# Integrating Internationalization in the User-Centered Software Development Process

José A. Macías

Escuela Politécnica Superior, Universidad Autónoma de Madrid
Tomás y Valiente 11, Madrid 28049, Spain
j.macias@uam.es

**Abstract.** Internationalization is a common practice today in software development. In the most basic sense, internationalization is carried out by applying localization design guidelines to face language translation, icon representation, character sets and so on. However, this practice is mostly intended for design purposes, which results insufficient when applying internationalization in huge projects and, specifically, through a concrete development process. In this paper, a broader framework is provided in order to ensure internationalization through a software development process. To this end, a set of activities and sub-activities will be presented involving not only design but pre-development, analysis, implementation and evaluation issues that need to be considered for a right internationalization assurance in international software development. The idea behind is to bridge the gap between simple and usual localization activities and the user-centered software development process as internationalization assurance also helps increase the quality and usability of the software overall.

**Keywords:** Internationalization, Localization, User-Centered Development, Usability, Software Engineering, Software Process.

## 1 Introduction

In a broad sense, software internationalization is garnering increasing attention from both industry and academy. A great variety of approaches have been proposed during the last years to deal with international software development aiming at the localization of software for use in more than one country. In general, existing proposals are mainly based on recommendations and design guidelines that take into consideration different aspects such as language translation, icon representation, character sets, numbers and currency, time and measurement units, and so on [1-3]. However, international software development also requires other activities that should be conveniently considered and planned in order to systematically create international software with an acceptable level of quality. In fact, internationalization is not only related to the user interface of the application. Instead, producing international software involves different technical and non-technical activities in addition to those intended to design the user interface. This way, internationalization issues should be addressed not only in the design and implementation stages of the development, but also in

P.L.P. Rau (Ed.): CCD/HCII 2013, Part I, LNCS 8023, pp. 243–252, 2013.

pre-development, analysis and evaluation stages, applying concrete activities to increase the quality of the resulting software through a user-centered development process. Among others, the mentioned activities should address issues concerning a suitable internationalization plan, multidisciplinary team building and user-centered analysis activities based on user, task and context study in order to conveniently separate functional and non-functional requirements to produce successful internationalization goals.

This paper proposes an explicit integration of several internationalization activities through a development process intended to produce usable software. To this end, specific stages and a set of add-on activities and sub-activities, in addition to the ones existing in conventional development processes for usable software, will be provided. Proposed sub-activities are intended for pre-development –before software construction, analysis –to categorize functional and non-functional internationalization requirements, design, implementation, and also evaluation.

The paper is structured as follow. Section 2 provides related work. Section 3 reports on the main contribution of this paper, that is, the add-on activities and sub-activities for the internationalization assurance through the proposed development process. Section 3 presents conclusions and future work.

## 2 Related Work

Development of international software often requires more than just dealing with language-related issues. In fact, it involves special needs of other countries and cultures that have to be taken into account in the context of a software development model and not in isolation, as it usually happens. In general, most of the existing approaches only provide internationalization techniques to be applied when developing the user interface [1-3], or explicit relationships between cultural layers and design features in general product development [4]. However, little or no approaches have been proposed to integrate common internationalization user-centered activities with a software engineering process, which is the main strength of this paper.

Some existing works can be considered of interest to facilitate such integration. For instance, one interesting approach to obtain contextual information and requirements in early development stages is cultural models [5], [6]. A cultural model compares the similarities and difference of two or more cultures by using international variables to organize cultural data involving national cultures, corporate cultures, the cultural diversity of groups of users, international markets, etc. Cultural information is useful to obtain the context of use necessary for early cross-cultural development activities [7]. In fact, knowledge about patterns of usage [8] and typical user goals may differ radically from country to country, and this information needs to be capture in advance to drive the development and design effective usability tests [9].

Concerning design, there is a larger number of existing guidelines involving international concerns. Those facilitate detailed design and implementation of localized software and user interface prototyping. To cite a few, Microsoft specifies internationalization as a combination of world-readiness and localization [10]. World-readiness is a developer task, which enables a product to be used with multiple scripts and cultures (globalization) and separating user interface resources in a localizable format. On the other hand, the W3C Internationalization (I18n) Activity [3] includes a working group on addressing different languages, scripts, and cultures, and providing guidelines and specifications to ensure the internationalization of software. Other interesting works report on the icon and symbol design issues [11], the general impact of culture on the design of the user interface, specific Asian and Arabic concerns and the management and production of multilingual documents [1].

As for evaluation, some existing approaches can be considered useful to be integrated in a user-centered development process, such as the international inspection method [12] that involves having people from multiples countries inspecting the international software and analyzing whether it would cause any problems in their countries of origin. Similarly, the international usability testing [1], [2] involves real international users testing the software, or doing it remotely [13] to reduce costs. All in all, cultural difference affects user research processes, and thus cultural effects should be taken into consideration in user testing [14]

As for development processes, there are no specific approaches concerning internationalization. However, this paper proposes a development process model based on the usability engineering lifecycle ISO 9241-210 [15], which replaces the ISO 13407 [16]. In contrast to the ISO13407, the new ISO 9241-210 proposes requirements (not recommendations, as in the previous version) for human-centered design activities. Such requirements imply to understand and specify the context of use (including users, tasks and environments), specify the user requirements in sufficient detail to drive the design, produce design solutions meeting the requirements, conduct user-centered evaluations of these design solutions and modify the design taking account of the results. Additionally, the new standard is based on the user experience. This way, the process is iterative, the design addresses the whole user experience and the development team includes multidisciplinary skills and perspectives, which is well suited for international software development involving international stakeholders.

## 3 A Process Model for Internationalization

In order to tackle the development of international applications, the process model depicted in Fig. 1 has been proposed. It includes four main activity groups:

- Pre-development, addressing initial activities of the software project such as project planning, management and team building.
- Development, where development activities take place. This is based on the ISO 9241-210 iterative process, where the design solution is continuously evaluated and validated by the user after implementing the final system.
- Post-development, including activates related to installation, operation and support, and maintenance until software retirement.
- Integral, mainly related to general quality assurance and documentation activities that take place during the whole project.

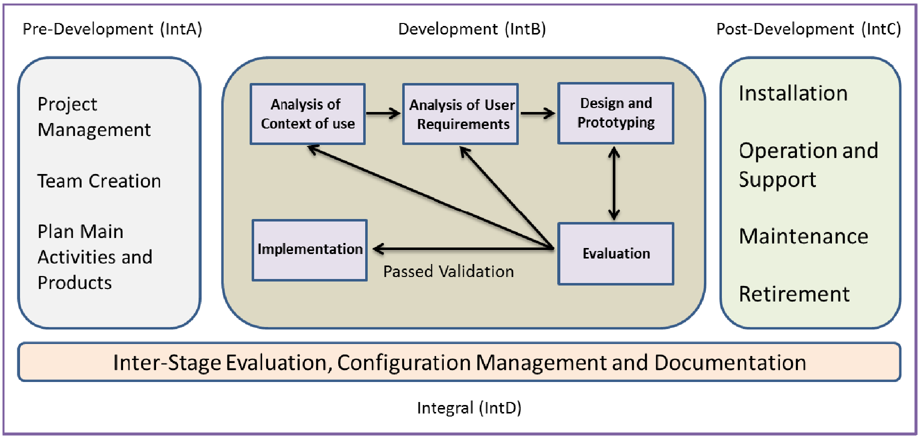

**Fig. 1.** Proposed process model as a framework for internationalization assurance

### 3.1 General Process Model Activities

Some of the activities outlined in the presented process model correspond to those usually specified in standards such as IEEE 1074-1997 [17], where activities are conceived in a development-centered paradigm mostly intended for software engineering. Such common activities are shown in Fig. 1 grouped in project management, including specific sub-activities such as project initialization, planning, monitoring and control. Similarly integral activities, mostly concerned with quality assurance, typically include software configuration management, documentation, and inter-stage evaluation. Also, post-development activities involve different usual sub-activities that can be considered in any project affecting software installation, operation and support, maintenance for correcting and producing new releases and, finally, the software retirement. Even though these mentioned activities and sub-activities are not specifically related to international software development, they have to be taken into account in the process model as international software development also involves different tasks needed to plan, control, install and maintain software.

### 3.2 Specific Activities Proposed for the Internationalization Assurance

Add-on activities proposed to ensure internationalization in the software development process are presented. To this end, proposed activities and sub-activities are given a clause, starting with IntA, IntB, IntC and IntD, to indicate pre-development, development, post-development and integral activities, respectively (see Fig. 1), and thus enumerate each activity correctly in the software process. As commented before, such add-on activities complement others concerning software development that should be also taken into consideration for a complete software development. This paper only highlights the principal activities mainly related to international concerns to carry through internationalization and usability assurance.

**Pre-Development Activities (IntA).** Table 1 depicts the activity grouping proposed for international software pre-development activities and sub-activities. Before software development, various initial concerns need to be considered. First, the use of quality standards should be stated according to suitable normative such as ISO 9241-210 to ensure usability and consider an iterative lifecycle with highly user participation (sub-activity IntA.1.1). Also, there must be specific procedures to manage the different international stakeholders involved in the project (sub-activity IntA.1.2) and communicate with them successfully. Besides, development team members should be selected according to specific internationalization requirements (sub-activity IntA.2.1) and assigned specific roles (sub-activity IntA.2.2). For instance, the development team needs someone who understands foreign languages and cultures and has a technical background. On the other hand, it is important to take advantage of the international nature of the development team [18] to carry out international inspection when necessary. Finally, an internationalization plan should be also defined at the beginning of the project for both usability and internationalization concerns (sub-activity IntA.3.1), including specific activities through a convenient project scheduling as well as intermediate products and evaluations (sub-activity IntA.3.2).

**Table 1.** Pre-Development Activity Grouping for International Software Development

| Main Activities | Sub-Activities |
|---|---|
| IntA.1. Project Management | IntA.1.1. Use of software quality standard for usability ISO 9126 and iterative lifecycle ISO 9241-210 |
| | IntA.1.2. Stakeholder management according to international requirements |
| IntA.2. Team Building | IntA.2.1. Members selection according to international concerns related to the project |
| | IntA.2.2. Assign specific roles to project team members |
| IntA.3. Plan Main Activities and Products | IntA.3.1. Plan internationalization user-centered activities |
| | IntA.3.2. Concrete internationalization user-centered intermediate products and evaluations |

**Development Activities (IntB).** Table 2 depicts the activity grouping proposed for international software development activities and sub-activities. First, a suitable user-centered development process should be considered in order to integrate internationalization activities with a software process that enables the development of usable international software. To this end, the ISO 9241-210 has been considered as a suitable framework to carry through the development of usable and international interactive software under a user-centered paradigm. The process model should be conceived as an iterative process, including highly user involvement in all development activities in order to ensure usability and internationalization at large. The proposed internationalization sub-activities include the analysis of context of use to obtain cultural variables and identify global information (sub-activity IntB.1.1), as well as the way the user works (sub-activity IntB.1.2), which can be achieved remotely (sub-activity IntB1.3). Also, contrasting information between team members and users is necessary under a user-centered approach (sub-activity IntB.1.4). In addition, analyzing specific international scenarios of use can report valuable information in the analysis stage (sub-activity IntB.1.5). Additionally, user requirements elicitation is necessary in order to find out and categorize functional and non-functional requirements. This way, the use of human-computer interaction techniques such as Personas [19], [20] to create specific international profiles (sub-activity IntB.2.1) can be useful. Most of the existing bibliography encourages interaction design guidelines including cross-cultural issues, but even through these design guidelines are used, specific software requirements should be elicited, those involving functional and non-functional internationalization requirements (sub-activity IntB.2.2). For instance, a calculation based on specific currency affects not only to non-functional aspects of the application (presentation), but also to functional ones concerning operations that must be carried out using a certain currency format. In addition, the use of visual techniques to represent and validate requirements is necessary as an input for the design stage (sub-activity IntB.2.3).

Concerning design activities, there have been a great variety of design guidelines and techniques that can be applied in this stage (sub-activity IntB.3.2), mostly related to language translation, icon representation, character sets, numbers and currency, time and measurement units and so on. However, it is necessary to specify some user-centered activities integrated with the process model, such as the evolutionary prototyping (sub-activity IntB.3.1) and the linking to other usability activities that can be related to internationalization ones (sub-activity IntB.3.3) .

With respect to evaluation activities, those should be also considered in order to provide quality and feedback to the iterative development process. To this end, the use of the cultural model previously selected can be useful to evaluate the effectiveness of the international user interface (sub-activity IntB.4.1). On the other hand, the international inspection (sub-activity IntB.4.2) and the international usability testing (sub-activity IntB.4.3) are suitable methods to have a real perception of the internationalization degree of the software. This can be achieved using remote usability testing with users living in their respective countries, which reduces testing costs. Also, intermediate products should be verified in order to ensure the resulting quality (sub-activity IntB.4.4). This happens at all levels, for instance, the translation of all literal strings – i.e., translation verification testing, normally conducted by a person who knows the target language very well. Finally, the validation by international users (sub-activity IntB.4.5) in all products should be considered prior to the implementation stage.

As for implementation activities, aspects concerning the programming language (sub-activity IntB.5.1) should be considered to optimize the code and meet the localization requirements. For instance, fonts and font sizes should be appropriate in the target language. Also, string size is important for the translation into the target language. Same concerns should be considered for writing bi-directional text, display images with localized text, check environment variables such as date, time zone, currency, etc. In general terms, verified internationalization libraries should be used in the implementation stage (sub-activity IntB.5.2) whenever possible to facilitate programming and increase the quality overall. Finally, a suitable international user documentation of the final product must be written and made it easy to understand according to the target audience (sub-activity IntB.5.3).

**Table 2.** Development Activity Grouping for International Software Development

| Main Activities | Sub-Activities |
|---|---|
| IntB.1. Analysis of Context of Use | IntB.1.1. Select a suitable cultural model to identify global information, cultural bias, metaphors and the degree of localization necessary |
| | IntB.1.2. Use of contextual inquiry to understand the user's work style and process |
| | IntB.1.3. Use of remote techniques to capture the user's interaction style by monitoring his/her behavior |
| | IntB.1.4. Check interpretations between international development team and the user |
| | IntB.1.5. Specify international scenarios of use |
| IntB.2. Analysis of User Requirements | IntB.2.1. Use of human-computer interaction techniques, such as Personas, to create specific international user profiles |
| | IntB.2.2. Splitting of functional and non-functional requirements according to international issues |
| | IntB.2.3. Use of visual techniques to represent and validate requirements with international users and team members |
| IntB.3. Design and Prototyping | IntB.3.1. Evolutionary prototyping focusing on localization issues |
| | IntB.3.2. Use of international design guidelines and existing international user interface design standards |
| | IntB.3.3. Linking usability and internationalization design activities |
| IntB.4. Evaluation | IntB.4.1. Use of the selected cultural model to evaluate the effectiveness of the international user interface |
| | IntB.4.2. International inspection evaluation method |
| | IntB.4.3. International usability testing |
| | IntB.4.4. Assessment of intermediate products |
| | IntB.4.5. International user validation |

**Table 2.** (Continued)

| | |
|---|---|
| IntB.5. Implementation | IntB.5.1. Use of character encoding and environment variables suitable for localizing the final product |
| | IntB.5.2. Use of internationalization programming libraries whenever possible |
| | IntB.5.3. Produce suitable multilingual documentation describing the use of the final product |

**Post-Development Activities (IntC).** There are not concrete add-ons involving internationalization sub-activities for this group of activities. As it mostly depends on the type of software, it would be convenient, however, to facilitate the software installation and distribution to foreign countries, provide international users with the right formation to operate the software and carry out the similar aforementioned sub-activities in the maintenance stage when a change occurs or when a new software release is planned.

**Integral Activities (IntD).** Also in this case, there are not concrete add-ons involving internationalization concerns. It is worth pointing out, however, the importance of inter-stage evaluation to ensure a right process model sequence, as well as the inspections and revisions needed to check the process model's inputs and outputs and the documentation of the project in order to improve its testability and maintainability.

# 4 Conclusions

Software internationalization is an important necessity today. In fact, applications should be developed paying attention to international aspects to be broadly used around the world and avoid the digital divide. This greatly improves the usability of the software overall. On the other hand, a great number of international sales depend on the international usability of products. Companies often have a very large proportion of their sales outside their own country. This is the reason why specific activities and a concrete process model should be provided, in order to ensure the quality of the international software development. This research is concerned with the idea of developing international interactive software, but also with providing a usability and *internationality* assurance at the same time. To do so, this paper presents a framework to integrate specific internationalization activities in a user-centered development process, which is the main strength of this research. Specifically, this research suggests concrete activities and sub-activities to plan and develop international applications, including pre-development activities before software construction, analysis activities to study the context and categorize functional and non-functional requirements involving internationalization, design and implementation issues, and also evaluation concerns that should be considered to ensure and provide feedback in the development model proposed based on a iterative usability engineering process ISO 9241-210.

As future work, the next steps to follow will consist in refining some sub-activities and incorporate others. In fact, it is planned to specify, for each internationalization sub-activity, concrete tasks, techniques, products and supporting tools in order to fully prescribe the process model. Another milestone is formally defining the *internationality* quality attribute by specific metrics to be measured throughout the whole software process and thus systematically control the internationalization assurance.

**Acknowledgements.** This work has been supported by the founded projects TIN2011-24139, S2009/TIC-1650 and TIN2011-15009-E.

## References

1. del Galdo, E., Nielsen, J.: International User Interfaces. John Wiley & Sons, NY (1996)
2. Nielsen, J.: Usability Engineering. Morgan Kaufmann, San Francisco (1993)
3. W3C Internationalization (I18n) Activity web page, `http://www.w3.org/International/`
4. Hsu, C., Lin, C., Lin, R.: A Study of Framework and Process Development for Cultural Product Design. In: Rau, P.L.P. (ed.) Internationalization, Design, HCII 2011,. LNCS, vol. 6775, pp. 55–64. Springer, Heidelberg (2011)
5. Hofstede, G.: Culture and Organizations: Software of the Mind. McGraw-Hill, NY (1991)
6. Hoft, N.: International Technical Communication: How to Export Information about High Technology. John Wiley & Sons (1995)
7. Sun, H.: An Activity Approach to Cross-Cultural Design. In: Aykin, N. (ed.) HCII 2007, Part I. LNCS, vol. 4559, pp. 196–205. Springer, Heidelberg (2007)
8. Schadewitz, N., Jachna, T.: Introducing New Methodologies for Identifying Design Patterns for Internationalization and Localization. In: Aykin, N. (ed.) HCII 2007. LNCS, vol. 4560, pp. 228–237. Springer, Heidelberg (2007)
9. Siegel, D., Dray, S.: Contextual User Research for International Software Design. In: Aykin, N. (ed.) HCII 2007, Part II. LNCS, vol. 4560, pp. 266–273. Springer, Heidelberg (2007)
10. Microsoft Internationalization for Windows Applications, `http://msdn.microsoft.com/library/dd318661`
11. Marcus, A.: Graphic Design for Electronic Documents and User Interfaces. Addison-Wesley, Reading (1992)
12. Nielsen, J., Mack, R.: Usability Inspection Methods. John Wiley & Sons, NY (1994)
13. Nieminen, M., Mannonen, P., Viitanen, J.: International Remote Usability Evaluation: The Bliss of Not Being There. In: Aykin, N. (ed.) HCII 2007. LNCS, vol. 4559, pp. 388–397. Springer, Heidelberg (2007)
14. Lee, J., Tran, T.-T., Lee, K.-P.: Cultural Difference and Its Effects on User Research Methodologies. In: Aykin, N. (ed.) Usability and Internationalization, Part I, HCII 2007. LNCS, vol. 4559, pp. 122–129. Springer, Heidelberg (2007)
15. ISO 9241-210: Ergonomics of Human-System Interaction. Part 210: Human-Centered Design for Interactive Systems. International Standards Organization (2010)

16. ISO 13407. Human-Centered Design for Interactive Systems. International Standards Organization (1995)
17. IEEE Standard for Developing Software Life Cycle Processes, IEEE Std 1074-1997. IEEE Press, New York (1997)
18. Nielsen, S.: Top Management Team Internationalization and Firm Performance. Management International Review 50(2), 185–206 (2010)
19. Cooper, A., Reimann, R., Cronin, D.: About Face 3.0: The Essentials of Interaction Design. Wiley Publishing, Indianapolis (2007)
20. Guo, F.Y., Shamdasani, S., Randall, B.: Creating Effective Personas for Product Design: Insights from a Case Study. In: Rau, P.L.P. (ed.) Internationalization, Design, HCII 2011,. LNCS, vol. 6775, pp. 37–46. Springer, Heidelberg (2011)