

Is your product ready for French localization?

A 20-point readiness checklist for SaaS, tech & gaming
teams — before you translate a single string.

Localization fails before translation begins

Most French localization problems aren't translation problems — they're readiness problems. Strings that can't be translated correctly, a UI that breaks when text grows 20% longer, a codebase that assumes English plural rules. By the time a linguist sees the file, the mistake is already baked in.

This checklist is the audit I run before taking on a French localization project. Go through it with your team: every unchecked box is a risk you can fix now — cheaply — instead of after launch.

How to use it: tick each box your product already satisfies. Count your checks at the end and read your readiness score. Anything unticked is your to-do list.

01 Technical foundations

☐ **Your product supports Unicode (UTF-8) everywhere.**

Accented characters (é, à, ç) and « » guillemets must display and store correctly across UI, database and emails.

☐ **All user-facing text lives in resource files, not hard-coded in the source.**

If a string is buried in the code, it can't be sent for translation without a developer digging it out.

☐ **Your UI tolerates text expansion of 20–30%.**

French runs longer than English. Buttons, menus and labels must not truncate or break the layout when text grows.

☐ **You use a localization-friendly string format (ICU MessageFormat or equivalent).**

It lets the target language define its own plural and gender rules instead of inheriting English ones.

☐ **Dates, numbers, currencies and units are formatted by locale, not hard-coded.**

France uses 24-hour time, a comma decimal separator (3,5), space thousands separators and the euro after the amount (12 €).

02 Strings & content quality

☐ **Sentences are never built by concatenating fragments.**

"You have " + count + " items" can't be translated properly — French word order and agreement differ. Use full parameterized strings.

☐ **Every placeholder has a descriptive name.**

{userName} tells the translator what it is; {0} forces them to guess and risks errors.

☐ **Strings come with context (screenshots, comments, or a description).**

"Post" is a verb or a noun depending on context — and translates to two different French words.

☐ **Your source English is clean and consistent.**

Inconsistent terminology in English becomes inconsistent — and more expensive — in French.

☐ **Idioms, humor and cultural references are flagged.**

They rarely translate literally and need adaptation, not translation. Mark them so they aren't rendered word-for-word.

03 Brand, tone & market fit

- ☐ **You've decided on tu vs. vous for addressing users.**
French forces a formality choice English doesn't have. It defines your entire brand voice and must be consistent everywhere.
- ☐ **You have (or are ready to build) a French style guide and glossary.**
Which terms stay in English, which are translated, how your product names itself — decided once, applied everywhere.

No style guide or glossary yet? Building them is one of the core services I offer — it's often the best first step.
- ☐ **Your CTAs will be adapted, not translated literally.**
"Get started" ≠ « Commencer ». French calls to action follow different conventions of register and urgency.
- ☐ **Marketing and UI copy share consistent terminology.**
A feature called one thing on your landing page and another in-app erodes trust — doubly so in a new language.
- ☐ **You've considered French SEO/GEO, not just translation.**
French users search with different terms — and increasingly ask AI assistants. Translated keywords ≠ researched keywords.

04 Legal, support & process

- ☐ **Legal texts (terms, privacy, GDPR notices) are planned for French.**
France enforces GDPR strictly and expects consumer-facing legal content in French. This is specialized translation, not an afterthought.
- ☐ **Your support channels can handle French inquiries.**
Localizing the product but not the support experience creates a broken promise at the first problem.
- ☐ **You have a plan for continuous localization, not a one-off.**
Every release adds strings. Without a process, your French version drifts out of sync within weeks.
- ☐ **A native French linguist will review in context before launch.**
Spreadsheet translation misses truncations, wrong register and context errors that only appear in the live UI.
- ☐ **You know who owns French quality on your side.**
A single point of contact for linguistic decisions keeps quality consistent as your team and product grow.

This is exactly the Language Lead role I take on for product teams — owning the guidelines, the glossary and the reviews.

Your readiness score

- **16–20 checks — Ready.** Your foundations are solid. Localization will be smooth; focus on linguistic quality and market fit.
- **10–15 checks — Almost there.** A few structural fixes now will save you costly rework after launch. Prioritize the technical section.
- **Under 10 checks — Prepare first.** Translating now would bake in problems. Fix the foundations before sending anything to a linguist — it's far cheaper at this stage.

Want a second pair of eyes « en français » ?

I'm Dorine — a freelance Language Lead who helps SaaS, tech and gaming teams make their products feel native in French. If this checklist surfaced a few gaps, that's exactly where I come in.

[Start a conversation →](#)

frenchsaas.com · contact@dorineparmentier.com