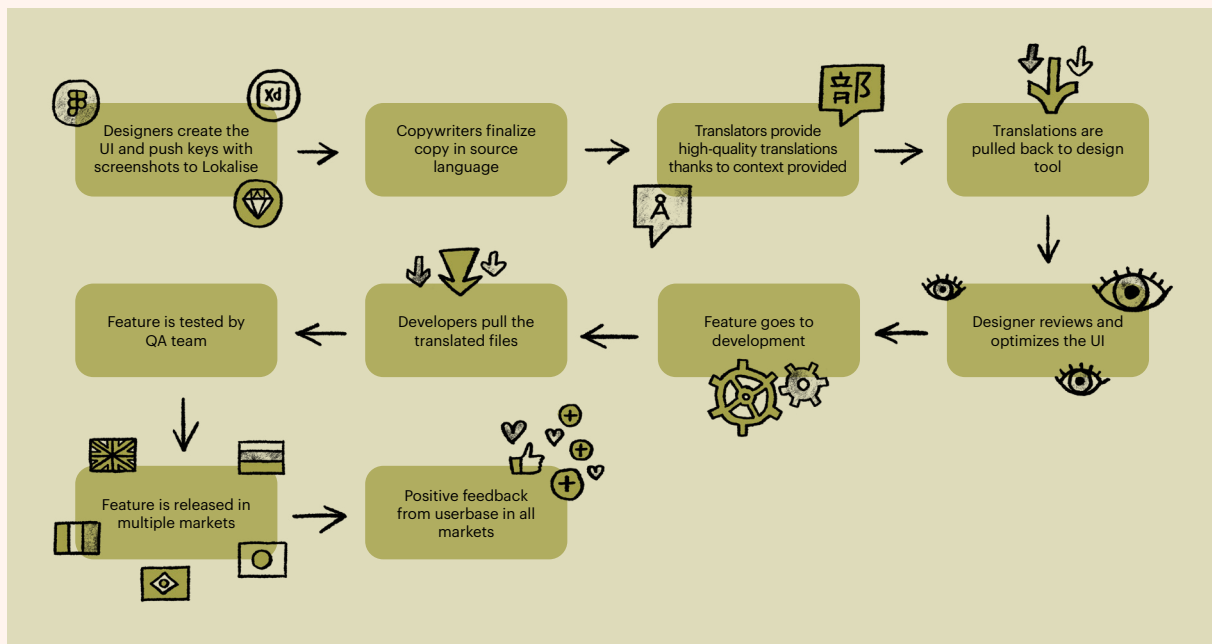
Designers start with a mockup, finalize the UI in the source language, and then linguists begin translating the UI before the feature goes to development. This means that fully localized prototypes are created ahead of time.

The most important thing to understand is that localization shifts from **post-release** (an afterthought) to **pre-release** (fully embedded in the product development process).

The easiest way to grasp design-stage localization is a continuous cycle with 8 key steps. Here's a high-level overview of the workflow:



Let's take a closer look at the steps involved in the above flow:



Design-stage localization is all about effective collaboration between your stakeholders before you build. If you're using Lokalise, you can use the dedicated plugins for [Figma, Sketch, or Adobe XD](#) so your designers, developers, and linguists can work with visual context and have overall coherence in the product experience.

In essence, the full scale of design-stage localization embeds localization at the beginning of your product development workflow to **streamline key naming, file handling, and error management.**

Types of design-stage workflows

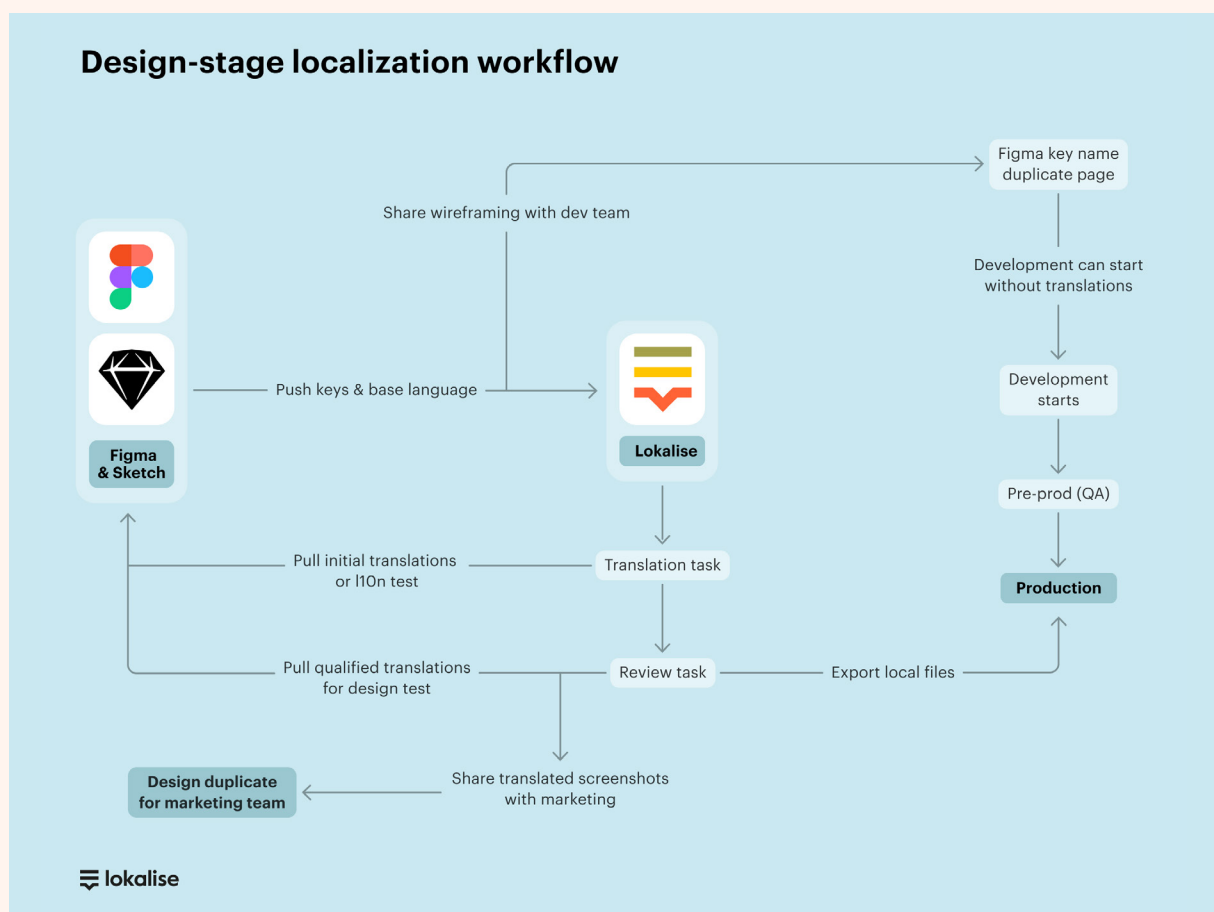
The good news is that design-stage localization isn't a one-size-fits-all approach. It's flexible and works fluidly with your existing tech stack, workflows, and use cases.

Advanced workflow using Figma/Sketch & Lokalise

Let's look at a more advanced workflow that allows product teams to accelerate the design-stage localization process further.

Using Figma/Sketch and Lokalise, localization teams can design mockups suitable for multiple languages **without** translations, and development can begin in parallel to translations.

Here's an overview of how it works:



Now, let's take a closer look at what this looks like in practice:

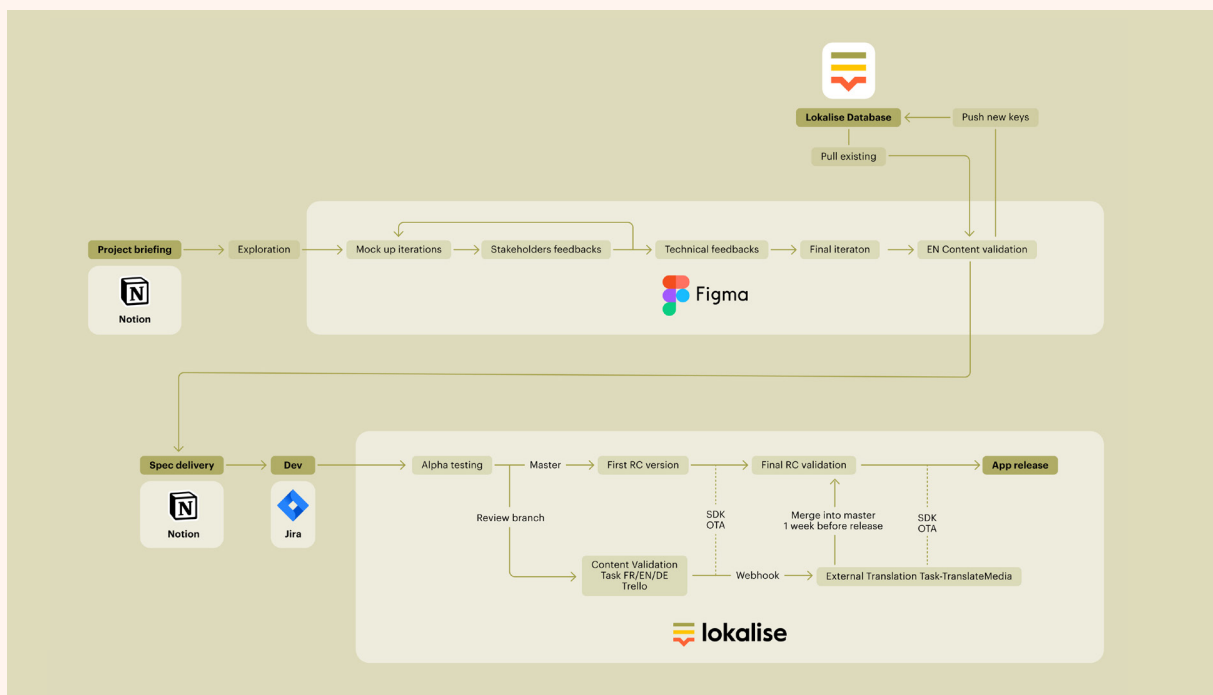
1. The designers/content team create the design with key names in Figma/Sketch.
2. The keys and base language are pushed to Lokalise.
3. Developers get the keys from a duplicate page in Figma/Sketch.
4. Linguists work in Lokalise throughout the entire flow.

The flow above allows designers and developers to dovetail their efforts for a localization cycle that follows the pace of development and of each feature release.

Instead of waiting for translations, development can start with key names reflected in the design as opposed to language values. Plus, teams can test the design using machine translation and set character limits to ensure that the design won't break.

The Withings workflow

This is a workflow pioneered by the product team at [Withings](#). It allows for developers to focus on developing the product in parallel with the translation workflow.



[Download](#)

1. Design & concept validation process is done in Figma
2. Design is consolidated in Figma
3. Product Owner (PO) pushes keys and screenshots to Lokalise via Figma plugin & pulls existing strings for content validation
4. Base language content qualification performed in Lokalise
5. If PO deletes a key, the dev team is notified via Slack
6. PO creates a task in a project branch for in-house translation cycles and in-house translators are notified via the Trello integration (French/English/German)
7. PO creates a second task in a separate project branch to inform LSP to translate all remaining languages (LSP integration via webhooks)
8. Once task is closed, keys are automatically tagged & Slack notification is triggered
9. PO then merges branches so that strings are ready 1 week before release to production environment

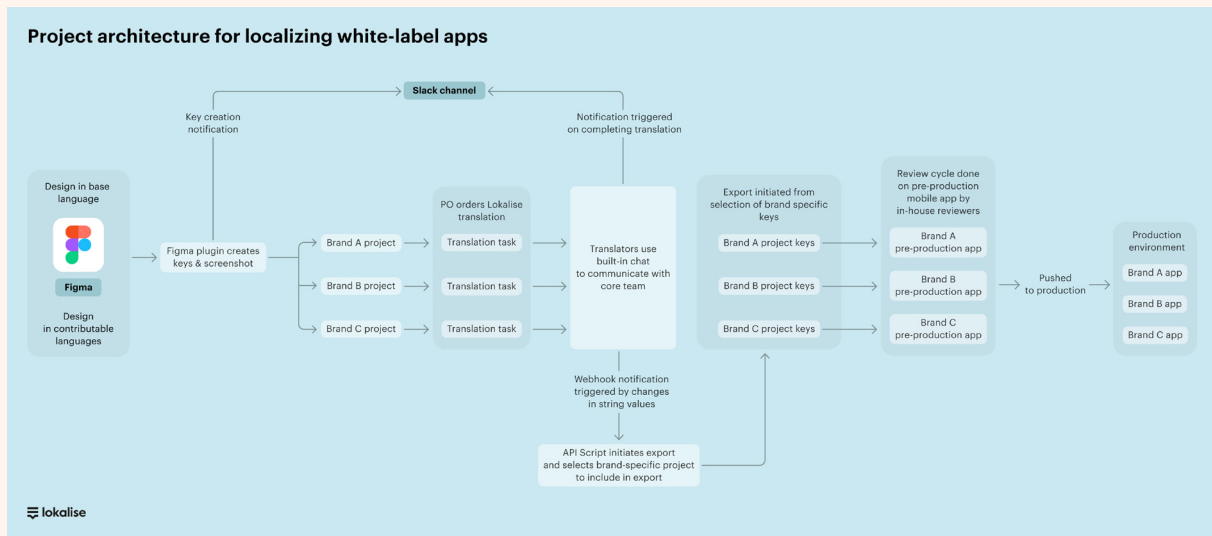
See it live:

Join us for a discussion on the Nimdzi podcast on September 28, 2021, to learn about the above workflow from Romain Dahan, Product Manager at Withings. You'll leave with Romain's insights on the workflow and what it takes to get stakeholders on board.

[Register here.](#)

Whitelabel workflow

The below is a workflow that some of our customers use to whitelabel their apps.



[Download](#)

1. The Figma plugin is used to create keys and screenshots, with 1 Figma design per brand connected to 1 Lokalise project
2. Slack notifications are triggered when new keys are created.
3. PO selects newly created keys and creates a Lokalise Translation Order.
4. Lokalise translators translate on platform & communicate via built-in chat.
5. Slack notification is sent out once the translation cycle is complete.
6. When any key export is initiated for a brand-specific project, a download is initiated from the generic project and the option to include keys from brand-specific project is selected.
7. Mobile bundles: uses webhooks & API calls to automatically download new bundles when changes in keys are detected.
8. Review cycle is done in-house on pre-production mobile app versions.
9. PO is notified and manually changes the strings based on feedback.
10. Once a pre-production app has passed in-house review, new releases are pushed to production.

How to get started

To create a unified localization workflow at the design stage, you can use a localization platform like Lokalise to integrate your design tools and the rest of your tech stack.

[Visit our designers page to learn how to get started.](#)

(P.S: Try it free to see the design integrations in action).

What types of companies and products does it suit best?

Success stories from our customer base show that fast-growing, agile companies that rely on CI/CD to any degree are natural adopters of design-stage localization.

Companies that rely on continuous development for products like mobile apps, web apps, and games already have the processes and technologies to fit design-stage localization into their workflows.

That said, our research also shows that companies that are between levels 1 and 2 of the [localization maturity model](#) can easily implement and benefit from design-stage localization.

In short, tech-savvy companies with strong product teams will find it especially helpful for accelerating localization, centralizing collaboration, and achieving their international expansion goals.

When is it the right approach?

If your priority is **quickly launching an MVP and you only operate in a few other markets that are similar to your core market**, a traditional localization workflow will likely be more effective.

But if you've **found a product-market fit and want to deliver great UX in multiple markets simultaneously**, then design-stage localization is the optimal workflow for your stakeholder and company.

Why should companies start localizing at the design stage?

When implemented correctly, design-stage localization almost always results in higher team productivity, more collaboration, faster time to market, and lower risks than traditional approaches.

The value of design-stage localization

When Romain Dahan created [Withings' design-stage localization workflow](#), the most important thing he did was align his stakeholders on the goal of localization: providing a great digital experience for customers globally.

In the long run, your most important assets are your customers and team. Bearing that in mind, here's how design-stage localization helps you collaborate effectively with your team and deliver the best possible product for your customers.

1. Streamline collaboration, align stakeholders, and get early feedback

Efficient, simple collaboration and excellent communication are key to building strong partnerships between localization and the parts of your business that require localization support.

With design-stage localization, you can streamline collaboration between your designers, developers, product team, and content team. Your team can work together before you build, while ensuring everyone is working from the same set of information and is focused on the same priorities.

In the words of [Keith Vaz](#), Interaction Designer at the super app [Gojek](#):



“UX writers can now begin the work a lot earlier and the workflow has urged us to work closely with each other to get things on track.”

With real-time translation updates in Figma, there is now a single source of truth for copy for both designers and copywriters.

This means, “no more having to go through several hefty Google Sheets, hunting for one line of copy and then having to manually copy-paste it into the designs.” No more sending emails to ask, “which is the latest wireframe?” or “is it done yet?”

2. Improve UI quality and get early feedback

Without leaving Figma, your team can create fully localized prototypes to get the gist of what the localized environment will look like. Translators can then see the entire UI as they translate content in Lokalise.

This means you can improve the quality of UI localization because:

- a) You can spot cultural and localization issues before you build
- b) UX writers aren't always limited to the design as designers can make changes quickly to accommodate copy
- c) Adjustments in both copy and design happen together

Aside from eliminating unnecessary costs borne by making adjustments post-release, it allows product teams to collect valuable feedback from users, local marketing teams, and other stakeholders before releasing a product in multiple markets:

“Translating our designs to our users’ native languages for user testing sessions is super seamless now,” says Keith Vaz. He adds that, “earlier it would take a few hours for the UX writer or the designer to manually edit all the copy on each screen.”

3. Prevent costly design breaks and localization errors

If you're designing something for multiple regions, you can imagine how re-adapting the product interface can slow down the entire localization process. It creates friction in designer-developer relations, unnecessary work for project managers and translators, and makes shipping localized products quickly (without errors) nearly impossible.

While designers can easily alter the design, developers have a much bigger job in reworking the code to reflect the new design. Often, this forces teams to choose between delivering localized products that don't look great or shipping late.

When you integrate designers into the localization flow early on, teams can quickly and easily check whether the design functions correctly across all target languages and make amendments pre-development. Here's how:

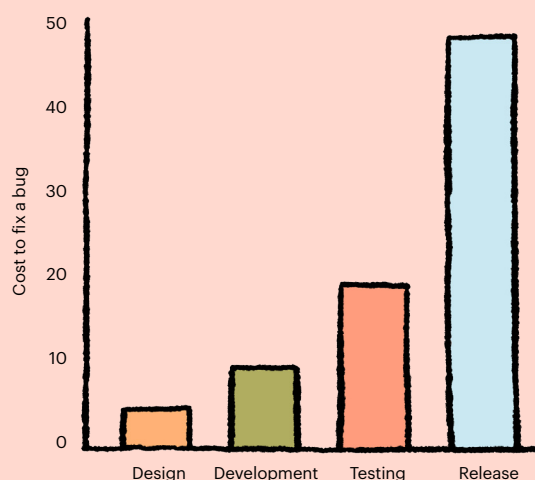
1. Linguists working with visual context can request design amendments to accommodate the translated text. Alternatively, you can set character limits to easily communicate constraints to linguists/UX writers and avoid excess copy that can break the design.
2. Once the content is finished or almost finished, designers can pull the translated text back into the design and perform design QA before the product goes to development.

And remember, if the content does change, you'll be able to adjust both the copy and design. Plus, you'll be able to leverage the translation memory to offset any costs caused by translating new content.

Did you know?: Resolving bugs in the design stage is 10x more cost effective than making adjustments after release.

Up to **10x cost decrease** if you resolve a bug in a design stage

Lingoport, Raygun



4. Get to market faster and match localization with agile development

Working in localization, you may have experienced the frustration of the process' reputation for apparently being slow, manual, and complicated.

Well, design-stage localization changes that.

Take this example from Luisana Valero, UX Designer at Elli (VW Group), who says they are now able to “deliver assets about 5 times faster for multi-language asset creation (app screens).”

At Gojek, Keith Vaz says, “accommodating manual copy-pasting ends up being a job for a day or two.” He adds that with the new workflow, this time should be cut to just an hour or two.

With faster turnaround times and a process that's embedded in product development, keeping up with continuous development becomes manageable. Teams working on localization are no longer slow and complicated and can partner with product teams to seamlessly marry localization and development.

Bearing that in mind, we were especially delighted to hear how Withings managed to create an efficient l10n process involving their product team, designers, developers, and content team.

In the words of their product manager, Romain Dahan, “By migrating to Lokalise, we've accelerated the delivery of new localized features by 90%.” He tells us that Lokalise's Figma integration served as the key to transforming the localization process at Withings and building a design-stage workflow.

Chapter 2:

Top design-stage challenges for localization teams

You might be asking “should I use machine translation to get an idea of what a localized prototype will look like?” ... “can our developers remain in control of key naming conventions?” ... “will this add to our designers’ (already very full) workload?”

Whatever the case, you likely have questions. And we’re here to answer them. For this discussion, let’s start by laying out the top design-stage challenges.

Shifting the way you work and ramping up stakeholders

A design-stage process demands something from every department and implementing it successfully requires a shift in mindset and how you work.

Shifting localization from a one-off activity that happens post-release to a continuous process that happens pre-release impacts your stakeholders – how you collaborate, share information, and function as a team.

Think about it like this: If your UX writer is the last in the pipeline of the design process before development handoff, it will require some initial coordination to ensure that they start their work earlier in the localization process. The same applies to other team members who have certain dependencies that need to take place before they begin their work.

The challenge is coordinating this shift between all the stakeholders – even if they aren’t entirely new to localization, it requires some basic onboarding to educate and ramp up your team. For example, designers may not be familiar with some newer terms like ‘[keys](#).’ They may require some training on how to link

keys to the text elements in the design—and it will likely take some initial effort to get them up to speed.

Breaking silos between tools, processes, and teams

Aside from inconsistent systems and processes, many of the problems that arise in localization programs are due to the fact that designers, developers, content teams, and localization teams work in silos.

When it comes to localization, there are four things that should be shared and integrated in some form. They are:

1. Lokalise
2. Design tools
3. The wireframes of the entire UI being localized
4. Your localization team

It's simple: All of the above have shared processes that work in tandem to create a unified, continuous localization process that starts at the design stage. In an ideal world, your prototypes will live in a design tool with existing APIs or connectors to your TMS. You will also need defined collaboration rules to help you function better as a team.

Centralizing tools and processes

Centralizing tools and changing processes is hard in the short term and your team might not want to change.

Some larger companies have designers, developers, and other stakeholders that work in different ways. For example, one multinational e-commerce company present in 17 markets has several different design teams which all have their own individual processes.

In fact, it's fairly common for companies to have multiple systems and

processes for managing content and prototyping product flows, especially in larger organizations. Aside from being an obstacle to effective team collaboration, multiple tools and inconsistent systems will create scalability issues further down the road.

Ideally, you will have a TMS that is connected to the design tool (e.g., Figma) you use company-wide. Often, companies use their TMS as the single source of truth for content in multiple languages, along with the English equivalent. However, some companies use their design tool as a single source of truth.

Bottom line: You need tools and processes that are truly central (and integrated) company-wide to get a design-stage localization process working like a well-oiled machine. Plus, ensure your TMS has integrations with your existing tech stack.

Conflicting key naming conventions

In some companies, engineers are the ones who remain in control of [key naming conventions](#). They may already have the bulk of their core product localized and a specific key naming pattern they would like to continue using.

The truth is, there is no single best practice here. But the good news is that a design-stage process accommodates flexible ways of working that suit various team structures, workflows, and tech stacks.

You may decide not to automate key names completely or create keys at the design stage and continue having your developers create keys instead. For example, one company decided to stick with engineers creating keys instead of them coming from Figma first.

The bottom line is: you need a process to align key naming with design systems and ensure you don't have conflicting key naming conventions or are not left waiting for engineers to create keys that will be linked to the design.

Note: If you're using Lokalise, you have various options to continue using your legacy key naming conventions, which we cover below. We'll also walk you through exactly how it works in our complete guide to design-stage localization.

Chapter 3:

Design-stage localization case studies

Design-stage localization is powerful, but don't just take our word for it: here are a couple of case studies to inspire you to take action.

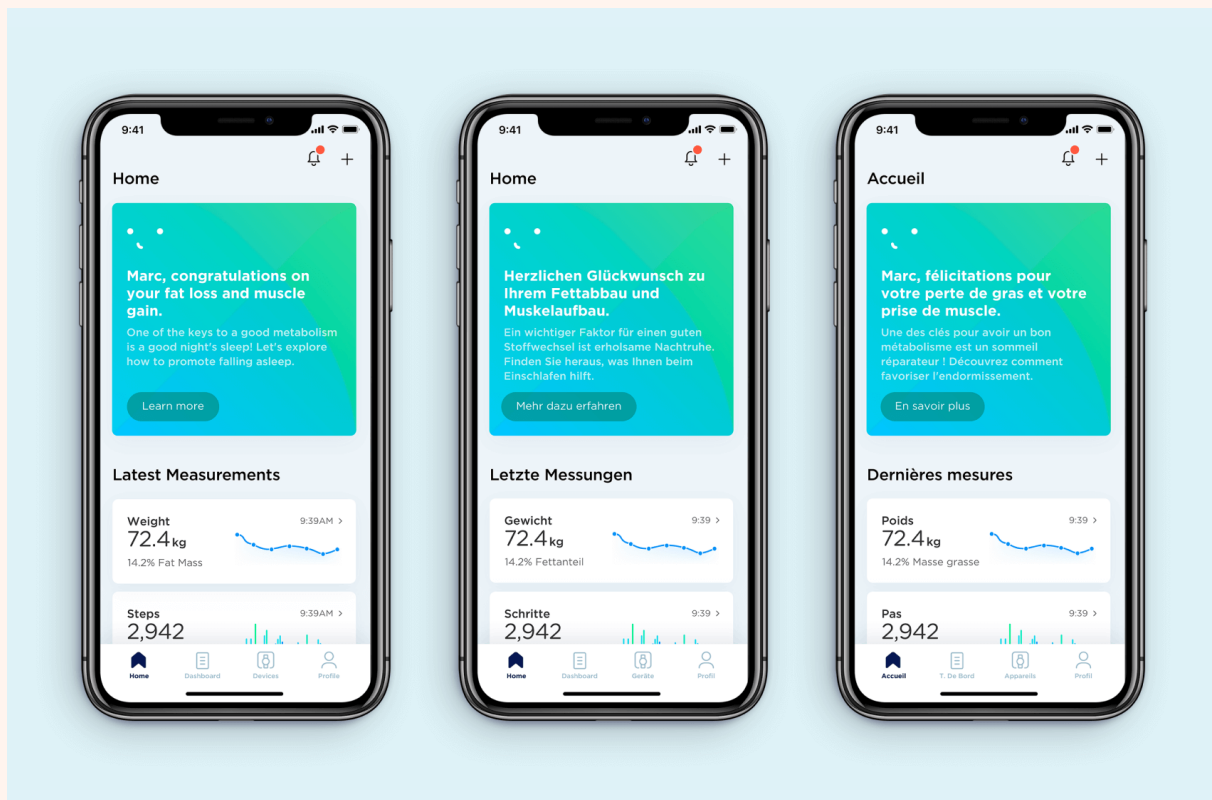
We've covered the value of design-stage localization above, but here's a quick recap – you can:

- Streamline collaboration, align stakeholders, and get early feedback
- Match localization with the pace of agile development
- Prevent costly design breaks and resolve errors pre-release
- Improve UI quality
- Launch in multiple markets simultaneously

Now, let's look at two examples for some quantifiable benefits in some of these areas.

90% faster feature rollout with design-stage localization at Withings

The context: Withings is a connected health pioneer that is localizing their Health Mate app (currently available in 190 countries and 11 languages). They deliver 3 features per release, and each originally took around 10 hours of manual localization work to create and search for keys, link duplicates, extract keys to create tasks for reviewers, and so on. Instead of focusing on strategy, the team at Withings was struggling to match quality localization with the pace of development.



The challenge design-stage localization solved – Implementing design-stage localization with Lokalise & Figma helped the team to:

- ➔ Seamlessly integrate localization into their product development workflow
- ➔ Break silos between departments, so teams can collaborate with overall context and coherence
- ➔ Spot localization defects before building as linguists have context for the entire UI
- ➔ Speed up the process as developers can start building without waiting for translations
- ➔ Create an efficient l10n process involving the content team, designers, and developers.

The results: The product manager at Withings went from 10 hours to 1 hour of task management spent on localizing a new feature. Plus, there are significantly fewer keys to go through each month, which means more time spent on improving localization quality. The bottom line: Withings gets localized products to market faster with far fewer “local customer complaints about translation quality.”

[Read the full case study.](#)



Localization related bugs at DBS fell from 30% to 8%

The context: DBS is one of the leading banks in Southeast Asia, serving consumer markets in Singapore, China, Indonesia, India, Taiwan, and Hong Kong.

The challenge design-stage localization solved: By using the Figma plugin, the l10n workflow has become much smoother and helped the team to:

- ➔ Solve issues with text not fitting into the design
- ➔ Reduce endless back-and-forths in email threads

- Give writers and developers context for what each key is used for in the design
- Remove confusion about which is the latest version for wireframes

The results: “After introducing Lokalise, localization-related bugs fell from 30% to below 8%. It helped improve the efficiency of the localization process and the team and made it a lot easier for DBS to create relevant, localized user experiences for global customers.

[Read the full case study.](#)

Key metrics to measure the success & goals of design-stage localization

There are myriad metrics and key performance indicators used to measure the efficiency of localization efforts. This is where most organizations use KPIs or measurements, such as:

Key metrics:

- On-time deliveries
- Number of linguistic or technical quality errors
- Project delays or extension requests
- Average turnaround time
- Time to market
- Adherence to budget

Business metrics look at the bigger picture and consider the contributions that localization makes towards overall business goals. Since the goal is never just to localize, what specific business metrics are the best to look at?

The most obvious candidate is ROI. While it consists of hard-to-quantify factors, implementing a design-stage localization flow should impact ROI when factoring in localization costs.

That said, whereas ROI is very important, it doesn't always come down to that. It might be creating an equitable experience for customers around the globe, delivering delight in the form of native experiences, or building brand awareness.

The bottom line is: you should align any new strategy with clear goals based on your primary business goals.

The bottom line is: you should align any new strategy with clear goals based on your primary business goals.

Who is it for? Your team in the design-stage localization process

There are several stakeholders that should make up every localization team. The localization process typically involves four primary groups of stakeholders:

1. Designers and software developers
2. Product, project, and localization managers
3. Copywriters and linguists
4. QA specialists and reviewers

Let's examine these roles, how they change, and how they impact the design-stage localization process.

Designers



As a designer, you already prototype user experiences in the base language and check to see if text fits within buttons, how layouts change with different devices, and make design decisions based on the best product flows. The only difference in a design-stage localization workflow is that you are prototyping experiences for global customers, simultaneously.

Here's what changes:

Key responsibility

Ensuring your product looks and feels how it should resonate with your global customers.

Skills

- Great organizational and time-management skills
- Creative and technical skills for developing localized concepts, graphics, and layouts
- Specific knowledge of UI and UX design (if the project requires)
- Ability to clearly communicate with all team members
- Ability to adjust the design according to feedback

Main challenges

- Quality issues due to localization after the design stage
- Making sure translated text fits the design and design changes
- Ensuring the localization is appropriate for the target market's culture (e.g., avoiding human imagery for Islamic audiences)
- Ensuring right-to-left languages are suited to the design
- Lack of context and the inability to create relevant visual experiences
- Jeopardized design quality because of an inefficient localization process (e.g., poor communication, idle waiting time, inefficient asynchronous collaboration)
- Preserving brand identity while still adapting to new markets.

Changes in the way you work

- Adopt a leading role in multilingual product development
- Ability to collaborate with overall coherence of product experience and avoid working in silos
- Begin localization work earlier in the product development process to resolve localization errors (like design breaks) in a much more agile way, prior to building
- Eliminating repetitive tasks like manual copy-pasting to test designs
- Understanding how the length of text for different languages impacts the digital experience during the prototyping process

Software Developers



As a developer working in an agile product development team that relies on CI/CD to some degree, design-stage localization is a natural fit for your development workflow.

Whether you want to remain in control of the key naming convention or pass on this responsibility to the localization or project manager is a flexible choice you can agree on with your team. The most important thing is that you will no longer be bombarded with Jira tickets related to localization. Here's what will change:

Key responsibility

Initiating the localization process by adding string identifiers (i.e., keys) to the product code, as well as fixing bugs and taking full care of the back-end of your digital product.

Skills

- Great knowledge of different programming languages as well as ability to internationalize code in the respective programming language
- An eye for detail and problem-solving skills
- Communication skills and ability to work in a cross-functional team
- Great knowledge of localization tools and integration tools such as API, CI/CD, etc. (Knowing how to connect them to a chosen translation management system is a must)
- Comprehensive knowledge of content management systems and different technologies (e.g., publishing and editing software)
- Open-minded, hands-on, and result oriented

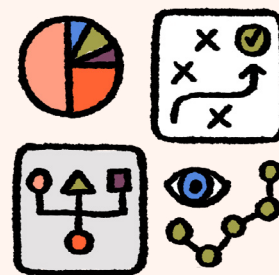
Main challenges

- Synchronizing translations between the TMS and code repository
- Downloading new translations in proper formats, and monitoring for changes
- Finding duplicate translations
- Allowing translators to work on features in parallel
- Using placeholders and plural keys properly
- Previewing translations in the design stage
- Providing context for translators

Changes in the way you work

- Start working earlier in the localization process
- Ability to work closely with product, designers, and the content team before the build
- Have more context of where translation keys are being used in the design
- Eliminate time spent correcting builds to fit new translations
- Eliminate time spent providing context to translators as context is provided directly via design tool integrations
- Ability to begin builds without translations in more advanced design-stage workflows

Product, localization, and project managers



If you're a product, localization, or project manager, here are some useful questions you should ask yourself about your existing workflow:

- How much time are you spending on task management and reviewing design bugs?
- Are different team members able to work in parallel rather than waiting for each other to complete work?
- How is this sequential approach impacting delivery time?

You want to be focused on product development, improving localization quality, and getting localized products to market faster – here's what will change with a design-stage process:

Key responsibility

Educating all stakeholders in the localization process about your product while also identifying requirements, tools, technologies, and the overall strategy.

Skills

- | | |
|---|--|
| • Great leadership and managerial skills | • Very result oriented |
| • Ability to prioritize, establish procedures, and set a system for tracking progress | • High level of social and emotional intelligence |
| • Ability to optimally allocate resources | • Ability to do qualitative customer research and establish a strong product roadmap for international expansion |
| • Agility, strategic, and analytics skills | |

Main challenges

- Delays caused by poor process efficiency and translations arriving late
- Delays caused by the multiple iterations needed in QA after the translation files have been received from the translator
- Converting localization files that come in different formats from different translators
- Having the time to work on product development without worrying about how to localize
- Ensuring the team stays motivated despite bottlenecks

Changes in the way you work

- Reduced time spent on project management and an increased focus on strategy
- Establishing new processes, documentation, collaboration rules, and training the product team on shifting workflow
- Clearly explaining to your stakeholders the impact of implementing a design-stage localization workflow
- The risk of a bad product/market fit is significantly reduced
- Collaboration between designers, developers, managers, and translators is streamlined, back and forths are eliminated, as well as idle waiting time

Content Team



The biggest shift for copywriters and linguists is that they can work together with product, designers, and developers before the build. That means you can ensure the overall coherence of the brand and product experience to create delightful, global experiences.

Key responsibility

Preparing all the content that needs to be localized and/or participating in [transcreation](#) to ensure it's well adapted to the target market.

Skills

- Strong strategic and writing skills
- Team player and great communicator
- Responsive to feedback and able to work in a fast-paced environment
- Ability to create and/or transform content that's culturally adapted while still on brand
- Ability to incorporate storytelling and use copywriting to create an emotional response and trigger a desired action from the target locale
- Ability to follow the set quality and style guidelines and switch between different tones of voice
- Good time-management skills

Main challenges

- Making sure copy sounds good in all languages
- Compromising translated copy quality due to character limitations imposed by designs
- Getting access to translations for temporary app/web content (e.g., sales enablement content or specific content campaigns)
- Constant translations
- Issues that come with outsourcing (e.g., tiring back and forths, missed deadlines due to inefficiency)
- Tracking the progress of campaigns
- Scaling without jeopardizing quality
- Adapting content culturally
- Reporting

Changes in the way you work

- Begin work earlier in the localization process as opposed to being the last step in the pipeline before dev handoff
- Ability to request design changes to accommodate copy that will resonate best with the target audience
- Have visibility over the entire UI and can better understand the flow and what is being built
- Linguists work in Lokalise throughout the entire localization process

Quality Assurance **Experts and** **Reviewers**



In a design-stage flow, QA is a collaborative process between the entire team. All the stakeholders can see the fully localized wireframes and share their feedback pre-release. The biggest shift is that teams must agree on quality before anything is built.

Key responsibility

Establishing a rigorous review process through proofreading and testing in order to deliver the agreed translation quality.

Skills

- Native speaker and/or advanced knowledge of a language and relevant domain
- Very detail oriented, with a strong problem-solving mindset
- Ability to work in accordance with quality standards
- Great multitasking and time-management skills
- Strong sense of ownership
- Receptive to feedback and comfortable with working in a team
- Strong verbal and written communication skills
- Deep knowledge of the product to identify language key changes and their impacts quickly

Main challenges

- Lack of visibility and standardization in translation project management process
- Spending too much time on quality assurance due to inefficient workflows
- Addressing ambiguity about terms to ensure translation consistency
- Difficulty spotting and fixing spelling and grammatical errors without missing deadlines
- Poor level of language comprehension of the locale
- Mastering the steep learning curve when it comes to the localization tech stack
- Not having a standard for performance and usability testing
- Too many manual and repetitive tasks, as well as tiring back and forths

Changes in the way you work

- It is possible to catch design breaks at an early stage of the product development process
- Time to market gets shorter while the quality of both translations and multilingual UX gets higher
- The risk of localization errors in the product is significantly reduced

How to implement a design-stage localization workflow around Lokalise & your design tools

To implement a design-stage workflow, It's important to scale things down. Start by putting together a proof of concept and focus on doing a pilot project using a localization platform like Lokalise.

If you are new to Lokalise, no problem: there is a [free trial](#) you can choose if you just want to test it out. Head over to our [designers page](#) and sign up for a free account (or look at the [pricing page](#) if you want to give the Pro plan a go).

Once you've signed up, click on the video below and watch the series that covers all the steps you need to take to connect your design tools and automatically start pulling translated strings into your designs:



Now that you have a good understanding of what design-stage localization is and how it can help you improve your website, it's time to take action.

Get the free checklist to help you and your team on your localization journey and create a free account to apply some of the guidelines today.

Send me the checklist

