Designing Interactive Systems for the Developing World – Reflections on User-Centred Design

Andrew Maunder, Gary Marsden, Dominic Gruijters and Edwin Blake

Abstract—**User-centred design (UCD) is a well-accepted and useful design methodology for designing interactive systems. In recent years, developing world researchers have attempted to utilise UCD but with mixed results. The results from two developing world, UCD projects, MuTI Mobile and CyberTracker, have shown that the analysis tools and techniques provided by UCD are useful but difficulties arise when interpreting the analysis findings to produce a requirements specification. In particular, traditional UCD methodologies fail to consider the broader and complex effects of the user's physical and social environments. The field studies also highlighted the limitations of existing early-stage prototyping techniques, such as paper-prototyping.**

The authors address these issues by presenting several tools and techniques that they feel are more suited to the developing world and essential components of a candidate 'UCD4Dev' methodology. These tools and techniques include the use of '4Dev' frameworks, such as the 'Real Access/Real Impact' criteria, to highlight pertinent developing world issues, the use of higher fidelity technology artefacts during early stage prototyping, the importance of developing a motivated user group and the need for a progressive participatory design approach.

Index Terms—User centred design, Human factors, Graphical user interfaces, Developing nations.

I. Introduction

User-Centred Design (UCD) has long been a useful design methodology for creating interactive systems. The process has largely been effective due to its relentless pursuit of putting the user first at all costs.

Whilst UCD has been successful in industry and research in the developed world, very few well-documented studies using UCD in the developing world exist. People/community centred approaches have been used within other frameworks for implementing ICT projects in the developing world but few detailed accounts exist where these frameworks have been specifically applied to a user-centred, interactive system design.

UCD as a design methodology does include a number of practices that are advantageous in a developing world context, but there are a number of major drawbacks, especially when attempting to represent the design solutions to the user or when describing and explaining the broader design processes. One possible reason is that technology is not ubiquitous within many developing world communities and organisations to the extent to which it pervades life in the developed world. Often work and social processes are not structured around technological solutions, nor are they easily adaptable to accommodate such technologies. Thus the users are not always able to envisage any technological based solutions; choose between various design options or abstractly place a technology into their daily lives or work activities – all key requirements for user-centred design processes.

Several UCD tools and techniques rely on assumptions about the users, for example assumptions relating to their existing attitude and familiarity towards technology. These assumptions are unfounded and misinformed, resulting in a largely haphazard, patchwork approach to UCD. It can be contended that the resulting interactive, technology systems are not optimally usable or even useful to the intended end-users.

This paper aims to present the shortcomings of UCD when applied in a developing world context. Insights extracted from two South African, UCD case studies will be used throughout the paper. In both studies, the analysis phases of a UCD process proved useful for understanding the users and their immediate context but failed to appreciate the broader environmental and socio-cultural factors. Several UCD tools and techniques proved difficult to apply in the developing world context and are thus critically reviewed. Finally, the authors present recommendations for a candidate UCD4Dev methodology.

II. Background

User or people-centered design approaches have been recommended by several ICT4Dev (ICT for Development) researchers and organisations [4, 12, 13] but there are few detailed accounts that document the application of a UCD methodology in its entirety. None of these accounts have evaluated the appropriateness of existing UCD methodologies in developing world design initiatives.

In the following section we discuss the applicability of existing UCD methodologies in a developing world design initiative, highlighting the strengths and shortcomings of these methodologies.

A. A User-centered approach

Traditionally, a '*user centred'* approach focuses on the target users from the beginning of the design process, continually checking the design with the users to ensure that

they are in fact comfortable with it [7]. Formally, the international standard ISO13407, *Human-Centred Design Processes for Interactive Systems*, defines a UCD approach in terms of the four core values of UCD [7], namely:

1. Understanding & specifying the context of use
2. Specifying the user & organisational requirements
3. Producing design solutions
4. Evaluating the design against requirements

The first core value, '*understanding & specifying the context of use'*, is encapsulated into what the authors term a user-centred, analysis phase. The findings/outcomes of this analysis phase are used as the input for the generation of a detailed requirements specification. Values three and four, '*producing design solutions'* and '*design evaluation against requirements'*, are central components of design and prototyping phases.

This paper will focus on and discuss the applicability of existing UCD tools and techniques in each of the first three phases listed above. The fourth phase, evaluation, we will leave as an open question – within our experience at least, discovering the requirements (see section 4) is such a complex task that it is hard to know if the project is finished, let alone be in a position to evaluate it.

III. Sample Projects

To aid in our analysis of UCD for the developing world, we present two case studies which were implemented in different parts of South Africa using different research teams. By using case studies, we aim to concretise the discussion within an understood framework.

A. The MuTI and MuTI Mobile Projects

The MuTI [4] and MuTI Mobile projects were initiatives that evaluated the impact of low cost, telecommunications technologies on a rural community. The field trials were conducted in the Eastern Cape of South Africa, a rural area notorious for a poor service infrastructure due to various historic, political, socio-economic and environmental factors, (see Figure 1). Both projects attempted to connect a remote rural clinic with the centralised rural hospital via long distance Wireless Fidelity (WiFi) networking. The clinic chosen was the most remote in the district due to it being the furthest away from the local hospital and only serviced by poorly maintained dirt roads. In addition, the clinic was void of any fixed-line telecommunication and electrical infrastructure. Worse still, the clinic was situated at the bottom of a valley where there was no cellular signal coverage thus leaving the clinic completely under serviced.

The original MuTI project utilised PC based hardware to serve as 'Voice over Internet Protocol' (VoIP) clients and aimed to provide the clinic nurses with a communication platform to connect their remote clinic with the local hospital. Despite an extensive training program, the overall system usage statistics were less than desirable leaving the technology under utilised. An initial expert review of the user interface and experience highlighted several serious usability issues that needed attention. The follow up, MuTI Mobile project, aimed to rectify these issues and simplify several complex interaction processes by using an MMS style metaphor on a Windows Mobile handset (see Figure 2).

Despite the MuTI [4] project aiming to be a participatory design initiative, the active user participation never progressed further than the work process and communication analysis phases. The designers experienced difficulties when attempting to engage users about technology related topics. The nurses were reluctant to provide any commentary on the PC based MuTI system until they had been adequately trained. It was clear that the nurses were completely inexperienced technology users and afraid of making mistakes. To avoid any embarrassment, they requested to be shown how to complete every task, step by step, thereby preventing true participatory design from being realised.

Figure 1: The rural clinic in the Eastern Cape of South Africa

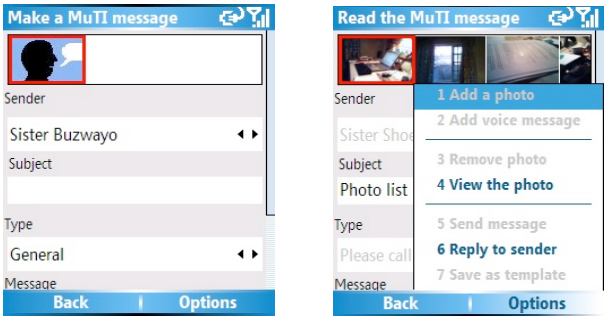

Figure 2: MuTI Mobile interface

B. The CyberTracker Project

The CyberTracker project [2] was initiated in 1996 with the purpose of creating a usable interface for collecting nature conservation data for use by illiterate animal trackers. The aim of this project was to preserve indigenous knowledge among the users and empower users through the use of a mobile device. The original study made use of a variety of user centred methods, such as ethnographies, user observations, paper-prototyping and critical action research.

Recently a follow-up study has been carried out to ascertain the penetration and adoption of this system in South African National Parks. The initial UCD phase of the project consisted of an analysis of the user's work processes and use of the current software interfaces. Interviews and artefact walkthroughs were conducted with a variety of system users. Following the lengthy analysis phase, UCD methods such as participatory design and paper-prototyping were utilised in an attempt to encourage a variety of users to participate in the design of a more useful and usable system, with the ultimate

goal being to improve the penetration and adoption of the CyberTracker software throughout South African National Parks. So far, several difficulties stemming from existing UCD tools and techniques have been experienced, resulting in a largely haphazard, patchwork UCD approach.

IV. User-Centred Analysis

A. Results from MuTI and CyberTracker

1) Contextual Design

The MuTI Mobile designers utilised Contextual Design [1] tools and techniques successfully during the analysis phases of the design process. The generation of various work and social models are examples of existing UCD tools that were found to be particularly useful in a developing world context. Model generation and re-design are key aspects of this popular user-centred methodology.

Contextual Design [1] is essentially a fusion of user-centred and ethnographical principles, where the designer/researcher is an observer trying to gain an understanding of the user and their immediate context (environment). Typically a contextual design process consists of seven phases, namely a contextual inquiry, work modelling, consolidation, work redesign, user environment design, prototype and testing [1]. Thus it emphasises the development of an understanding of the users' daily goals, tasks and how the users respond to various prototyping techniques and artefacts. Parikh, Ghosh and Chavan [13] demonstrated the need for contextual, ethnographic analysis techniques to be used. His results showed that such techniques are useful and thus a key component of any developing world design approach.

2) Workflow and communication models

A contextual inquiry provided the researchers with valuable information concerning the nurses, their daily tasks and their working environment. The results of the contextual inquiry were factored into daily workflow and communication models.

The communication model attempted to capture the daily, weekly and monthly communications shared by the nurses, thereby gaining an understanding of how the nurses related to the community, managers and hospital staff from a professional standpoint.

The workflow models were instrumental in the identification and understanding of the nurse's daily work processes, in particular, existing paper-based processes associated with the dissemination of laboratory results from the hospital laboratory back to the rural clinic. Parikh and Lazowska [12] comment on the value of identifying such paper-based processes when designing technology based systems in the developing world.

The same holds true for the CyberTracker [2] project where a careful analysis of the trackers' work practices revealed how mobile technology could be used to augment this process. The analysis tools and techniques (phase 1) provided by existing UCD methods have thus proven to be useful and effective in a developing world context. Techniques that observe and monitor users in their natural work environments, in particular, have assisted UCD practitioners in gaining an understanding of a users work processes and work environments. Observational ethnographies and contextual inquiry [1] are examples of such techniques.

The success of these methods, within a developing world context, can be attributed to the fact that there is no dependence on the users understanding or knowledge of a technology, choosing between technological alternatives or a technology assessment of any sorts. The whole of the analysis phase is independent of any technological understanding outside of the participants' usual frame of reference. The users and participants are observed with little pre-conceived ideas or assumptions of how they live and work. The UCD practitioner aims to embark on this process with a clear understanding that they do not 'know better' than users and are willing to accept the learning role in the process.

V. Requirements Specification

Whilst existing user-centred analysis tools and techniques have proven to be useful/applicable in developing world design initiatives, no guidelines or frameworks exist, within UCD, that address developing world specific issues nor does UCD provide any guidelines regarding the application of a UCD methodology within a developing world design context.

This problem manifests in phase 2, where observations are translated into requirements. In the developed world, a new ICT system is often added to an already existing ICT ecology. In the developing world it is more likely that any ICT system developed will be completely novel within the existing context. This means that it is hard to generate requirements upfront as the introduction of the system will massively impact the context in which it is being deployed. To help focus the requirements specification on pertinent developing world issues, we turned to the Real Access, Real Impact criteria.

A. Bridges.org

'Digital Divide' commentators, Bridges.org [3], provide valuable insights on how technology policies and initiatives, in particular Information Communication Technologies (ICTs), should be planned, researched, monitored and evaluated within a developing world context. They believe that the target people need real access to ICTs, '*access that goes beyond computers and connections so that the technology use makes a real impact on their socio-economic development [3]*'. To aid in this goal, they created twelve criteria focused on creating ICT projects for the developing world. We shall examine just those criteria which we believe have bearing on the UCD process.

B. The Real Access, Real Impact Criteria

The Bridges.org, '*Real Access/Real Impact*' (RA/RI) criteria [3] represents an abstract, analysis framework that addresses the real issues surrounding technology access and use in developing world communities.

Despite the criteria not originating from interaction design

research (HCI, UCD etc.) specifically, the abstract nature of the criteria, and the fact that it exhibits a *'people-centred'* focus, allows it to be applied to the several UCD processes. The criteria are particularly useful when they are used as a lens during analysis or evaluation phases of a design process [9]. It is important to note is that RA/RI criteria is not a generative framework, it should rather be considered a set of guidelines that aim to encourage designers, researchers and practitioners to ask pertinent and probing questions relating to the technology and its use within a developing world context.

As existing UCD analysis tools and techniques fail to address issues stemming from the broader physical and social usage context, a candidate developing world, user-centred design methodology would need provide the analysis tools and techniques that are able to address user specific issues as well as issues relating to the broader environment of use.

Thus an RA/RI approach helps augment UCD and focus the designer's attention on pertinent developing world issues, aids critical reflection and emphasises the need for assessing micro (user-centred) and macro (community or environmental-centred) factors.

C. RA/RI Environmental Analysis

An analysis of the physical environment in which the technology will be used, is a key component of the RA/RI criteria. The criteria, *"physical access to technology"* and *"appropriateness of the technology",* both focus on developing world issues relating to the micro and macro physical environment.

The rural Eastern Cape is void of any fixed line infrastructure due to the notorious, undulating landscape. The MuTI Mobile network designers opted for a multiple node, point-to-point WiFi network that stretched between the local hospital and the remote clinic. The RA/RI criteria reminds designers that developing world regions are typically void of basic service infrastructure and that they must be aware of these constraints in order to design an appropriate technology solution. An example developing world requirement may be that the technology's battery life should be long and as a consequence the design would need to energy efficient. This requirement stems from the fact that rural areas exhibit low power line penetration or that the existing power lines are unreliable at best and thus may prevent a user from recharging the technology on a daily basis.

Thus, by using these RA/RI criteria in the requirements specification phase, the system creators can start to assess the impact that the ICT intervention will have in the widest possible context.

These ideas are echoed by Parikh, Ghosh and Chavan [13] who points out that these physical factors also influence the design of software applications. His CAM framework compensated for environmental constraints by utilising a robust *'store and forward'* model to compensate for sporadic network and power outages. The MuTI Mobile system design, followed a similar approach by utilising a cellular phone platform with store-and-forward messaging model for these exact reasons.

D. RA/RI User Analysis

Designing a useful and usable system requires a detailed understanding of the target users skills, needs, goals, desires and motivations and is an essential step in the UCD analysis process. The RA/RI criteria aim to focus the requirements phase of a UCD process on important, developing world, user specific issues.

As a first example, the RA/RI criteria consider the affordability of a technology based on the financial situation of the target users. Parikh, Ghosh and Chavan [13] feel that personal computers are not affordable for rural users and that poor communities are unable to foster the economic environment necessary to support such technologies. Developing world communities and users often are some of the poorest and their economic situation may not allow them to invest in costly technologies or technology use, even if it is hugely beneficial. The RA/RI criteria led the MuTI Mobile designers to consider minimising the financial burden on the users as well as the broader community, thus the design utilised WiFi technology when transferring communication data.

Another essential user-centred, RA/RI guideline is to ensure that the *"accessibility of the technology"* is considered and ensured at all times. Again the value of the RA/RI criteria is shown in that it addresses micro (user-centred) and macro (community-centred) issues relating to technology accessibility. These micro accessibility issues may relate to how the users interpret imagery displayed on the user interface or the literacy levels of the users. Whereas at a macro level the broader physical accessibility issues may be highlighted, such as the policies concerning how the technology is stored or who controls access to the door keys. Both micro and macro accessibility must be ensured if the solution is to have any chance of success and impact on the users daily lives.

Broadly speaking the RA/RI criteria aim to highlight sensitive and critical issues that, if ignored, may become a potential point of failure. What makes these issues unique is that they would probably never be considered in a requirements specification from the developed world. Consider the *"accessibility of the technology"* criterion and the issue of storing the technology safely. Imagine a laptop computer in a rural clinic that needs to be locked away at 5pm every day for security reasons. What if the caretaker, who holds the keys, falls ill and is not contactable? The caretaker is a vital component in the technology accessibility chain but due to the sensitivity of this chain, a single stress can cause it to break thereby limiting any access to the technology.

VI. Socio-cultural Complexities

RA/RI criteria 7 is optimistically titled *"Socio-cultural factors."* This highlights the fact that existing UCD analysis approaches often fail to identify deeply rooted and intricate socio-cultural issues, thus leaving these issues unexposed. In the developing world, it is often the case that the technology designer and technology user come from completely different cultures, language groups and countries. A UCD practitioner

may therefore believe that progress is being made towards a seemingly appropriate, useful and usable design, only to be derailed by complex socio-cultural issues later in the design process.

Below is an extract from an interview with user from the CyberTracker study, wherein a socio-cultural issue is highlighted:

"My rangers don't use the CyberTracker units because they know I can see where they've been during the day and they think I'm using this system to police their movements so I can reprimand them when they're not doing their jobs"

It is obvious that the users view of the technology is warped and that the true benefits of the technology are overshadowed by the rangers notion of the technology being a policing device rather than a animal tracking tool. Socio-cultural issues, such as those presented in the CyberTracker example, must be addressed early, if not these issues may prevent a seeming appropriate, useful and usable technology system from being utilised effectively in the future.

The situation is aggravated by the fact that organisational managers and community leaders, despite being highly influential in the user's daily tasks and activities, are unable or unwilling to support new technology based tasks and activities. It is often the user alone who decides whether or not to use the technology, thus these socio-cultural issues directly effect the users motivation and drive to use the technology.

Existing UCD analysis tools and techniques fail to address these deep-rooted and intricate socio-cultural and organisational issues, leaving seemingly appropriate, useful and usable technology designs at risk of being 'unsuccessful'. RA/RI makes little recommendation other than to note that these factors (whatever they might be) should be taken into account.

UCD practitioners need new tools and techniques that are able to expose these issues early, thus avoiding costly changes late in the design process. Currently, no suitable UCD tools or techniques exist that offer practitioners any assistance in exposing complex socio-cultural issues other than time-honoured ethnography.

VII. Producing Design Solutions

Designers often use prototypes as a means of capturing design solutions and presenting it to the users as part of a user-centred evaluation. Feedback from the user is then used to inform the design process and motivate design changes. During a typical user-centred design process, designers may present prototypes of varying fidelity to the user. The goal being to ensure that the users real needs and desires are captured within the design, thereby cultivating an environment conducive to open communication, participation and a shared understanding between the design team and the users.

Embarking on the prototyping phase of the UCD process traditionally makes a few assumptions:

1. Users understand their own work and processes enough to inform the design process
2. Users understand how technology might aid their work or daily lives
3. Users are able to grasp abstract design concepts
4. Users are sufficiently knowledgeable about technology to make informed decisions when given a choice

These assumptions are typically made when attempting to establish the design space in which the designer and the user will participate. The authors have found that developing world users typically have difficulties seeing the benefit of the design process it self. This is a direct result of being a technologically inexperienced user and a user who does not fully understand the design process or how these unfamiliar technologies may fit into it. Such technological disparities are a reality when attempting to design systems that aim to bridge the so-called *'Digital Divide'*.

UCD practitioners would typically begin a prototyping phase with low-fidelity prototypes in order to present the current design concepts to the user in an abstract manner, early in design process. The goal is to quickly generate design representations, at low cost, that are able to elicit valuable feedback from the user. Low-fidelity prototypes also aim to give a sense of 'changeability', encouraging the user to suggest changes.

Paper prototyping [15] is a common technique used for low-fidelity prototyping purposes and has already been used in developing world design initiatives [10, 12] with mixed results. Thus far, no formal evaluation of how effective low fidelity prototyping techniques are when applied within a developing world design context has been conducted and documented. Parikh, Ghosh and Chavan [13] and Medhi, Sagar and Toyama [10] briefly mention some of the difficulties experienced during paper prototyping sessions but do not provide any significant data on this topic.

A. Shortcomings of Existing User-centred Prototyping Tools and Techniques

Two issues arise. Firstly, Medhi, Sagar and Toyama [10] states that the users were intimidated due to a lack of technology experience or basic understanding of how to use technology. This is understandable when considering that technology systems, of any sort, are often not ubiquitous within developing world communities. Thus, the design team had to actively find ways of making the user group feel comfortable during the early stages of the design process. Both Medhi, Sagar and Toyama [10] and Parikh, Ghosh and Chavan[13] attempted to limit user intimidation through the use of low-fidelity prototyping techniques.

Whilst cultivating a less intimidating design atmosphere, low fidelity prototypes rely on the use of design abstractions when representing design concepts. Thus, the second issue of concern relates to the use of design abstractions and how developing world users often misunderstand and misinterpret low-fidelity prototypes, as noted by both Medhi, Sagar and Toyama [10] and Parikh, Ghosh and Chavan [13].

Similarly, the MuTI Mobile designers attempted to introduce the local ICT trainer to the MuTI Mobile concept through the use of low fidelity paper drawings. Throughout the duration of the session there were significant

misunderstandings on behalf of the trainer. The confusion stemmed from the abstract nature of the entire activity, thus making the goal of the design session unclear.

Lim et. al. [8] reports on this issue, highlighting the difficulties associated with the validation of low fidelity mobile prototypes, specifically when attempting to determine if the results are an effect of the prototype it self or the design concept that the prototype wishes to capture. In the developing world, the results are often an effect of the prototype alone, leaving the design process, design concepts and design goals completely misunderstood.

Parikh, Ghosh and Chavan [13] opted to migrate to a higher-fidelity prototype after only a few low-fidelity prototyping sessions. The use of a high-fidelity prototype did expose the design concepts more clearly to the users but once again the results and findings of the prototyping session were effects of the prototype itself and not purely an effect of the design concept. In addition, opting for a high-fidelity prototyping approach during the early phases of a UCD process is costly due to the time and effort needed to produce the prototype.

Thus, both high and low fidelity, early stage, prototyping approaches exhibit shortcomings and at present UCD offers no clear solution to this problem. Practitioners, such a Medhi, Sagar and Toyama [10], have managed to overcome these difficulties in a largely haphazard manner. They persevered with low-fidelity prototypes over an extended period time, clearly highlighting a flaw in the method. Such an approach is most certainly not suitable for industrial design initiatives that are known for their tight schedules. In addition, persisting with low-fidelity prototypes may lead to a situation where intricate socio-cultural issues, relating to the technology it self, are only discovered late in the design process. Often these discoveries are only made once a higher-fidelity prototype is deployed and as such making changes difficult and costly.

VIII. User-centered Design for Development – A New Approach

Firstly, the authors suggest that a candidate UCD4Dev methodology not only attempt to analyse the user and his/her environment but also prioritises the development of the user and the supportive structures within the user's living and working environments. Existing UCD tools and techniques often rely on assumptions about the target users in order for their proper operation. UCD currently lacks strategies that aim to prepare the users for participation within the design processes, thus allowing existing approaches to function.

A user-development focus would ensure the progression and development of user's knowledge base and skill set, thereby enabling the user to better understand the technology, the benefits it offers and how to utilise it effectively. An essential step if active user participation is desired. The development of supportive organisational and social environments for technology use is another important area of concern when designing technology systems for the developing world. Developing world, managerial and community leadership processes are often found to be unsupportive of technology solutions.

In keeping with the true spirit of UCD, UCD4Dev practitioners must evaluate the design against the user and broader, environmental requirements. Thus, before any design prototype is presented to the user for evaluation, the user and the supportive environment must be developed to a level suitable for the current design phase.

The result is the co-evolution of the user and their environment as the technology design progresses, thereby providing the best possible opportunity for active user participation.

A. A Progressive Participatory Approach

The concept of progressive design, proposed by Chin and Rosson [5], provides an initial view on such a design approach. Progressive design includes the incremental development/evolution of the user's activities and processes alongside the design of the technology that is to support it. Thus, a progressive design could also provide a platform for the development of the users knowledge and skills that are necessary to support the newly evolved activities and processes.

If we consider Clement and Van den Besselaar's retrospective look at Participatory Design [6] they remind us of Kensing's basic requirements for participation, he states that employees (users) must –

1. Have access to relevant information.
2. Have the possibility for taking an independent position on the problems.
3. In some way participate in the process of decision-making.

To satisfy these requirements the users would require a certain level of insight and experience into ICTs, but these are often lacking within a developing world user group. A progressive approach focuses specifically on the development of skills and experience, thus attempting to satisfy Kensing's requirements and striving for a design environment that fosters participation.

B. Simple, Technology Artefacts with Instant Utility

A candidate UCD4Dev methodology would need to address the shortcomings of existing prototyping tools and techniques, in particular issues relating to the fidelity of the prototypes used.

Ramachandran et. al. [14] provides some early insights regarding this issue. Their approach to early-stage prototyping utilised simple, technology artefacts that were introduced into the target user group or community. The artefacts acted as *'technology baselines'* for stimulating dialog between the designers and the users. This approach exhibits several advantages over low-fidelity prototyping, firstly technology artefacts are fully functional technologies that are simple and display instant utility. When an artefact operation is demonstrated to the user, it is immediately obvious what the technology is capable of; its capabilities are explicitly exposed. The user is then able to reflect on the action performed by the technology and how it may correlate with

his daily tasks and activities without needing to grapple with complex abstract design concepts and ambiguities that may exist as an effect of the prototype itself.

The use of technology artefacts also exhibits an advantage over high-fidelity prototyping techniques. The artefact is usually an off-the-shelf product and therefore requires little modification or adaptation before it can be used. The result is that technology artefacts can be deployed during the early phases of the design process; at low cost.

When mobile, technology artefacts were presented to the MuTI Mobile trainers and nurses, the response was very positive. The nurses were able to quickly grasp the core concepts behind the technology and the application, as they were able to draw parallels between the MuTI Mobile application and the MMS feature of their own phone. The artefact achieved the desired goal of establishing a technology baseline and opened technology related dialog between the nurses and the designers. The same held true for the CyberTracker project as programmers were dispatched to the field to rapidly create interactive software prototypes. Working with the programmers gave the users a greater insight into not just the application, but also the mutable nature of software.

C. *Finding and Developing a Motivated User Group*

Ramachandran et. al. [14] also points out how public, technology-artefact demonstrations can be used as a means of attracting interested or curious participants. These participants would, in some way, understand the benefits of the technology and possibly form the basis of a motivated user group; possibly acting as key local informants [11] when addressing any design or socio-cultural issues that may arise.

The MuTI Mobile and CyberTracker projects identified user motivation as a key area of concern and recognised that the identification and development of a motivated user group is a vital component of the design process. The MuTI Mobile designers noticed that the motivation levels of the nurses often fluctuated. Initially the nurses were highly motivated to attend the PC training workshops but their motivation waned when left to utilise the technology on their own. Over time it became clear that the nurses understood that it was beneficial to become PC literate especially when applying for a position at a city hospital and this motivated them to attend all the training sessions and become active participants. These results lead the design team to believe that the nurses were at least able to use the basic communication features technology but once the training sessions had concluded, the communication features of the technology were not utilised.

On later reflection the designers recalled that one nurse made a comment in her home language. It was later interpreted as describing the system as a *'spy tool'* for the doctors. An obvious concern for the nurses when considering that they're work schedules was often in violation of their employment contracts.

Initially the nurses represented a user group that was motivated but this motivation was short lived and could not be sustained or developed due to an underlying socio-cultural issue and thereby adversely affected the entire design process.

D. *Exposure of Complex Socio-cultural Issues*

Complex socio-cultural issues, such as the one presented above, can adversely affect the success of a technology design initiative and thus designers must strive to uncover and understand these issues early in the design process.

Ramachandran et. al. [14] also highlights the importance of understanding the social structures that exist within the community and the influence these structures have on technology acceptance and use. A potential technology-based system design must appreciate the effects of these social structures and in fact must attempt to leverage them to ensure that the technology stands the best chance of acceptance within the community.

Intricate socio-cultural issues must be identified as early in the design process as possible, to avoid the potential benefits offered by the technology being obscured by these issues. Thus far, only Ramachandran et. al.'s [14] description of technology artefact usage in the early stages of the design process, offers a viable technique that is applicable and suitable for use in a UCD4Dev methodology. In our work, we have tried to find members of the target community who have lived or worked in a technology rich culture. Often these people can help identify the socio-cultural issues and act as a bridge between the cultures – in effect, we exploit their latent anthropological knowledge.

IX. Conclusions

Designing interactive system for the developing world poses some unique challenges. UCD practitioners have begun to comment on the sheer amount of field time that is necessary to ensure technology designs that are truly understood by the target users and truly meet their needs. The authors believe that this is partly due to practitioners utilising UCD tools and techniques that are not suitable for developing world design initiatives. At present UCD analysis, tools and techniques, only provide designers with insight relating to the user and their immediate context.

'*Digital Divide'* commentators and researchers, such as Bridges.org [3] believe that an appreciation for design factors originating from the broader social, organisational and physical environment is also required, even if the relation between the two is not immediately obvious. Their RA/RI criteria go further and encourage designers to ask pertinent and probing questions relating to ICT access and usage within a developing world context. The goal being to reveal critical and sensitive design issues that, once addressed, will ensure a design space that is more resilient to external stresses.

A candidate UCD4Dev methodology should also support two other important processes (both of which are absent from existing UCD methodologies), firstly, it should provide tools and techniques that probe socio-cultural attitudes towards technology, exposing socio-cultural intricacies that may affect the overall design and secondly, the development of the user and his/her supportive environment, thus ensuring a user that is able to actively participate in the design process and an

environment that is able to support the associated tasks and activities.

Ramachandran et. al. [14] presents a novel approach for early stage technology prototyping that attempts to probe social attitudes towards technology by establishing a *'technology baseline'* that is used to foster technology related dialog between the user and the practitioner.

The authors also believe that a progressive design approach is required to support the necessary user and environment developmental processes. Essentially, design iterations should provide an accompanying set of development processes that aim to prepare the user and his/her supportive environment for the upcoming design phase. Thus, the result is an empowered, confident, motivated user that is able to actively participate in every step of the design process.

Currently, no proven and replicable UCD4Dev methodologies exist and significant work is still required to define a candidate UCD4Dev methodology suitable for industrial and academic design purposes.

ACKNOWLEDGMENT

This work has been supported by IDRC, Telkom/Siemens Center of Excellence, bridges.org and HP.

REFERENCES

[1] Beyer, H. and Holtzblatt, K. Contextual Design. *ACM Interactions*, 6, 1 (Jan./Feb. 1999), 32-42.

[2] Blake, E., Steventon, L., Edge, J. & Foster, A. 2001. A field computer for animal trackers. In *Proceedings 2nd South African Conference on Human-Computer Interaction (CHI-SA 2001)*. ACM SIGCHI, 2001.

[3] Bridges.org. Real Access/Real Impact framework for improving the way ICT is used in development – Concept Note. December 2005.

[4] Chetty, M., Tucker, W. and Blake, E. Developing locally relevant applications for rural areas. *In Proceedings South African Institute of Computer Scientists and Information Technologists (SAICSIT 2004),* Cape Town, South Africa, 239-243.

[5] Chin, G., Jr. and Rosson, M., B. Progressive Design: Staged evolution of scenarios in the design of a collaborative science learning environment. *In Proceedings of CHI conference on Computer Human Interaction (CHI '98)*. (Los Angeles, CA, USA, April 18 - 23). ACM Press, New York, NY, 1998, 611-618.

[6] Clement, A. and Van den Besselaar, P., A retrospective look at PD projects. *Communications of the ACM 36*, 6, 2003, 29-37.

[7] Jokela, T., Iivari, N., Matero, J. and Karukka, M. The standard of user-centred design and the standard definition of usability: analyzing ISO 13407 against ISO 9241-11. *In Proceedings of the Latin American conference on Human-computer interaction (CLIHC '03)*. (Rio de Janeiro, Brazil, 2003). ACM Press, New York, NY, 2003, 53-60.

[8] Lim, Y., Pangam, A., Periyasami, S. and Aneja, S. Comparative analysis of high and low-fidelity prototypes for more valid usability evaluations on mobile devices. *In Proceedings of NordiCHI conference on Computer Human Interaction (NordiCHI'06)*. (Oslo, Norway, October 14-18). ACM Press, New York, NY, 2006, 291-300.

[9] Maunder, A.J., Tucker, W.D., and Marsden, G. Evaluating the relevance of the 'Real Access' criteria as a framework for rural HCI research. *In Proceedings of the 5th Conference on Human Computer Interaction in Southern Africa (CHI-SA '06)*, (Cape Town, South Africa, 2006). ACM Press, New York, NY, 2006, 75-79.

[10] Medhi, I., Sagar, A. and Toyama, K. Text-Free user interfaces for illiterate and semi-literate users. *International Conference on Information and Communication Technologies and Development*. (Berkley, USA, May 2006), 72-82.

[11] Millen. D., R. Rapid Ethnography: Time deepening strategies for HCI field research. *In Proceedings of the DIS Conference on Designing Interactive Systems (DIS '00). (Brooklyn, New York)*. ACM Press, New York, NY, 2000, 280-286.

[12] Parikh, T. and Lazowska, E.,D. Designing an architecture for delivering mobile information services to the rural developing world. *International World Wide Web Conference (WWW2006)*. (Edinburgh, Scotland, May 23-26). ACM Press, New York, NY, 2006, 791-800.

[13] Parikh, T. Ghosh, K. and Chavan, A. Design studies for a financial management system for micro-credit groups in rural India. *Conference on Universal Usability (CUU '03)*. (Vancouver, British Columbia, Canada, November 10-11). ACM Press, New York, NY, 2003, 15-22.

[14] Ramachandran, D., Kam, M., Chiu, J., Canny, J. and Frankel. J.L. Social Dynamics of Early Stage Co-Design in Developing Regions. *In Proceedings of CHI conference on Computer Human Interaction (CHI '07)*. (San Jose, CA, USA, April 28 - May 3). ACM Press, New York, NY, 2007, 1087-1096.

[15] Snyder, C., Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces. Morgan Kaufmann, 2003.