

# **What precautions must be taken into account during the software development stages while localising software?**

**COMP160 - Software Engineering Essay**

1608289

March 30, 2017

I propose to underline the key components needed within localised software while developig it. I will go through each component thoroughly with relevant references to assist and also back up my key points. I will also describe how to go through the quality assurance stages of the development and the limitations the quality assurance stages can be hard to determine the overall quality of the software.

## **1 Introduction**

Throughout this essay I will underline a few of the most important precautions you must follow when developing software for localising worldwide. I will give examples of software that is used worldwide but excludes certain languages and how it doesn't correctly format them to the country's customs or desired needs. The majority of modern software is localised to reach a much larger

audience than intended a couple of decades ago. Many companies wish to expand their profits and to also offer their services to other countries other than their own. By the end of this essay it should give the reader an insight of what components are required when localising software.

## **2 The key components to localising software**

When localising software all the key components need to be set appropriately, otherwise the software won't serve its intentional purpose to provide a service to a much wider audience than the original intended target audience. The target audience when localising software will range from a couple to a few different audiences. This means that the software needs to be perfectly accessible to each target audience with ease to provide a much smoother experience. This can prove to be challenging and also a lot of work, a lot of research will need to be performed beforehand into the different languages, currencies and units of measure. The key components to localising software are as follows:

- "Adapting graphics to target markets
- Modifying content to suit the tastes and consumption habits of other markets
- Adapting design and layout to properly display translated text
- Converting to local requirements (such as currencies and units of measure)
- Using proper local formats for dates, addresses, and phone numbers
- Addressing local regulations and legal requirements"[1]

Google is a prime example of a software that has been localised professionally with near perfect standards. Google will translate, automatically convert currencies, units of measure, current times/dates, address and also phone

numbers to suit the country that the user is using the software in. For example, if I were to use Google in the UK then it would automatically be in English (UK) rather than English (US) and will also show our currency, units of measure and also current times/dates appropriately with the text in the correct format which is quick and easy to read. Although Google tailor for many different countries and also cultures it doesn't strictly cover every language and culture, for example, "Baidu beats Google in China." [1] This is because Baidu is more popular when compared to Google as Baidu formats appropriately and also provides a near perfect service than Google as well as precise and accurate research results.

### **3 Quality assurance**

Quality assurance is one of the most important processes before releasing a localised software. Although the translation stages can be time-consuming and also very obnoxious to undertake due to the high standards set by multiple quality assurance tools, there are many factors that need to be taken into account when undertaking quality assurance, these are:

- "Understanding of the language and its current usage. Fluency in both the source language and target language must be observed. Mere translation of text may reflect an inherent weakness in the translation and may yield adverse results.
- Understanding of target user environment with its complex social and cultural implications. No matter how long the person involved in the localization has lived in the target language's country, cultural perception is different. And no matter how much the individual claims to know the language and its people, there are still aspects of the culture that

may not be acquireable.

- Target software should possess the Look-and-Feel of the original program.”[2]

Text must make sense to the target audience intended, this means a mere text translation will have multiple errors such as misspellings, punctuation errors as well as not making an sense. One of the best possible ways to ensure the translation of the text is correct is to have a fluent translator for the target language to proof-read and correct it afterwards. Although this manual method can take great amounts of time to undertake, it will should provide a more precise translation than an automated translation tool. The UI of the software should maintain the same or similar format to the original program. An example of this is again, Google. Google’s format remains the same for each language while altering slightly to adapt the formatting to how the language is read/laid out. An example of this is the Japanese language, it is laid out from right to left.

## **4 How do we measure the quality of localisation?**

Defining professionally developed localised software can be hard. ”Given the inherent complexity of internationalisation, and its potential impact upon the development process - from requirements analysis right through to the task of support and maintenance of the delivered product”[3] There are many approaches to assessing the quality of localisation, ”one reasonable approach is to adapt to the Test Process Improvement Model which uses a matrix ranking testing performance across twenty key areas. A study of companies which have been successful in developing a product for diverse international markets is important to guide the identification of key areas and maturity levels for such a model.”[3] By comparing a series of successful

companies localised software can give a clear and precise analysis of your own localised software via comparing them altogether. This can also give the company an open window of how to improve by adding more components that other companies haven't done in the past, however, this can be a risk. By attempting to add more components to localised software it can give one of two results, an improvement or deteriorate the new software.

## 5 Conclusion

Throughout this essay I have underlined the precautions and key components for developing localised software. If any of the key components I have listed above are missing or contain errors then the software will refrain from becoming successful until the patches have been made. Releasing localised software that contains errors can severely deteriorate the software's reputation and therefore reduce the amount of users within the future. In order to release a successful and fluent localised software then multiple quality assurance tests may need to be undertaken throughout various stages through the development stages.

## References

- [1] "What is localization?" Sep 2015.  
[Online]. Available: <https://www.gala-global.org/language-industry/intro-language-industry/what-localization>
- [2] M. Farid, "Software engineering globalization and localization," in *Systems, Man and Cybernetics, 1990. Conference Proceedings., IEEE International Conference on.* IEEE, 1990, pp. 491–495.
- [3] J. M. Hogan, C. Ho-Stuart, and B. Pham, "Key challenges in

software internationalisation,” in *Proceedings of the second workshop on Australasian information security, Data Mining and Web Intelligence, and Software Internationalisation-Volume 32*. Australian Computer Society, Inc., 2004, pp. 187–194.