

10+1 Tips to prepare your software for the world



localization • translation • interpreting • consulting

In computing, internationalization and localization are means of adapting computer software to different languages, regional differences and technical requirements of a target market.

Internationalization is the process of designing a software application so that it can potentially be adapted to various languages and regions without engineering changes.

Localization is the process of adapting the application content for a specific region or language by adding locale-specific components and translating text.

If you are thinking of or currently creating a new website/software with the purpose of localizing it into other languages, then you will find the following tips quite useful:

1. Analyze the situation and plan ahead

Many companies, do not think about software localization until the last-minute before a product release. Scheduling and planning should be made well in advance and it should take into account all the necessary steps of the process: translation, review, testing and regression in order to deliver a quality product.

2. Externalize all translatable content

Taking the text out of the code and placing it in resource files is the first step towards a properly internationalized application. Separating the text to be localized from the code helps to avoid code duplication issues and also lets translators and engineers work on updates at the same time. Furthermore, it removes the possibility of damaging code during translation.

3. Invest in a style guide and glossary

The style guide defines the style, terminology and conventions from the beginning and provides uniformity in style and formatting throughout your software and documentation. It improves the quality of your translations, minimizes inconsistencies, adds professionalism to your work and saves time and money.

4. Understand translation tools

Learning more information about how translation tools work will help you take maximum advantage of this technology. The Translation Memory is a tool that stores all translations into a database in real-time as the translator works. This database stores "segments", which can be sentences, paragraphs or sentence-like units (headings, titles or elements in a list) that have previously been translated, in order to aid translators. These segments can be reused when the same segment is repeated elsewhere in the project, or in updates. These tools greatly diminish the time and cost of translation.

5. Provide room for text expansion

Translated text in most languages takes up to 30 percent more space than the English text. Leave enough room on your layout for expansion or program dynamic UI expansion into your software. If there are strings that cannot exceed a certain size, you should include comments in the resource file for those items.

6. Use Unicode/UTF-8 encoding of strings

Make sure to always source your string tables or software resources in Unicode/UTF-8 encoding. These character sets are created to enable support of any written language worldwide. Having just one way to process text reduces development and support costs, helps to avoid extra conversion steps, improves time-to-market, and allows for one single version of source code.

7. Avoid concatenation and overuse of single strings

Most languages do not follow the English syntax and word order. Concatenated strings and strings that are used in multiple contexts end up having awkward grammatical constructions and gender agreement issues.

Concatenation only works when content is written for a specific language. Now, when it comes to localization, concatenation makes it difficult – even impossible in certain cases.

8. Internationalize dates/numbers etc

This step is very important because it enables dates, numbers, and other region-specific data to appear in a familiar way to users all around the world. Such data may differ even between regions that speak the same language. For example, while US use MM/DD/YYYY for date UK use DD/MM/YYYY.

Developers writing code should always remember that other countries:

- Might format their dates and times differently
- Might not take the same days off at the weekend
- Might use an entirely different calendar system
- Might be more used to numbering their weeks than naming their months
- Might be in a time zone that has a partial-hour offset
- Might use a different measurement system
- Likely use a currency or currencies different to your own
- Use different phone number formats

9. Provide comments in software resources

The use of comments in software resources can be very helpful for translators because knowing the context and use of certain strings can help them choose the right translation from the beginning. Most translation tools will let translators see these comments while translating.

10. Localize Help and software (GUI) at the same time

Many non-English users around the world have noticed when Help or a User guide prompts them to click on a button that is worded differently in the software itself. Try to localize the user's manual, online help files and graphical user interface (GUI) at the same time to ensure consistency.

11. Test your software

Testing the software before its release is an integral step in the translation process. It should be performed by trained localization QA professionals and it will help to expose possible technical issues related to UI sizing, text truncation, hard-coded strings, character corruption and over-translation. This final step also gives the linguist the chance to actually see his translation in full context, often resulting in necessary changes to the translation.



References

<http://en.wikipedia.org>

http://www.testingexperience.com/issues/testingexperience10_06_10.pdf

<http://userguide.icu-project.org/i18n>

<http://eyegene.ophty.med.umich.edu/unicode/>