

Adapting Lightweight User-Centered Design with the Scrum-Based Development Process

Degif Tekla
Faculty of Informatics
Hawassa University
Hawassa, Ethiopia
degiftk@gmail.com

Yvonne Dittrich
Department for Computer science
IT University of Copenhagen
Copenhagen, Denmark
ydi@itu.dk

Mesfin Kifle
Department of Computer science
Addis Ababa University
Addis Ababa, Ethiopia
Kiflemestir95@gmail.com

ABSTRACT

User-centered design (UCD) ¹ addresses the design of interactive systems placing the users in the center of the design with the aim of improving usability and user experience. Developing economies are in dear need of UCD; low IT literacy, low infrastructure and funds; and heterogeneity in culture and livelihood result in special requirements on usability in order to harvest the possible benefits of IT. Traditional UCD methods, however, are often regarded as heavy-weight and expensive. Agile software development methods are light-weight, flexible and iterative in order to accommodate the changing requirements and unsure funding and are therefore important for IT companies in developing economies. Can we adjust UCD methods to fit the need of developing economies and with agile development while taking advantage of the iterative character of agile development methods? The research appropriated an action research approach called Cooperative Method Development (CMD). Based on the empirical investigation, UCD challenges were identified, innovative use of light-weight UCD methods was deliberated and implemented. The improvements include: working with local IT personnel, light-weight and incremental use of personas, support departments performing acceptance testing on release versions, culturally adapted user testing in pairs and heuristic evaluation as adapted UCD practices. The evaluation together with the involved practitioners shows improvements in the development process including reduced reworks; satisfied users; better collaboration with stakeholders; and a close understanding of users and their needs. The evaluation of the resulting integrated approach with the involved practitioners as well as

software engineers not involved in the research indicates that the results are transferable to similar contexts.

KEYWORDS

Lightweight UCD, Agile Development, discount usability methods, personas, heuristic evaluation, user-pair test

1 INTRODUCTION

Software engineering in developing countries faces additional usability challenges compared to developed countries: complex socioeconomic conditions, the lack of infrastructure, lack of skills and experience and the cultural heterogeneities in developing economies need to be considered [1]. Developing economies prioritize their budget for health, nutrition and other development activities that for IT projects, especially in the rural areas there is poor availability of electricity and mobile network for using mobile applications that are potential services to address the disadvantaged community. The hierarchical culture and cultural uncertainties that are more prevalent in these regions influence users from interacting and participating in the development activities. Some of the challenges existing in such countries and guidance for information and communication technology (ICT) design has been documented in the real access/real impact document that discusses the digital divide [2]. The impact of digital divide is not only limited to the one existing between developing and developed economies but it also includes the differences in the rural and urban areas within the same nation.

In developing economies ICT services in many cases are leapfrogging paper-based administration [3]. Especially in rural areas, part of the population is illiterate or semi-literate and does not understand the advantages of ICT services. Furthermore, most of the developing economies have a higher socio-cultural diversity than developed economies and often several languages are spoken in different countries. Addressing these usability challenges is of core importance for the developing economies to make use of the improvements promised by the use of IT services.

Agile software development methods are iterative and incremental software engineering methods that aim at

¹ Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
SEiA '18, May 27–28, 2018, Gothenburg, Sweden
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-5719-7/18/05\$15.00
<https://doi.org/10.1145/3195528.3195530>

working software, customer collaboration, flexible to change of requirements even late in development and fast delivery of prototypes to meet marketing requirements. Agile software development methods are adapted and used in the developing economies as they provide flexibility to accommodate changing scope and funding [3-4]. Due to lack of experience and skills, users cannot articulate the requirement of future products from abstract interfaces [1], and hence the flexibility of agile methods bring potential benefit in these areas. While the values and principles defined in the Agile Manifesto [5] are important to software development, agile methods focus on functional requirements but do not address usability explicitly. The right functionality, however, may miss the main target of development projects due to the lack of usability [6]. As agile methods are lightweight, adapting and integrating lightweight UCD methods can improve usability in agile software development.

User-Centered Design (UCD) is an iterative design process that focuses on user research, user interface design and usability evaluation to produce useful and usable software. The ISO 9241-210 defines UCD in terms of human-centered design as: 'approach to systems design and development that aims to make interactive systems more usable by focusing on the use of the system and applying human factors/ergonomics and usability knowledge and technique' [7]. The norm as well as related methods put emphasis on the inclusion of end-users as well as other stakeholders who can contribute domain knowledge and important usability requirements.

UCD and Agile development already share some characteristics: The quick iterations in agile methods can be seen to support UCD as they allow evaluating a design quickly and get early feedback. The onsite customer in eXtreme Programming (XP) and the product owner in Scrum act as customer representatives and represent end-user perspectives in the project. Though, they cannot substitute the involvement of end-users [6], the existence of these roles clearly places the responsibility regarding user requirements. There are two challenges for UCD in agile development for developing countries: a.) Traditional UCD methods, however, are often regarded as heavy-weight and expensive. Many software organizations – in developing economies [4] as well as in developed economies [8] – do not consider even to evaluate the usability of their products as a start for UCD. Can we devise light-weight UCD methods that can be used in a 'by need' fashion in agile development? b.) While UCD provides an array of important methods for designing for user experience and usability, traditional UCD methods lack consideration of complex socioeconomic conditions, the lack of infrastructure and the cultural heterogeneities in developing economies.

Accordingly, the research question is: How can we integrate lightweight UCD and agile development to support a more sustainable approach of software development in developing economies? The paper aims at understanding the software development practice and its challenges in the local situation

and finding a place for adapting and localizing lightweight UCD and usability practices that could be performed within the specific context.

The following section presents the related work. Section 3 introduces the research methods applied. Sections 4, 5 and 6 present the relevant empirical findings. Section 7 discusses these findings in light of the related work. The conclusion in section 8 presents our answers to the research question.

2 RELATED WORK

Agile development focuses on satisfying customer needs and business requirements. Customers are rarely end users [6, 9]. A customer i.e. onsite-customer in XP or the PO in Scrum may be a domain expert but might not use the software at all and may not clearly know the need of end users [10]. Furthermore, there are studies that indicate that customer representatives in agile development can become too exposed to the inner workings of the software due to the daily contact with developers and may not truly represent the actual users [10-11]. What the users say they want and what they actually need might be two different things as it has also been reported in a user involvement and usability evaluation survey [4]. One of the guidelines for UCD is "watch what users do, not listen to what they say" [12, p. 5]. However, clients or customer representatives are usually taking design decisions, driving many HCI design and usability considerations influencing the design process even if such a person may not be a user.

Discount usability evaluation methods have been first introduced by Jakob Nielsen [13]. Kane proposed to integrate these with agile methods [14]. Sohaib and Khan [15] developed a theoretical framework for integrating discount usability with XP. The four discount usability methods proposed by Nielsen are prototyping, heuristic evaluation, simplified think-aloud protocol and card sorting [14]. These discount usability methods are indeed economical and light-weight for usability evaluation and adding usability in agile methods but need to be tailored to both agile methods and the use context.

Software development, especially for rural projects, is affected by over-specification possibly because of the distant situation, cultural differences between the developers and users, and the less IT skills and experience of users. The development teams are less familiar with the context of such users and the product and customer relationship managers usually want to have a control on the user interface.

Research on targeted software development methodologies to ICT for development (ICT4D) projects is relatively sparse. Few articles in this area report that established and western developed and publicly available methods do not work to the specific situation of developing economies for the ICTD projects in these areas [1, 16-17]. Doerflinger and Dearden [18] suggested adaptation of existing methods to the context

to increase stakeholder and user involvement. Research on ICT4D states that ICT projects should be community driven, initiated with a pilot project in the community, and iterative and incremental in nature [19]. This implies that UCD and usability are critical for the sustainability of ICT projects.

Project management process and skills are very important for the successful implementation of improvements in the development process. Among the challenges of local software organizations reported in Ethiopia is the lack of project management skills over the high demand of ICT products and outsourcing of large-scale projects especially public projects in the local context [17]. One of the factors influencing software organizations to follow flexible software process models like agile development is the software procurement process. Also reported in Ethiopia is that the software procurement process is very rigid and limits software organizations from using flexible development processes that allow for usability evaluation [20]. However, organizations are adapting prototyping, agile development and incremental development processes to the project management process despite these constraints [20-21]. Furthermore, the call for tendering (CFT) document lacks considering usability requirements to support organizations do usability activities [4]. The lack of support for software organizations from the policymakers in motivating them do usability activities is not only limited to developing economies but it is also a challenge in the developed economies [8].

3 RESEARCH METHOD

This report is part of a PhD thesis. The empirical study has been setup with two software organizations labeled Org. A and Org. B located in Addis Ababa. The focus here is on the first project (proj. A) among the three projects studied in Org. A and a project in Org. B. The other two projects used in Org. A are not detailed here.

Proj. A is composed of two sub-projects for different clients but sharing the same code base. It is a rural mobile application project meant for digital supply of collecting crop from farmers and input supply to member farmers. The research resulted from participatory observation from the first phase of proj. A are published [3]. The project in Org. B is a contract project whose client is a large public organization that works in judiciary and document authentication activities in its thirteen branch offices in Addis Ababa.

For the deliberation and introduction of UCD practices, it requires a closer study and understanding of the organization's development practices, working with the practitioners from inside the organization. Through such cooperation, UCD methods can be adapted to the organization's particular resources and circumstances. The Action Research (AR) approaches like the cooperative method development (CMD) [22] are recommended for empirical studies to bring changes in the industrial settings working in close collaboration with practitioners. CMD is an AR approach

appropriated to software engineering by Dittrich et al. [22]. It is based on mainly three phases: understanding the practice, deliberation of improvements and implementation and evaluation of improvements performed cyclically as shown in figure 1. The adaptation of CMD made here in this research is the initial exploration study to guide setting the empirical research and further the cycle from phase 3 to phase 2 where a further understanding of the practice is not important that also helped to work with the agile time-boxed iterations.

The data collection methods used include interviews, participant observation, field visit of users, meetings with practitioners, discussions with users and practitioners, workshops with practitioners and users and surveys. The qualitative data collected using the instruments have been analyzed following qualitative thematic coding techniques similar to the grounded theory approach [23].

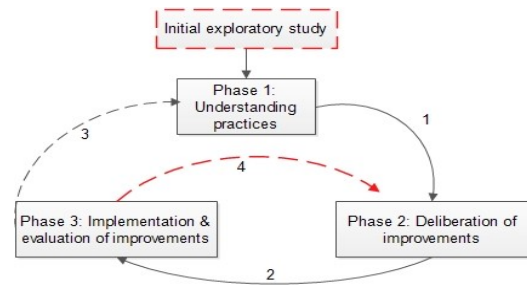


Figure 1: The CMD action research cycle.

4 THE DEVELOPMENT PROCESS

Scrum like agile development process has been adapted to the project management activities in both cases. Org. A adapted the process in 2014 and Org. B in 2015. The project in Org. B is an update and modification of an old version of a product developed by the organization due to the need from the client side. However, the proj. A is a new project and the clients and users were new to the developers. Both case organizations use similar development approach, adapted Scrum development process and the development process is shown in figure 2. The difference from the publicly known Scrum is that for example in Org. A standup meetings were not regular, though it takes 10 to 15 minutes the schedule varies as it may start at any time between 9:00 to 11:00 mornings. Standup meetings are not observed in Org. B. The product owner (PO) in Org. A was from customer representatives in the supportive departments while the PO in Org. B is one of the technical development team members who is a junior programmer.

The pregame phase includes planning and high-level design. Informal communication and meetings with stakeholders and users, task observations, and experience surveys with the domain people are also activities of the pregame phase. The focus on the phase is to gather requirements, high-level architectural design and setting up business plans.

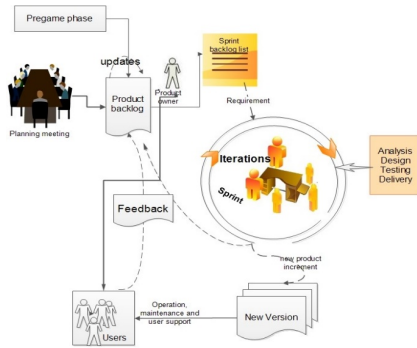


Figure 2: The Scrum process in the case organizations.

The pregame phase results in initial backlog items, followed by prioritizing the items for a sprint and developing user stories. User stories are assigned to each member of the development team and the PO and the scrum master (SM) who is also known as the project manager guide the team and follow the sprint. Analysis, design and development occur in a sprint that results in a potential release version. Upon the end of a sprint, tasks that are deferred to the next sprint are taken back to the backlog for prioritization and development continues. Depending on the decision of the management, the developers, the PO and the SM the prototype at the end of a sprint might be shipped for deployment at the customer site.

5 THE RESEARCH PROCESS AND RESULT

The report in this section is presented according to the projects taken in the case organizations. Initial interviews with two software practitioners, a software engineer and a project manager in each of the case organizations, have been performed to explore the practices and challenges and to start up the action research collaboration as also represented in the adaptation of the CMD as in figure 1. These initial interviews helped to develop close collaboration with the practitioners for the empirical research following the initial study.

5.1 Project A in Organization A

The project is composed of two sub-projects. These projects develop special purpose applications of mobile services connecting rural communities to the capital. They are designed for two different organizations, with different customers and users, geographically separated, speaking different local languages and culture. One of the application is for the northern region of Ethiopia where the spoken language is Amharic, while the other is developed for the southern region of Ethiopia where the spoken language is Afan-Oromo. However, the two projects have similarities in the application domain and, to a large extent, share the same code base. The project team is composed of three software engineers, one scrum master (SM), a project manager, one product owner (PO) and two operational personnel one for each sub-project.

Part of this project with other follow up project in the same organization has been published in the article [3] [24].

5.1.1 CMD phase 1: Understanding the practice. Close collaboration with developers, the PO, product operations, marketing people and users helped to understand and identify the challenges through observation, field visits and interviews. The close and participant observation of the first author with the company practitioners resulted in unfolding the practices and challenges: the PO developed the UI, met frequently but usually unscheduled with the developers, and reviewed sprints. We observed that, due to her frequent contact with developers, she seemed to become influenced by the development perspective; and end user representation was put at risk. The developers were observed to perform acceptance testing themselves for delivery which usually results in self-referencing rather than testing how the product works for users. Furthermore, the observed challenges include different personnel talking about users and their needs differently to developers, low IT skill of users, and cultural and language difference between software developers and users. The users speak the local language, 'Afan Oromo' and developers do not understand the language. The users do not like to give direct critique to developers on their challenges of operating the application. They consider it as their failure not to operate the application and also consider that it is not ethical to critique technical people. Uncertainties with respect to technologies by users has been observed as the farmers fear to provide their profiles to be recorded in the application. Additionally, developers moved from one project to another before completing projects and participated in parallel projects.

Field visits and user research by the first author who accompanied product operations to the rural site unfolded the challenges of the proxy users (IT supporters operating the application on behalf of the farmers) and uncertainties of the farmers related to the use of the mobile application to record their crop yield with their details. In a meeting with the stakeholders, a farmer asked 'I feel that the government may use a record of my annual production to charge additional income tax'. Other difficulties of users: language of the UI, lack of features for prioritized data for generating report by the users, reentering farmer details at each crop record, and the poor and intermittent mobile network that jeopardized timely sending of data on collected crop to the central server.

5.1.2 CMD phase 2: Deliberation of improvements. Two workshops were held with the involved practitioners during the project. The first was to prioritize the identified challenges. The second workshop has been used to propose improvements. Based on the identification and prioritization of the challenges together with the practitioners, the improvements proposed include the use of personas and the working with local IT personnel to mediate the differences between developers and users. Personas were developed to address the distance between urban developers and rural

users, the differences in culture, language and economic situations. Furthermore, in the deliberation of personas the developers complained that different personnel in the software organization communicated different user requirements to developers, and as a result of the situation developers seek single point contact for users' requirements. To substantiate this for example, a developer in the deliberation workshop said, 'we need single point of contact regarding users' requirements'. The other deliberation is the arrangement of support departments to do acceptance testing to support developers get early feedback.

5.1.3 CMD phase 3: Implementation and evaluation of improvements. Field visit of users and interviews with users have been used to develop user profiles. Personas have been developed lightweight and used by the PO, developers and support departments doing the acceptance test. A sample persona developed during the project is shown in figure 3. Personas were observed to be helpful and used by the practitioners to refer user needs and constraints. However, persona usage was not evaluated as it is not continuously updated and worked with due to project termination.



Debela (user persona)

Occupation: IT Operator

Gender: Male

Age: 24 years

Education: Debela has Diploma (12 + 3) in accounting. He is a young graduate and has no professional experience.

Language: less skills in using English and Amharic languages but fluent in Afan Oromo which is a working languages of Ethiopia and Oromia region.

Technology: He is eager to know technology but has less experience of using touch phones and has less IT skills. He has learnt basic computer application from a common course in his college. He learn from his friends in operating IT products and for some internet services from the touch phone.

Debela is new to the product (supply chain) and digital payment system. He is to use the application in rural market setting for 4 to 6 hours especially on Tuesday and Saturday (local market days).

Motivation:

The main motivation of using the specific product is in getting salary but also supporting farmers in his area who are illiterate and semi-literate where they have problem of utilizing IT services.

Figure 3: Sample persona for proj. A

Support departments from marketing, and product operations including the PO helped by doing acceptance tests to help developers get early feedback on release versions. Besides personas, local IT personnel has been taken as a role to bridge rural end users and urban developers and mediate the heterogeneities, the less IT skill and experience of the end users and users' uncertainties. The new stakeholder interaction is shown in figure 4. Fast delivery of working prototypes helped support departments do acceptance testing and collect user feedback. Understanding the challenges closely helped to take corrective measures by developers such as offline storage of data and forwarding techniques to alleviate the intermittent electricity and the unstable and low network bandwidth.

The project was terminated before completing the planned works such as automating the payment system for cashless transactions due to lack of funding and lack of interest on the continuation of the project on the client side.

Due to the short duration of the case projects in Org. A, the experience developed in proj. A has been used in later two projects of the same organization as the projects took place sequentially one after the other. These two later projects in Org. A that are not detailed here confirmed the use of personas but they are also explored for adaptation of discount usability methods: usability heuristics and usability testing with user-pairs as a cultural adaptation of the simplified think-aloud protocol.

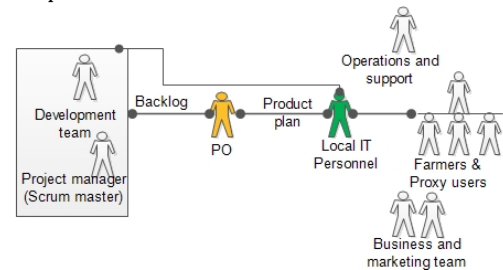


Figure 4: Stakeholders and their interactions in proj. A.

5.2 Project of Organization B

The second project detailed here is a redesign of an existing program and adding new functionalities. The redesign of the project is started since March 2016. The team for the project is composed of four members, a project manager, two software engineers and a junior programmer who also works as a customer relationship manager and user support.

The team adapted a Scrum-like process, tailoring to the project management activities to fit the CFT and the contract as in Org. A. Most of the projects in this organization are public and contract projects. The publicly known scrum ceremonies and artifacts such as stand-up meetings are not commonly scheduled tasks. The client of the project is a large public organization which has thirteen branch offices located in Addis Ababa. Most of the services provided in the client organization are related to document authentication and registration, authenticating power of attorney to court cases, transfer of ownership, authenticating car and house sales and other services not documented here. The services automated during the research period are authentication of sales cases, transfer of ownership and authentication of agency.

5.2.1 CMD phase 1: Understanding the practice. The study unfolded the challenges in the project such as 'user resistance', 'lack of understanding of the complexity of the tasks of the users by the developers', 'unstable and low network bandwidth', 'lack of IT skills of users', 'too much rework for developers', 'users do not understand their needs', 'turnover of employees in the client organization', 'users do not like to give critics on their usage of the system directly to developers'. Some of the usability challenges documented are common usability problems that include: 'when the user traverse back to the previous page he/she is not able to get the filled records', 'clicking a drop-down button takes the user

to the server that added latency’, ‘inconsistencies from what the system produced and what the user knows’.

5.2.2 *CMD phase 2: Deliberation of improvements.* The experience gained in proj. A and other projects of Org. A has also supported in proposing improvements in this project. Deliberation meetings resulted in the prioritization of the challenges and proposition of improvements. Local IT personnel hired by the client organization to support users and mediate between end users and developers is among the deliberations to closely support users. Personas, adapted simplified think-aloud protocol, usability heuristics and prototypes were among the proposed improvements.

5.2.3 *Implementation and evaluation of improvements.* Field observation of users and interviews with users have been used to develop user profiles. Personas have been developed from the profile and to be lightweight with the help of local IT personnel, and customer representatives in Org. B. However, after the first release developers claimed to know their users long before in their previous projects. As a result personas are not updated anymore.

Local IT personnel successfully mediated between the end users and developers. These local IT supporters have a university degree in the fields related to IT and computer science. They effectively support users communicating their challenges to developers. End users of the application were more comfortable to talk their difficulties and their critics to these local supporters than providing direct criticism to the industry people. In addition to providing local IT support services the critical feedback of this local IT personnel to software engineers include ‘providing Amharic fonts that are compatible to the system of the user’, ‘automating error reporting mechanisms to lower user frustration’, ‘closely understand users day to day task to make the application support their tasks’. The other improvements implemented include lightweight usability methods: user-pair testing, an adapted version of simplified thinking-aloud protocol and heuristic evaluation of UIs and prototypes.

Evaluation of improvements: It has been observed that there are many matching cases on the results of usability testing using user-pair testing and heuristic evaluation of the UI. Both usability testing methods show similar results because the user-pair testing has been performed following the heuristic evaluation before the developers are acting on the result of the heuristic evaluation due to deployment decisions. Some of the matching results between the two usability tests are shown in table 1. The heuristics H1, H2, H5, H6, H8, H9 in table 1 are from the list of heuristics adapted from the thirteen usability heuristics discussed by Pierotti [25]. H1 stands for the heuristic ‘Visibility of system status’, H2 stands for ‘Match between system and the real world’, H5 stands for ‘Help users recognize, diagnose, and recover from errors (Error messages should be expressed in plain language)’, H6 stands for ‘Error prevention’, H8 stands for

‘Flexibility and minimalist design’ and H9 stands for ‘Aesthetic and minimalist design’. However, the match also confirms that heuristic evaluation plays an important role in identifying usability problems that could be addressed early before deploying the product to the customer.

Table 1: Frequency of Special Characters

Heuristics	Sample Result of Heuristic Evaluation and Violated Heuristic	Sample Result of User-pair Testing for Usability Test
H5; H9	Error prevention; Error messages should be expressed in plain language	System error generated when recording customer details for attorney that the user could not understand and neither by the IT supporter
H8	The time it takes to acquire a target area is longer	There is too big a gap between the command buttons such as Next, Cancel and data entry fields on a page
H2	A Thousand separators are needed to be entered manually, the system does not give automatic formatting of such field	The system does not support the user in formatting monetary values using a thousand separators
H1; H6; H8	Inappropriate filters and non-relevant data	There are items listed in the attorney detail registration page that are not relevant to the selected category

In addition to the local IT personnel, culturally adapted user-pair testing as a solution to the cultural influence of uncertainties by the users helped in unfolding usability challenges that are not straightforward. Practitioners who have only basic knowledge of usability are able to use heuristic evaluation using checklists for testing the web application and providing early feedback to developers.

6 THE INTEGRATED APPROACH

6.1 The Integrated Approach of Lightweight UCD and Scrum

User personas were developed to be light and lean to fit the Scrum process. Personas and prototypes are among the artifacts adapted to the lightweight UCD method in the development process in addition to the deliberation on roles including working with local IT personnel to bridge the differences in culture, language and IT skills and distance between rural users and urban developers and support departments doing acceptance testing to support developers getting early feedback. Heuristic evaluation using checklists and user-pair testing for culturally adapted simplified think-aloud are among the discount usability techniques applied. The adapted lightweight UCD methods with the Scrum development process into the project management process of the organizations studied is shown in figure 5.

Figure 5: The integrated approach.

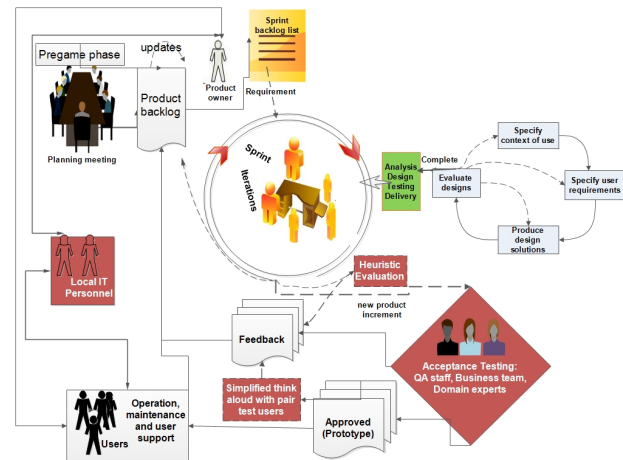
6.2 Evaluation of Improvements

In addition to the evaluation of the implemented practices at each of the CMD phases, evaluation with the selected and involved practitioners has been held based on interviews. The result has been analyzed using qualitative open coding and shows improvements in the development process including 'reduced reworks', 'satisfied users' and 'better collaboration with stakeholders' and 'close understanding of users and their needs'. The integrated approach is further evaluated by developers and project managers who are not involved in the projects following similar interview technique but using a different guideline. The evaluation with these external practitioners also confirms some of the evaluation points of the involved practitioners and possible application of the approach in organizations with similar context.

7 DISCUSSION

ICT policies and national standard documents such as CFT documents need to encourage software organizations to perform usability evaluation. The challenges documented including, shortage and lack of skilled professionals, switching developers from one project to another and working on multiple parallel projects are in line with an earlier research undertaken by Biru [17]. Biru reported that the software development in Ethiopia is characterized by large gap between the demand and supply sides. There is high demand and large projects in the local standards are outsourced but the supply side has been characterized by lack of skills and unrealized benefits, inadequate educational and training support infrastructure and absence of national standards or guidelines.

Taking software engineering process to the case of ICT for development, especially the communication difficulties between software developers, usually residing in urban areas, and farmers in rural areas as observed in proj. A have specific and unique challenges due to the distance, skill and language differences. Socioeconomic challenges such as lack of funds make the governments in developing economies prioritize their budget for development activities that IT projects. In proj. A lack of infrastructure such as intermittent electricity and the low network bandwidth has been acted by developing an offline store and forward service. The study here also confirms the research finding of Maunder *et al.* [1] and Doerflinger and Dearden [18] for the need to adapt UCD and software development methodologies. The digital divide between the urban and rural areas need special attention in software development projects in developing economies. It also needs a means of creating closer collaboration with project members and make users involve and participate to influence the development practice to make the development projects have a sustainable impact. Practices from participatory design as one of the UCD approach could assist



in addressing the problems which also needs support like funding in the development projects for a long-term engagement with these users.

The software practice and usability challenges studied in companies in Ethiopia as a case: tight contracts and short-term projects, developers have to work with multiple projects sometimes in parallel may have impacts in the quality of the development projects. One reason for this organizational practice is due to shortage of trained and skilled professionals as it has also been unfolded during interviews in the survey [4] and in the empirical research in the case projects. This has created specific difficulties to connect to the end user. It is also a common practice for local software organizations to outsource projects to external companies due to lack of trained professionals and due to the larger scope of the projects in the local context.

ICT policy failing to have appropriate software procurement and failing to motivate to work on user-centric issues and usability, taking, for example a failure to consider explicit usability requirement in the CFT documents may limit software organizations and stakeholders from taking improvement actions.

In this research a repository of lightweight methods for UCD in the lifecycle including UCD roles for the iterations and activities in the development process has been implemented. Local IT personnel and personas bridge between end users and developers. Based on the empirical investigations, local IT personnel bridge the heterogeneities and communication gaps between end users and developers and this became more visible for the case of the rural project. Personas are more important to developers and practitioners when the users are new to the developers. However, the use of personas may not be effective when the practitioners are familiar with their users as verified in the project of Org. B. The work of Singh [11], shows to support the conventional PO with a second PO who is a usability professional. However, different projects have different contexts. Within the context studied here, the PO is supported by the local IT personnel and personas.

Local IT personnel as mediating the differences between users and developers is an important UCD role in developing economies. However, this may not be a long-term solution as with developments in skill and advancements in technology.

The use of workshop as one of the methods used for usability and UCD is discussed by Jia et al. [26]. Workshops are important tools to engage stakeholders and users in the development projects. In a collectivist society like that of Ethiopia [27], workshops are dynamic tools to get collective mind solutions. In this research workshops have been intensively used as part of the research method CMD with the involved practitioners as well as with the users and other stakeholders including clients.

UCD methods and methods supporting UCD activities need to be contextualized to the context. In the case projects in this research, the hierarchical cultural context and uncertainties of users as indicated in [4, 27] and also observed in the cases informed for deliberation and implementation of the role local IT personnel and adaptation of usability testing with user-pair testing.

The lightweight UCD methods implemented here including prototypes and personas are supported by the adapted discount usability methods, user-pair testing and heuristic evaluation. Discount usability methods are proposed to be integrated with agile methods for improved usability [14-15]. However, as with the research here discount usability methods need to be tailored to the context to help for effective evaluation of usability.

8 CONCLUSION

The empirical investigation has helped in answering the research question 'how can we integrate lightweight UCD and agile development to support a more sustainable approach of software development in developing economies?' During the empirical research, specific challenges of users, practitioners and limitations in the development process have been identified based on participant observation, field study of users and practitioners and interviews. The identified challenges have been prioritized and improvements have been deliberated. Meetings and workshops are used to deliberate improvements. The improvements include local IT personnel mediating between end users especially rural end users and urban software engineers, however, working with local IT personnel has also brought important benefits in the case of the urban project of Org. B. For the Scrum process used, working with local IT personnel helped the work of the PO and identified as one of the roles to support UCD practices in the situation. Customer representative in the software organizations may not articulate the necessary requirements for designing interactive systems as it might be difficult to visualize future system. It is important to have usability specialists in the software development team working in collaboration with developers to help for designing UIs in a participative way with users. However, in a situation where

there is lack of usability professionals like the case considered here, working with supplementary roles including local IT personnel and support departments doing acceptance testing for early feedback to developers will help as a solution in the existing situation.

Light and lean personas, prototypes and the roles local IT personnel are supporting the lightweight UCD practices. Support departments perform acceptance testing supporting developers getting early feedback. Lightweight usability activities such as culturally adapted user-pair testing and heuristic evaluation supported UCD activities to identify user challenges and early identification of design flaws for developers. A repository of lightweight methods developed for UCD in the lifecycle including UCD roles for the iterations and activities in the development process is supposed to be a sustainable approach to software development in developing economies. Local IT personnel are important in the context, however, with improvements in skills and advancements in technology it might not be a preferred solution.

As the implemented practices and results start spreading and bring improvements in the development process, the management needs to support in leading the adaptation process and integrating it to the project management activities for sustainability of the practices.

ACKNOWLEDGMENTS

We would like to thank our collaborating organizations for their support and cooperation. Thanks to the development teams and operational and support personnel for their openness and participation.

REFERENCES

- [1] Maunder, A. Marsden, G. Gruijters, D. and Blake, E. 2007. Designing interactive systems for the developing world-reflections on user-centred design, In *Information and Communication Technologies and Development, ICTD 2007*, IEEE.
- [2] Bridges.org. 2005. Real Access/Real Impact Framework for Improving the Way ICT Is Used in Development (Bridges.org Concept Note). https://pasdbp.files.wordpress.com/2008/04/bridgesorg_real_access_real_impact1.pdf
- [3] Teka, D. Dittrich, Y. and Kifle, M. 2016. Usability challenges in an Ethiopian software development organization, In *Proceedings of the 9th International Workshop on Cooperative and Human Aspects of Software Engineering*, pp: 114-120, DOI: <http://dx.doi.org/10.1145/2897586.2897604>, ACM.
- [4] Teka, D. Dittrich, Y. Kifle, M. Ardito, C. and Lanzilotti, R. 2017. User Involvement and Usability Evaluation in Ethiopian Software Organizations, *The Electronic Journal of Information Systems in Developing Countries, EJISDC (2017)* 83, 8, 1-19
- [5] Agile Manifesto, Manifesto for Agile Software Development, <http://agilemanifesto.org>, Accessed on Tuesday November, 2017
- [6] Blomkvist, S. 2005. Towards a Model for Bridging Agile Development and User-Centered Design, A. Seffah (eds.), *Human-Centered Software Engineering – Integrating Usability in the Development Process*, 219-244, Springer.
- [7] International Organization for Standardization. 2010. Ergonomics of human-system interaction- part 210: Human-centred design for interactive systems (ISO 9241-210:2010).
- [8] Ardito, C. Buono, P. Caivano, D. Costabile, M. F. and Lanzilotti, R. 2014. Investigating and Promoting UX practice in industry: An Experimental Study, *International Journal of Human-Computer Studies*, 72(6), pp. 542-551. <http://dx.doi.org/10.1016/j.ijhcs.2013.10.004>, Elsevier.
- [9] Brhel, M. Meth, H. Maedche, A. and Werder, K. 2015. Exploring principles of user-centered agile software development: A literature review, *Information and Software Technology*. 61 (2015) 163-181, Elsevier B.V.
- [10] Rannikko, P. April 2011. *User-Centered Design in Agile Software Development*, M.Sc. Thesis, University of Tampere, School of Information Sciences

- [11] Singh, M. 2008. U-SCRUM: An Agile Methodology for Promoting Usability, *Agile 2008 Conference*, IEEE
- [12] Nodder, C. and Nielsen, J. 2009. *Agile usability: Report on best practices for user experience on agile development projects*, 2nd edition, Fremont, CA: Nielsen Norman Group, 2009. Retrieved from <http://www.nngroup.com/reports/agile/>
- [13] Nielsen, J. 1995. Applying Discount Usability Engineering, *IEEE Software*, vol. 12, no. 1, pp. 98-100
- [14] Kane, D. 2003. Finding a Place for Discount Usability Engineering in Agile Development: Throwing Down the Gauntlet, in *Proceedings of the Conference on Agile Development*
- [15] Sohaib, O. and Khan, K. 2011. Incorporating Discount Usability in Extreme Programming, *International Journal of Software Engineering and Its Applications*, V5, No.1, pp. 51-62
- [16] Winschiers, H. 2006. The challenges of participatory design in an intercultural context: Designing for usability in Namibia. *Proceedings of the Ninth Participatory Design Conference*, Vol-II (pp. 73-76)
- [17] Biru, T. 2008. *Reflective Steps: A Collaborative Learning Oriented Approach to Software Development and Process Improvement*, PhD Dissertation, Hamburg University
- [18] Doerflinger, J. and Dearden, A. 2013. Evolving a software development methodology for commercial ICTD projects, *Information Technology and International Development*. 9 (3), 43-60
- [19] Pade-Khene, C. Mallinson, B. and Sewry, D. 2011. Sustainable rural ICT project management practice for developing countries: investigating the Dwesa and RUMEP projects, *Information Technology for Development*, Vol. 17, No. 3, 187-212
- [20] Aregawi, T. 2013. *Framework to Define Software Process Model in Ethiopian Context*, Master's Thesis, HiLCoE School of Computer Science and Technology
- [21] Dino, H. 2012. *A Framework for Integrating Software Usability into Software Development Process*, Masters Thesis, HiLCoE School of Computer Science and Technology
- [22] Dittrich, Y. Rönkkö, K. Eriksson, J. Hansson, C. and Lindeberg, O. 2008. Cooperative method development: Combining qualitative empirical research with method, technique and process improvement, *Empir Software Eng (2008)* 13:231-260, Springer
- [23] Robson, C. 2011. *Real world research: A Resource for Users of Social Research Methods in Applied Settings*, Wiley
- [24] Teka, D. Dittrich, Y. and Kifle, M. 2017. Contextualizing User Centered Design with Agile Methods in Ethiopia, *AFRICON, 2017 IEEE*, pp: 911-916, DOI: 10.1109/AFRICON.2017.8095603, IEEE
- [25] Pierotti, D. 2005. Heuristic evaluation—a system checklist, Tech. Rep., Xerox Corporation, Society for Technical Communication
- [26] Jia, Y. Larusdottir, M. K. and Cajander, A. 2012. The usage of usability techniques in scrum projects, In *International Conference on Human-Centred Software Engineering*, pp: 331-341, Springer Berlin Heidelberg
- [27] Hofstede, G. <https://geert-hofstede.com/ethiopia.html>, accessed December, 2017