Research and Practice in Technology Enhanced Learning
Vol. 1, No. 3 (2006) 239–267

ETHNOCOMPUTING IN TANZANIA: DESIGN AND ANALYSIS OF A CONTEXTUALIZED ICT COURSE

ERKKI SUTINEN

Department of Information Systems, Massey University
Palmerston North, New Zealand
sutinen@cs.joensuu.fi

MIKKO VESISENAHO

Department of Computer Science, University of Joensuu
Joensuu, Finland
mvaho@cs.joensuu.fi

A crucial challenge in the teaching and learning of information and communication technologies (ICTs) in developing non-Western countries is the irrelevance of much Western learning content, materials, tools, and methods from the perspectives of local communities. Our approach for designing an ICT course is based on the concept of ethnocomputing — finding culturally suitable entry points for understanding, utilizing, and producing ICT in a relevant way. In order to study the feasibility of our approach, we designed a contextualized introductory programming course with Tumaini University, Tanzania, and offered it to second-year undergraduate students in teacher education in the academic year 2004–2005. The approach was to encourage students to create programs that would meet the needs of their own local communities. The course made extensive use of learning technologies, such as robotics, to concretize the first steps in understanding ICT. We applied an ethnocomputing-based analysis to three aspects: (1) the development process of the course; (2) the role of learning technologies in the course; and (3) students' contextual learning outcomes. First, representation, utilization, and appropriation appeared at various stages in the development of the course. These included unexpected spin-offs for local communities, such as the introduction of educational robotics to assist young patients in a rural hospital. Second, contextual relevance of learning technologies — digital learning materials, the robotic I-BLOCKs, and the Jeliot program visualization tool — was analyzed by the CATI model, and the results indicate their relevance at all the four levels of the model: Contextualization, Application, Transfer, and Import. Third, student learning outcomes were observed by using content analysis of questionnaires and interviews. There is evidence that students developed a contextual understanding of ICT, especially at the application level of the CATI model. Overall, the study indicates that an ethnocomputing-based analysis can be used for designing needs-based ICT education by analyzing its feasibility with various operationalizations of the ethnocomputing concept. Moreover, research results envisage new components and methods for an entirely contextualized undergraduate curriculum of ICT.

Keywords: Ethnocomputing; ICT education; Tanzania; development.

1. Introduction

One of the critical challenges facing those who attempt to introduce information and communication technology (ICT) in non-Western countries is its irrelevance from the perspectives of local communities. Conventional learning content, materials, tools, and methods are mostly of Western origin and do not necessarily address the needs of potential non-Western users of ICTs. Moreover, for most non-Western learners, ICT is a completely new subject field, and hence its contents cannot easily be built on the previously known. Teaching of ICT also mainly aims at certain skills, such as programming, and most teachers prefer to follow a path which exposes students to the area step by step. However, even if ICT education emphasizes skills, the learners find it very hard to apply them to real world tasks, especially at the beginning of their studies.

The term "Computing" refers to the family of closely related disciplines of Computer Science, Computer Engineering, Software Engineering, and sometimes also Information Systems Science. ICT can be understood to belong to this family as well, as a practically oriented subject, intended mainly for minor studies. Nearly everyone who works in Computing Education Research (CER) uses the universal ACM/IEEE Computing Curricula (ACM & IEEE, 2001) to anchor her/his understanding of what students should learn, and on which basis learning outcomes should be measured. Although such an approach might be justified on purely *conceptual* grounds, it might attract a researcher to ignore the *realities* of the learners' background in those cases where the cultural assumptions of learners are radically different from those of learners who have grown up in so-called Western cultures where ICTs are more commonly an integral and accepted part of everyday life.

One of the achievements of traditional universal computing education is that the teaching of programming has been designed and analyzed mainly from a point of view determined by the *syntax* and *semantics* of various programming languages and paradigms. An alternative *pragmatic* approach would be to begin the design process by making a careful study of what ICTs in general (and specifically programming) would look like if they were designed to accommodate learners whose social and cultural realities are very different from those of learners from highly developed countries, such as the US, Japan, Australia, and those in the European Union. We call such an approach *ethnocomputing* (Tedre, Sutinen, Kähkönen, & Kommers, 2006) because it strives to create a discourse of teaching and learning that frames computing as theory and of the practical use of ICTs in culturally relevant terms that are immediately familiar to local learners.

Because Computing Education is traditionally standardized, it provides us with an interesting perspective for studying the role of various cultural factors in different aspects of a course. This applies particularly to learning technologies as they represent both the content and learning tools of an introductory ICT course. This explains the choice of the three aspects of this study: (1) the development process of the course, (2) the role of learning technologies in the course, and (3) students' learning outcomes.

Most learning technologies that have to date been applied in, say, programming courses have been *theory-driven* (or course-driven or content-driven). For instance, students use a compiler to apply their knowledge as they build reliable and predictable programs, or they visualize them so that they can see how various syntactic structures function. The theory-driven approach builds upon an emphasis on the syntax and semantics of programming.

A *community-driven, needs-based* approach is an alternative to the theory-driven approach of learning ICT. It is consistent with the innovation systems approach for economic improvement that is emphasized in the recent report on achieving the Millennium Development Goals (Task Force on Science, Technology, and Innovation, 2005). Comprehensiveness, common to community-driven and innovation systems approaches, requires that multiple viewpoints can be taken into account simultaneously. The community-driven approach is based on the pragmatic focus of ICT education.

The key question in this study is "What is the design process of a community-driven ICT course?" A complementary question is "How can such an approach be feasible; that is, how can a community-driven ICT course make sense in its context?" This requires a feasibility analysis of the approach. These two aspects of design and analysis are closely related.

We will answer the first (design-related) question by describing the design and implementation of an introductory ICT course that was devised for second-year undergraduate students in the Department of Education at Iringa University College (IUCO) of Tumaini University in southern Tanzania. The course material and program were constructed to reflect the everyday life of local learners. Extensive use was made of learning technologies such as robotics to concretize the first steps in understanding the opportunities of ICT. The students were encouraged to create programs that took cognizance of the expectations, needs, and traditional concerns of their communities. This kind of approach to studying ICT reflects the essence of ethnocomputing which requires one to identify culturally meaningful entry points for understanding, utilizing, and producing ICT in a relevant way.

We will answer the second (analysis-related) question by presenting a feasibility analysis of our approach that consists of the three aspects of this study mentioned above. We based the analysis of outcomes on questionnaires and interviews. We classified the primarily qualitative data by using Atlas.ti software and interpreted the way in which students learned ICT by means of the CATI model's four levels of **C**ontextualization, **A**pplication, **T**ransfer, and **I**mport (Vesisenaho, Kemppainen, Islas Sedano, Tedre, & Sutinen, 2006).

In a study like this, it is important to recognize the limitations and reservations that the researchers' background might have on the actions and observations throughout the research process, as well as the comprehension and interpretation of its results. The researchers came from a Computer Science department of a Finnish university, but they have been in a regular working relationship with the Tanzanian university from the year 2000. This collaboration became even more intense

Table 1. Ethnocomputing approach to the design and analysis of an introductory ICT course in Tanzania (The numbered items refer to sections in this paper.).

	Design	Analysis
Development process of the course	2.2	4.1
Role of learning technologies	2.3	4.2
Students' learning outcomes		4.3

throughout this study, and the role of local teachers was crucial for improving and analyzing the course.

The overall research scenario is presented in Table 1. It is a conceptual test of the validity of ethnocomputing in developing community-driven studies in ICT. The structure of the paper follows the research scenario. Section 2 describes the context, including the course itself and the use of learning technologies therein. Section 3 presents the research method, starting from the three operationalizations of the ethnocomputing concept. Section 4 reports the findings of the study, with our interpretations. The results are discussed in Section 5, and Section 6 concludes the paper.

2. The Context

2.1. *Iringa University College, Tumaini University*

Tumaini University is a private university run by the Evangelical Lutheran Church of Tanzania. It consists of four campuses in different parts of the country. Its biggest campus, Iringa University College (IUCO), is located in the southern highlands of the country. Iringa itself, the capital of the region, has a population of 350,000 (The United Republic of Tanzania, 2002). IUCO, in which this study took place, consists of four faculties: Arts and Social Sciences (which includes Departments of Journalism, Cultural Anthropology and Tourism, and Education), Business Administration, Law, and Theology (Tumaini University, 2005).

The student profile of IUCO clearly reflects the demographics of a sub-Saharan African university in which students come from many different ethnic backgrounds in terms of both nationality and tribe. Because the University is a private one and charges relatively high tuition fees, a number of students are reliant on scholarships. Most students come from rural areas and have no tradition of academic study in their families. What this means in practice is that their prior experience of ICT is either nonexistent or very limited indeed.

Out of all the IUCO graduates in 2004, a total of 41% were female (Bahendwa, 2004) compared with a Tanzanian average of 27% (The United Republic of Tanzania, 2005). Although these figures are not directly comparable, they indicate that the gender division at IUCO is somewhat more even than it is in other Tanzanian universities. The student body at IUCO has been growing. It was 200 students in 1999, 670 in 2004, and more than 1,000 in 2005 (Tumaini University, 2005).

The 27 students who took part in the current study are representative of the student demographics referred to above. Their ages ranged from 24 to 44 years, with the average age being 35 years. There were 16 men and 11 women in the group. All except two students had preliminary teacher education. The group came from 16 different regions of Tanzania and from 17 different tribes. Most of the students' parents were farmers and housewives, showing that the University serves the whole community and not just the rich and privileged. About eighty percent of students had used computers for only one year.

The mission statement of IUCO (Tumaini University, 2006) includes a commitment to provide staff, facilities, and support appropriate to a university of the highest order, to promote research, its organization and application to the learning environment, and to support service to the community, region, and nation through the church, the government, industry, public, and private organizations. The mission statement reflects a contemporary understanding of the three main tasks of a university: teaching, research, and commitment to the community. The successful implementation of a mission statement such as this in the modern world requires a certain level of ICT efficacy. This also explains why IUCO places a strong emphasis on, and makes a clear commitment to, expanding the use of ICT in all its activities.

IUCO obtained its first second-hand computers in the mid 1990s. Even in those first years, the focus was already on including ICT applications such as word processing in the curriculum. In 1999, IUCO launched a strategic plan for ICT development (Ashford, 1999) that emphasized human resource development, institutional capacity building, and the development of infrastructure. This plan has since served as a solid basis for the use and implementation of ICT in education.

From the point of view of this study, it is important to pay attention to how ICT can help a teaching and research institution, such as IUCO, to fulfill its societal task. The teaching and learning of ICT need to be contextualized in terms of the local culture precisely because IUCO is committed by its ethos and mission to community service. Any kind of teaching at IUCO must therefore be seen to have such a focus. The expectation implicit in this kind of teaching is that students who have been taught in this way will take the lead in shaping the future of their country in a way that emphasizes service to the community. This perspective indicates also the relevance of this type of research beyond its original scope.

2.2. *Course description*

2.2.1. *Aims of the course*

The introductory course on ICT (known as the Contextualized Introduction to Programming) is a compulsory course for second-year undergraduate students in the Department of Education. The aim of the course is to help students to understand and appreciate the extent to which ICT can enrich and enhance human activities in authentic environments. Students who are trained in this way should be far more likely to use ICT as effective and efficient application tools wherever they can in their future jobs compare to those who are not trained in this way. Although most of the students from a group such as this will end up as subject teachers of science or mathematics in secondary schools, the purpose of the course is to give such teachers an in-depth appreciation of just how effectively ICT can be used in any typical Tanzanian community setting to serve the community itself. In addition, such teachers will be in a position to inspire their future students with a vision of just how constructively ICT can be used, so that the students themselves ultimately become intelligent consumers of ICT rather than hapless users.

The aims of the course determine its curriculum in terms of content, methods, materials, and activities. In this case, the course objectives emphasized *programming competence* at the cost of practical experience in learning how to use various types of office software. The idea was to equip these future teachers with a certain level of skill in programming so that they could resolve technical difficulties and use ICT for their own purposes, either by requiring more from ICT services or, in some cases, programming by themselves. We also felt that a technically orientated course would give teachers the practical wherewithal to inspire their future students with the excitement and potential inherent in ICT.

However, before the course evolved to this stage, it underwent an incubation period of almost five years. This period of incubation is described in the following section.

2.2.2. *A brief description of the history of the course*

Table 2 gives an outline of the various phases through which the course design process evolved. We began course development in the *Preparations phase* with two masters-level students from Tumaini University studying at the University of Joensuu in Finland. These students helped us to tailor the new ethnocomputing approach to ICT education that we proposed for their own local Tanzanian context. As a starting point, we invited them to engage in our Finnish ViSCoS (Virtual Studies of Computer Science) non-degree program which allows high school students to take first year, university level Computer Science courses over the web (Haataja, Suhonen, Sutinen, & Torvinen, 2001).

One of the two master's students went back to Tumaini after he had graduated. With him and other ICT experts from IUCO, we organized *Workshops* in 2003 and 2004 to evaluate the usefulness of robotics in ICT education (Vesisenaho & Lund, 2004). Our hope in doing this was that we might obtain new and useful ideas for developing contextualized ICT education at IUCO.

On the basis of the encouraging opportunities that we identified in the Preparations phase, as well as the feedback from the Workshops, we launched a more serious *Consultations* phase in which we planned to launch a completely revised and redesigned Introduction to Programming course. When we analyzed the implementations of the programming courses of 2002 to 2004 that had been held at IUCO, we noticed that this course, targeted at novices, included, in its earlier stages, an introduction to the BASIC programming language and, later on, the implementation of advanced data structures and concepts such as stacks, queues, and trees

Table 2. The five phases of contextualizing the Introduction to Programming course.

Phase	Time	Objective	Partners	Contents and/or Activities	Learning Technologies
Preparations	2000–02	To get ideas on how to contextualize basic Computing concepts in the Tanzanian setting	UJ	Two IUCO BBA graduates at UJ studying for MSc in CS	The Finnish version of the first year virtual CS studies, ViSCoS
Workshops	2003–04	To identify the role of robotics in an introductory ICT course	SDU, UJ	One-day workshop with students who were building and programming robots	I-BLOCKs
Consultations	2003–04	To create a syllabus for a new, contextualized ICT course	IUCO, SDU, UJ	Analysis of IUCO's existing programming course	
Course I	2004–05	To design and analyze the first version of the course	IUCO, SDU, UJ	Workshop, lectures, assignments, project, examination	I-BLOCKs, Jeliot, ViSCoS.tz
Course II	2005–06	To implement a new, improved course	IUCO, UJ	Workshop, lectures, assignments, project, examination	I-BLOCKs, Jeliot, ViSCoS.tz

IUCO = Iringa University College of Tumaini University; SDU = University of Southern Denmark; UJ = University of Joensuu; I-BLOCKs, Jeliot and ViSCoS.tz are explained in Section 2.3.

in the C language. Students who did this course did not have any programming experience before the course began.

We continued to hold consultations between Tumaini University and its partner universities (the University of Joensuu and the University of Southern Denmark) in the first half of 2004 on the basis of what we, by then, knew to be the requirements of the local programming course. At the same time, we began to develop the course materials and structure. During this design process, we occasionally returned to the course structure of previous years. Part of the problem of negotiating a mutual understanding between ourselves and those at IUCO arose out of the necessity to define the general requirements of programming courses in terms that would be universally accepted and understood in the global Computing community. In the end, we reached amicable agreement about what the course content should be.

We launched the first newly designed course, referred to here as *Course I*, in October 2004 with an activating workshop based on robotics. A comparison with traditional programming courses will show that the syllabus of this course paid less attention to covering all the topics of a standard programming course and, instead, encouraged students to build their own programs and projects — even with incomplete knowledge. This course, as well as its successor, *Course II* (2005–2006), which was an improved version of Course I based on student feedback, ended with a concluding workshop in which the students presented the projects which they had partly carried out in local schools or hospitals.

2.2.3. *Content of the course*

The course that we agreed upon in this way included an introduction to the basics of computer programming. This introduction placed a special emphasis on Java which, because it is the lingua franca of the web, bridges the way to more technical tasks. Our first expectation was related to learning technical skills: the students should have a good general knowledge of the methodology of creating computer programs after they had successfully completed this course. They should also acquire a general understanding of robotics and modern technologies. The second expectation was more contextual: the students should appreciate the usefulness of programming and its application in their own society. Our long-term hope was that students could map their technical skills onto the needs of their surroundings.

From the point of view of technical content, the course was an unconventional hybrid of two topics seemingly far removed from each other at first sight: programming by building and the basics of Java. The first topic covered an introduction to robotics and mainly event-driven programming, and the second one covered procedural programming with control structures and applets. The course carried a weight of three local credits, which is equivalent to about four ECTS (European Credit Transfer System) credit points.

2.2.4. *Course I arrangements*

Course I was presented by two teachers, one from the University of Joensuu and the other from Tumaini University. Because the Tanzanian teacher had other responsibilities apart from teaching, he was only occasionally present. The actual teaching thus fell to the teacher from the University of Joensuu. The course consisted of the following activities:

- a workshop;
- lectures that were supported by weekly exercises (including a learning diary) and an optional tutoring session;
- a project assignment;
- a project presentation session; and
- a final examination.

The first part of the course consisted of a one-and-a-half day workshop in which students used I-BLOCKs (intelligent building blocks) for programming (see Section

2.3.2). After this, they were taught the Java language in a computer laboratory during six hours of weekly contact teaching over a six-week period. Each week's teaching program also included web-based exercises, a learning diary, and three hours of optional tutoring (Lund, Nielsen, Sutinen, & Vesisenaho, 2005).

During the course, each student received six sets of weekly exercises that were based on the six units of the course material. While such exercises gave students opportunities to practice their skills, they gave their teachers information about levels of understanding among the students. The weekly assignment also included a learning diary in which students were required to reflect on their learning and develop applications relevant to the local context. This exercise was also intended to make both students and teachers more aware of, and sensitive to, their local context.

For the purpose of projects, students were divided into groups that contained three to five members per group. Half of the groups undertook a project on teaching basic ICT skills to secondary school students by using I-BLOCKs. The other groups engaged in a Java programming project in which they had been asked to develop a program that would be beneficial to secondary school teachers or to students. The topic choices included the programming of a calculator, making a program that could create graphic representations of polynomial functions, making a program for a simplified Bao game, and the making of three different mathematical programs of their choice. All of these topics made sense for the future subject teachers of mathematics.

2.2.5. *Changes in course arrangements for Course II*

The objectives of Course II were the same as the objectives of Course I in the previous year, except that there was less focus on robotics and I-BLOCKs because the I-BLOCKs had been temporarily misplaced after the first course. We also set aside less time for workshops than we had done in the previous year and extended the completion time for the whole course. The students in this year had better access to printed reference material because the books they needed had finally reached the library. In all other respects the course was the same.

For programming projects, two groups of students taught secondary school students about ICTs by using I-BLOCKs. In another group, the students introduced orphan children to the potential of ICT by making use of I-BLOCKs and A-BLOCKs (African I-BLOCKs), a newly created set inspired by the feedback from a special project related to Course I with hospitalized children. In the Java projects, students were given the freedom to develop any program that might be beneficial to teachers or students in secondary schools in Tanzania. The topics chosen for those projects were: a $2 * 2$ matrix program, solving integration of exponential functions, and a statistics program for analyzing student scores.

The teacher from the University of Joensuu was effectively the only active teacher of the course. The reason for this was that the new local ICT teacher had to be engaged elsewhere in teaching other courses. He also had an onerous workload as a technician and simply did not have time to update his own programming skills as well as plan and attend the lessons. He was nevertheless kept informed about what was happening in the program as a whole. The four best students from Course I were selected as assistant teachers so that they could supervise projects during the course. The idea was to train them, so that at least one of them would be able to present the course in future with only online support from the University of Joensuu.

2.2.6. *Spin-offs*

The contextualized ICT courses we organized benefited various Tumaini University students and communities in the vicinity of Iringa in some unexpected ways. First, three students from Course I were selected on merit to undertake special projects in ICT as part of their studies. The titles of the projects they were given were:

- the influence of teaching strategies on students' achievement in learning computer programming in Tanzanian secondary schools;
- an investigation of the role of computer programs in the teaching and learning of mathematics: the case of Java and Maple 9; and
- an investigation into the application of I-BLOCKs in a Tanzanian context.

In addition to this, one Course I student was awarded a five-month grant to continue her studies in Finland. Second, the University of Tumaini has begun to build the Tumaini University Science Park, the first of its kind in East Africa. Successful prototypes of A-BLOCKs have been developed and are due to be produced in the Science Park. Plans for an ICT bachelor degree with a strong local-content curriculum are also getting underway. The Technology for Education in Developing Countries Conference (TEDC 2006) was held at Tumaini University in July of 2006.

Third, student projects became seriously engaged with local concerns in the Iringa region and extended their activities in various fruitful directions. An interesting example of such a project in which course content and methods are used to further learning activities (in this case in local hospitals) may be seen in how students have been using ICT to produce therapeutic aids for hospitalized children (called the Ilembula Project).

2.3. *Role and use of learning technologies*

The learning technologies used in the Contextualized Introduction to Programming course can be divided into digital learning materials and activating tools. Since the design of the course was guided by the principles of ethnocomputing, we stipulated that learning technologies should comply with the following requirements:

- Digital learning materials should be presented in such a way that students should feel comfortable, rather than culturally alienated, while studying any new topic in this field. The presentation should be culturally adapted that students will find it easier to come to grips with the material.

- Activating tools should equip students to express themselves in terms of the new concepts of programming that they are learning. These tools should also stimulate students to create new and exciting ways of applying ICT in their own communities. In order to learn skills and get technical competence, one has to practice; that is why it was crucial to have activating tools in addition to more or less static learning materials.

2.3.1. *Digital learning materials*

The main course material, called VisCoS.tz, formatted as light web-based material in HTML, was developed by researchers at the University of Joensuu with assistance from a lecturer from IUCO. We used Flash for navigation animations. We had Java applets in the materials and then made the most of programming code locally available to the students because the limited Internet capacity made downloading, or even browsing, slow. We also used other web-based materials because books were only available on a limited basis and web-based material is easy to deliver and update. We also circulated most of the web-material in the form of printed handouts to students because they had only limited access to computers that were connected to the University's LAN. We also ordered a few textbooks on Java, entitled *Java for students* and *Introduction to Java,* for our students. The simultaneous use of web material and printed course materials was a novel experience for the students at that time.

We published the course material for each week (comprising six units) in the local server. The metaphor of the instructional interface was a village (Figure 1).

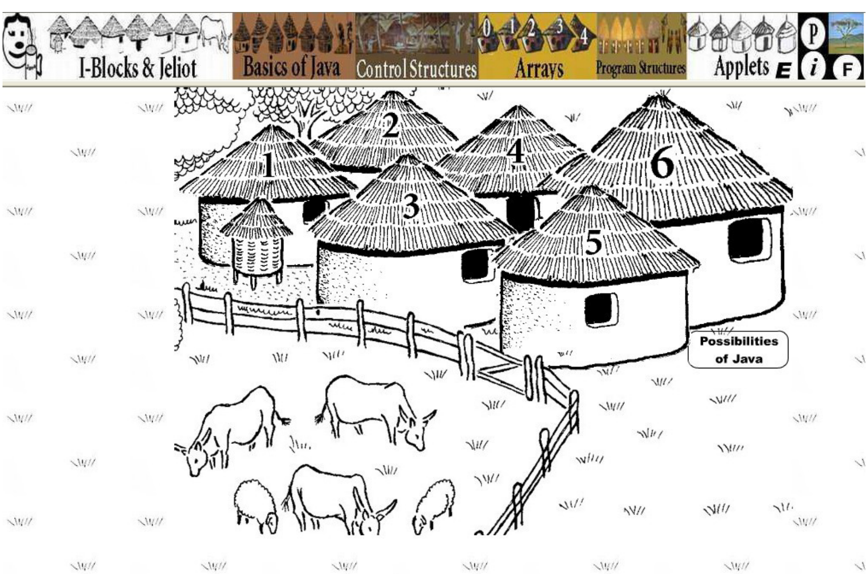

Fig. 1. The learning village as a graphic metaphor of the course material.

Students using the site could obtain the information they needed by clicking on the huts and the people in the village. We used this symbolism deliberately in order to demonstrate that computer programming does not have to be presented in obscure "high tech" Western-specific formats, but it can be made easily grasped if it is taught by means of imagery and concepts that are entirely derived from the local social and cultural context of learners. It was, of course, interesting to see how students received our graphical representations of their own context.

We included programming examples that students could download and modify. Even though they were given weekly problems to solve, they were also encouraged to come up with their own ideas and plans for making programs. The course also supported four PHP-based forums: question–answer, project work, general information, and free discussion.

2.3.2. *Activating tools*

Since the course was a hybrid of two complementary approaches to learning programming, we felt that it was necessary to support each of these approaches with an appropriate activating tool. We were also keen to find out how these tools could be integrated into the overall design of the course and how they might further be developed by means of feedback from our learner group. We hoped that because our learners had not had much exposure to ICT, they might be able to offer some original, ingenious, and stimulating suggestions for modifying our existing tools.

I-BLOCKs (Figure 2) allow users to build a program by connecting together blocks of three different kinds (Lund, 2003):

- *input blocks* with a sensor or mechanism to set a value;
- *output blocks* producing a tone, displaying a value, etc.; and
- *operator blocks* for arithmetic or logical operations.

Their modular design makes it easy for learners with no prior programming skills to use I-BLOCKs to assemble, say, a program that produces an alarm (by an output block with a sound synthesizer) when it recognizes a touch (by an input block with a touch sensor). We used I-BLOCKs to introduce programming concepts because these I-BLOCKs teach students the fundamentals of modern technologies while they

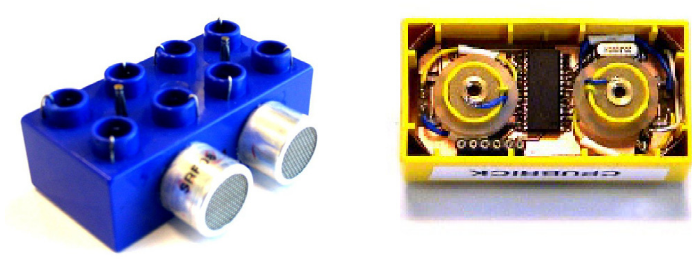

Fig. 2. I-BLOCKs with microprocessor and communication channels (Lund, 2003).

use them in various arrangements to construct an assortment of devices. The University of Southern Denmark also developed a simple programming interface called I-BLOCK Application for the course. This interface allowed students to program I-BLOCKs for different functions in an easy manner. The QEL Micropro program was used for downloading programs to the Microchip inside the I-BLOCKs.

We were particularly intrigued by two aspects of the use of I-BLOCKs by our students. The first was how I-BLOCKs inspired students to come up with new ICT-based innovations that were relevant to their own communities. The second was how familiarity with I-BLOCKs enabled students to make valuable suggestions about ways in which they might be further developed and refined.

Jeliot program visualization software allows students to follow each step of what happens in their Java programs (Figure 3). It has been shown to be particularly helpful to students who are new to programming (Levy, Ben-Ari, & Uronen, 2003). Jeliot is also useful for illustrating the behavior of the basic control structures and arrays. In some ways, it functions much like I-BLOCKs. It is a hands-on tool that allows students to experiment with code and see the otherwise inaccessible configurations of a running program.

One of the main challenges when using any visualization tool is how to make it an integral component in the course design. Visualizations that may look quite appealing to teachers might make little sense to learners. The ethnocomputing approach requires us to ask how exactly any visualization tool has been culturally adapted to facilitate the learning process. *Adaptability* in this context means the extent to which learners can adjust the visualization to suit their cultural preferences. An adaptive system adapts automatically to the user's profile. How cultural adaptation may be distinguished from cognitive adaptation is addressed in various places in the existing literature about program visualization.

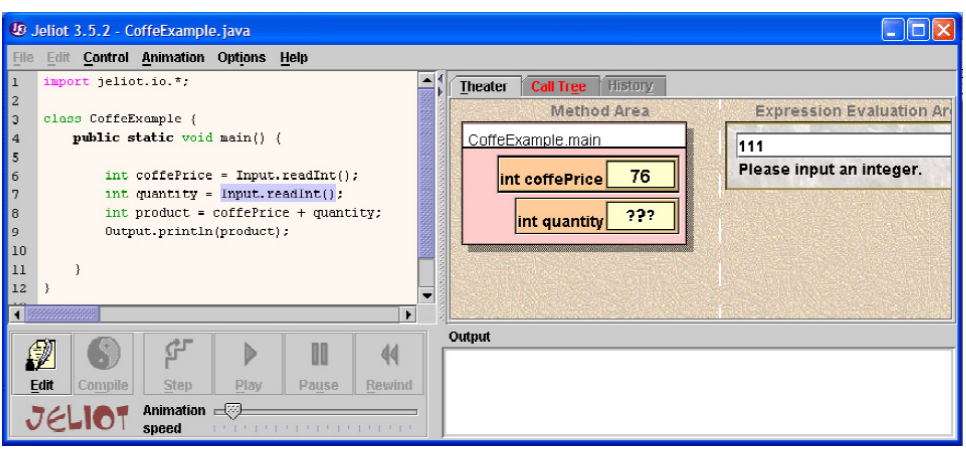

Fig. 3. Screenshot of Jeliot visualizing a program. The code is in the left window, its dynamic visualization in the right one.

3. Methods

The first question in this study focuses on what happens when a learning technology takes the cultural background of learners into account in order to make learning possible. We will answer this question by describing the design process of the course.

The second question about the feasibility of the design can be answered by presenting an analysis of the three aspects of the learning outcomes as they became evident in the students' application of ideas for ICT. Successful ideas for the use of ICT that were culturally relevant to local communities were considered by us from the very beginning to be indicators of successful course design.

In this section, we will describe the methods that we used in the study. The general framework of our analysis is rooted in the concept of ethnocomputing which is described in Section 3.1. At this juncture, our analysis will explain how we deliberately incorporated certain cultural adaptations into the design of the Contextualized Introduction to Programming course. We used the CATI model to operationalize the concept of ethnocomputing (Section 3.2). This model ensures that an ICT is culturally *sustainable* or *rooted* in a community because it tests whether or not the ICT concerned complies with the four stages designated by the CATI model (Contextualize to Apply to Transfer to Import). The CATI model can be used to analyze both the design of the course and its learning processes or outcomes. The analysis of the learning outcomes requires a content analysis, and we will return to this in Section 3.3. The research scenario is represented in Figure 4. The arrows show which operationalizations of the ethnocomputing concept are used to analyze each of the three aspects. Learning technologies, for example, need to be analyzed both in terms of ethnocomputing and the CATI

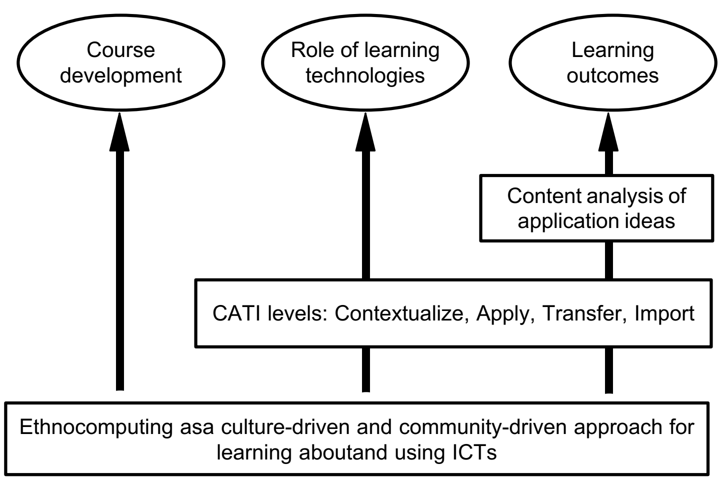

Fig. 4. The overall research scenario.

model. We will analyze the course development process in Section 4.1, the role of learning technologies in Section 4.2, and the learning outcomes of the students in Section 4.3.

3.1. *Ethnocomputing as an explanatory framework*

Ethnocomputing (Tedre *et al.*, 2006) refers to an approach that makes it easier to understand, use, and apply ICTs and the theory of computing. This can be done by interpreting such ICTs in terms of the *particularities* and idiosyncrasies of an ICT user's own culture rather than in the terms that were developed and that are usually regarded *universal* in the developed world. While the universal language of computing discourse cannot be set aside, the difficulties of translating computing theory and methods for using ICT into terms that ordinary people can understand, constitutes a clear failure of the computing community to transfer the extraordinary benefits of its expertise to those who desperately need it.

We have begun to compensate for this serious omission by teaching community-driven or culture-driven computing in the spirit of the UN Millennium Goals (United Nations General Assembly, 2000) that call for no person to be excluded from the benefits of this science. In our practice of ethnocomputing, we teach the subjects that comprise the Computing family to student populations from previously disadvantaged communities that have had little or no exposure to modern ICTs. In ICT education, *ethnocomputing* refers to any approach to the learning and teaching of computing theory and of ICTs that is mediated by a discourse that has been specially adapted to make it familiar and user-friendly to those whose culture is different from the industrialized culture of developed countries. Ethnocomputing emphasizes the need to create culturally meaningful entry points for learners to understand, utilize, and produce relevant ICT.

Ethnocomputing in any given sociocultural setting can be analyzed from the point of view of representation, utilization, and appropriation (cf. Eglash, 2004). *Representation* refers to conceptual models, mental models, and methods of teaching. *Utilization* deals with the various uses of technologies, diffusion patterns, and social attitudes towards technology. *Appropriation* refers to creative initiatives that challenge our mainstream ideas about what should be happening. These include the use of technology for nonstandard purposes, job creation that is stimulated by innovative business ideas, and the creation of effective ad hoc solutions to technological problems.

In this case, as we are using ethnocomputing to analyze the design process of a particular computing course, it can be applied at the different stages of the course's lifespan. An analysis of this kind would include:

- analyzing the needs that gave rise to the course in the first place and identifying the reasons why the course was created;
- establishing the objectives, goals, and aims of the course;

- organizing the course (this includes teacher allocation, methods, scheduling, etc.);
- developing the learning materials;
- designing and testing the learning technologies;
- evaluating the learning process and outcomes; and
- taking note of whatever unexpected spin-offs the course may give rise to.

We will analyze the Contextualized Introduction to Programming course from the point of view of representation, utilization, and appropriation (Section 4.1). This will give us a broad understanding of the design process that the course underwent. For a more specific analysis, such as an analysis of the role of learning technologies or that of learning outcomes, we will need a more accurate model. The following section explains how we used the CATI model (Vesisenaho *et al.*, 2006) to apply the principles of ethnocomputing.

3.2. *The CATI model as the analysis method*

The CATI model allows us to determine how sustainable and viable a particular ICT solution or course is by applying the four levels of the model — Contextualization, Application, Transfer, and Import. The CATI model (Figure 5) can be used for developing sustainable ICT in developing countries, for analyzing the planning and implementation processes of ICT, for education, and for evaluating a particular person's ability to apply previously acquired knowledge.

Let us now define each of the levels in the four-level model. We are particularly interested here in how the model can illuminate and serve the *local needs* of the population for which it is intended.

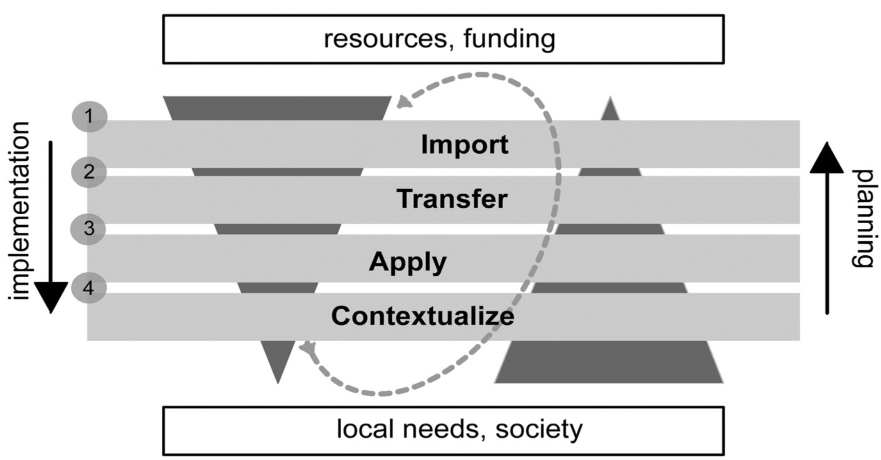

Fig. 5. The four-level CATI model for representing and analyzing the planning and implementation processes of ICT (Vesisenaho *et al.*, 2006).

- *Import*: Importation refers to any situation in which an innovative technology is imported from abroad before any local needs analysis has been undertaken. Importation may thus by implication also indicate a situation in which non-sustainable development is offered to local people.
- *Transfer*: Transferred innovations are those that are accessible to their users and potentially applicable in a local context. The analysis of local needs in such cases may have been weak, incomplete, or even nonexistent.
- *Apply*: Application means that the inherent potential of a transferred innovation has being realized. An application of this kind will not be merely mechanical in its implications. It will contain elements that show that users have exercised their own initiative to apply the ICT.
- *Contextualize*: Contextualizing means that users have thoroughly integrated ICT with the needs and concerns of their community. For a developing country, the development and application of contextualized innovations, with its origin in the country, is essential so that the innovations could benefit the country by meeting its particular development needs. If ICT is to be sustainable, it needs to be contextualized to serve the local community.

Although these four levels of the CATI model sum up the three constituent parts of ethnocomputing (representation, utilization, and appropriation), it can also be fruitful to use these two dimensions independently of one another for purposes of analysis. The mere importation of ICT might lead to the creation of new jobs — and that would be an indication of the appropriation of an ICT. Representation is also present in all ICT teaching, independently of its CATI level.

The levels of the CATI model allow us to determine the *sustainability* of a learning technology at the level of design in any given ICT course (Section 4.2).

When we analyze the learning process or outcomes of a particular ICT course, the focus of our CATI analysis is on the extent to which students have elaborated the ideas and skills they obtained in a particular course. The content analysis required for this is offered in the following section.

3.3. *Analysis of the learning outcomes*

In this article our operational ethnocomputing approach in ICT focuses on education that would be relevant to students' background knowledge, to the state of ICT in the local country, and to the local development in globalizing world. Relevant use of ICT also requires awareness of application opportunities.

Our approach to learning outcome analysis is two-fold. We gathered data by questionnaires and interviews (Figure 6). In the first phase, the learning outcomes were measured in terms of how students' ideas for applying ICT changed during the course. Therefore, questionnaires were designed to indicate how aware students were of ICT and which opportunities they saw in it. The questionnaires were administered in the beginning and end of Course I, in early October 2004 and January 2005.

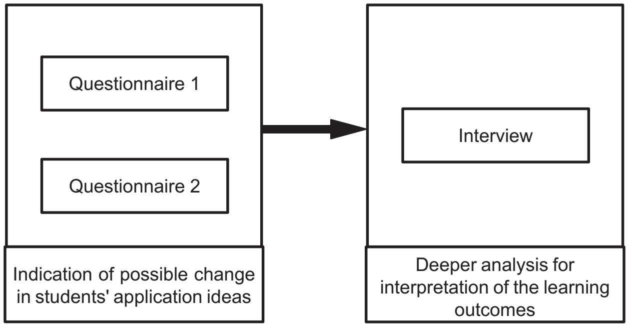

Fig. 6. Research approach to the analysis of students' learning outcomes: data collection and analysis.

The thematic interviews were conducted at the end of the course in January 2005, to give a deeper picture on the students' acquired understanding of ICT in terms of the CATI levels. We analyzed the learning outcomes of the students of Course I by using a content analysis approach with prefixed categories (Neuendorf, 2002). We used Atlas.ti software to classify the qualitative data that we collected from these interviews, and we then interpreted the data we thus collected in terms of the CATI model. The CATI model thus functions as an explanatory framework that guides our content analysis, seeking evidence of learning and application in students' profession and in society.

In order to increase the inter-rater reliability and to examine the CATI model in this purpose, we had two independent coders carry out the coding of each interview. The coding was done twice and independently. One of the coders was a researcher who had worked in the Tanzanian ICT project since 2000, and the other was a MSc in Computer Science interested in cultural dimensions of ICT. The focus was on students' attained CATI level of each coding unit and the general CATI level across all coding units in an interview based on students' learning and application ideas.

Free-marginal Kappa, a chance-adjusted measure of agreement, was used to evaluate inter-rater reliability (Brennan & Prediger, 1981). There were no other courses which could have affected the outcomes, because there were no other theme-related courses held at the same time. As the students were, on average, 35 years old and had 9 years of teaching experience, we did not expect age or teaching experience to affect the results. We will present the analysis of students' learning outcomes in Section 4.3.

4. Analysis

The analysis that we present in this section is based on data from Course I (2004–2005) unless otherwise indicated. The first step is the overall analysis of the course development process, then the learning technologies, and finally the deepest

Table 3. Phases of the development process of the Contextualized Introduction to Programming course that are relevant for the ethnocomputing aspects of representation, utilization and appropriation. For phases, refer to Table 2.

	Representation	Utilization	Appropriation
Analyzing the reasons for the course	Preparations	Workshops	Consultations
Setting the objectives, goals and aims of the course	Course I	Preparation, Workshops	Consultations
Organizing the course (teacher allocation, methods, scheduling, etc.)	Consultations	Workshops	Consultations
Developing the learning materials	Preparations, Course I	Preparations	Course I
Designing the learning technologies (support tools)	Preparations, Workshops, Course I	Workshops, Course I, Course II	Workshops, Course I, Course II
Evaluating the learning process and outcomes	Course I, Course II	Course I, Course II	Workshops, Course I, Course II
Identifying the unexpected spin-offs of the course	Course II	Course II	Course I, Course II

insight is on students' learning outcomes, focusing on their application ideas of the course content and ICT (Figure 4).

4.1. *Course development process*

We will analyze the development process of the Contextualized Introduction to Programming course (Course I) in terms of the ethnocomputing aspects of representation, utilization, and appropriation. Table 3 shows how each category was taken into account at a different phase of the development process and at a particular stage of the course. In the Consultations phase (see Table 2), for example, the teaching methods of the course were at least partly agreed upon. This is indicative of *representation*. The analysis clearly shows how the course development process strongly reflected the ethnocomputing design approach.

The course design and implementation process was also analyzed using the CATI model (Vesisenaho, Duveskog, Laisser, & Sutinen, in press). The application level was most often reached in practical operations. Objectives were evaluated with respect to the contextual level. Joint activities were based on the transfer level.

4.2. *The role of learning technologies*

If we use the description in Section 2.3 of how learning technologies should be used, we can illustrate the extent to which these technologies complied with the ethnocomputing approach for the design of Course I by using the levels of the CATI model. The analysis is summarized in Table 4. It shows that the role of learning technologies indicates cultural relevance. That is to say, all of the three applied and

Table 4. The CATI levels of the learning technologies used in the Contextualized Introduction to Programming course.

	Contextualize	Apply	Transfer	Import
Digital learning materials	The visual design of the materials indicates that students can use ICT in their communities.	The material is rich in exercises. It encourages users to create their own programs.	The material is available to users at IUCO even though they may not have access to the web.	The core contents of the course comply with the globally standardized CS curriculum.
I-BLOCKs	Students can continue to refine the design of I-BLOCKs so that they become even more useful and valuable in their contexts.	I-BLOCKs inspire students to invent and create prototypes of ICTs that can be used in their communities.	I-BLOCKs are easy to use and make. They make programming accessible even to those who have no prior expertise in the field.	The I-BLOCKs were designed abroad (in Denmark), and were imported into the situation.
Jeliot	Using Jeliot challenges students as well as teachers to come up with ideas for cultural adaptations of program visualization.	Jeliot opens the black box of computer operations and inspires learners to think about program applications.	Jeliot permits users to create visualizations with their own programs.	Jeliot was imported into the situation.

Table 5. The frequency of references to learning technologies in the interviews of students.

Learning Technology	Contextualize	Apply	Transfer	Import	Total
Digital learning materials	0	3	0	0	3
I-BLOCKs	2	19	15	4	40
Jeliot	1	7	3	0	11
Total	3	29	18	4	54

Note: For each student, multiple references to a learning technology in a certain CATI level were counted as one reference only.

analyzed technologies also represent the Contextualization and Application levels of the CATI model.

While Table 4 represents the *potential* of each learning technology in terms of different CATI levels and thus reflects the Western interpretation of the researchers, we studied how students regarded the relevance of these technologies with respect to the CATI model. Table 5 shows the number of references to each technology in terms of a certain level in interviews of students.

It is understandable that most coding units were from the application level. The large number of the I-BLOCK related coding units (40) and the significant number of these at the application level (19) indicate the major role of application in the learning process, as well as in understanding the concept of programming and ICT. Some of those answers also refer to the contextual problems of I-BLOCKs; they are not so far available in Tanzania and are not especially Tanzanian-made.

Yeah, I think the importance of I-BLOCK ... is that ... this when you learn something by manipulating or by using hands you learn better than just thinking abstract ... yeah, so that was, I find that ... to learn something that you touch and try to make experiments what is going to happen, I find that it is good ... So also the important, other important that ... I-BLOCKs can be used to teach students of low level, for better understanding.

Yeah, I realize that there's a need of I-BLOCKs and Java because in I-BLOCK is where you are starting to know how the program works and how the structure of the program is and the parts of how the each other, when we are talking about output and input in processing, through I-BLOCKs can understand the whole structure and thereafter you go to the Java but my (...) *is, the course is, there are a lot of things to do in just course but the time is not so* (...) *the whole coverage because it make you to be to rush quickly, not to have enough time to think about not* (...) *how to practice, how to do more things because the time is not enough.*

Jeliot was mentioned only 11 times, probably partly because Jeliot and Java programming were very much related in the course. An interesting point is that J-Creator was mentioned much more often, although it is only an editor, but is was of course used very often.

Yeah, in Java ... it has two categories. One is founding, using Jeliot, so in Jeliot you find that what is going on (...) *going on in the process are seen in the, are seen in the, so you say in easy for visualization, so you see the display of the what you have been doing ... and another in J-creator you find that ... that one is the most ... is above this Jeliot because you find that it performs complex programs ... and these are put in the* (...) *or in the applet.*

The digital learning materials of the course were not referred to very often (3), and so it is quite difficult to evaluate the contextual usefulness of them. When compared to the printed part of the material, the digital material included more examples, exercises, and animations. The village structure of the digital learning environment was to link it to students' everyday life, and a coding unit seems to support our idea of it. It also tells how novel the computers are for these students.

... let me tell you something, where I come from ... it is the first time to touch computer here, no computer because my schools is in the village ... I started touching the computer first time here ... but now each of the villages, which come in front of me is a new and I am happy to know it, for example, this Java it is the, the (...) *I had there when I am making a work, a workshop of* (...) *two days, now since* (...) *new course, actually I am happy very much, not only*

to me either my other friend in the class, I hope if you ask them such a question they are going to tell you such thing, we are very much with this course.

4.3. *Students' learning outcomes*

We started tracking the changes in the ICT application ideas of the students right from the beginning of Course I. Section 4.3.1 presents awareness of ICT and ideas of its contextual opportunities that students had at the beginning and end of the course. The learning outcomes of the students are analyzed by qualitatively based content analysis of their interviews in Section 4.3.2.

4.3.1. *Application ideas as awareness of ICT and its opportunities*

We devised questions (Table 6) focusing on the kind of ICTs students identify in everyday life (1), where they could apply robotics and programming (2), and how they could identify the opportunities of ICT personally (3), and in the society (4). We divided these into two dimensions: changes in awareness $(1, 2)$ and identifying opportunities $(3, 4)$. These results show a definite progression in the application ideas during the course. Especially, the combined variable (paired samples t-test) shows that the students were more aware of ICT and more able to identify its opportunities in the end questionnaire than in the beginning questionnaire $(p = 0.002)$.

In the awareness question (1), the biggest increase was on the realization of ICTs in institutions such as schools, hospitals, and the police; also that technical devices and mechanical aids are part of everyday life. The other awareness question (2) gave a positive view to application in education, as the students are becoming teachers. Examples of the major additions in responses in detail per question are in Table 7.

In the opportunities of ICT (3), the interesting increase was in the fields of personal and economical development. In question 4, students also identified important social and economic opportunities increasing the pace of development and improving communication in their country by means of a more general and creative use of ICT in society at large.

The questionnaire-based data shows that the number of responses increased and that very many of the student application ideas were outside of school life. The deeper analysis of the state of application ideas by an interview-based CATI analysis is taken up in Section 4.3.2.

4.3.2. *Application ideas in profession and society*

In this section, we present a CATI analysis of the thematic interviews administered to students after the course had ended. The focus on analysis was to define students' CATI level based on their learning in terms of their application ideas in their own profession and society in general. Each coding unit of each student was coded with

Table 6. The average number of application ideas identified at the beginning and at the end of Course I.

Question	No. of Responses (Average) Per Questionnaire (Q'aire 1)	No. of Responses (Average) Per Questionnaire (Q'aire 2)	Mean Difference	Significance of the Change
ICT awareness/ICTs in the students' environment				
1. Where is information and communication technology or robotics around you in your everyday life?	2.54 (SD 1.70)	3.12 (SD 1.14)	+0.58	0.053
2. Where could you apply programming or robotics skills in your society? How?	1.69 (SD 1.41)	2.12 (1.24)	+0.43	0.086
Opportunities presented by ICTs				
3. What are the opportunities for ICT in your life?	1.46 (SD 0.99)	2.19 (1.02)	+0.73	0.005
4. What are the opportunities for ICT in Tanzania?	1.38 (SD 1.10)	2.00 (1.10)	+0.62	0.015
Combined				
Questions 1, 2, 3 & 4 combined	7.08 (SD 3.53)	9.35 (3.17)	+2.27	0.002

Note: Responses of 26 students; SD = Standard Deviation.

a CATI level. Also, a general CATI level was defined for each student. In all, there were 189 coding units.

The following question themes were coded:

- Could you describe what you have done during this programming course?
- Could you apply your skills somehow in your coming work? If yes, how?
- Do you think the things that you have been learning or that have been taught during the course will benefit society somehow?
- Is there a relation between I-BLOCKs and Java, as you have studied both of those in the course? If yes, what kind of relation?
- What have you learnt? Describe your learning. What do you think was the most important learning experience during the course for you?

Our analysis, based on interviews coded in terms of the CATI model, shows (Table 8) that most of the students attained the application level (17). The second largest number achieved the transfer level (10). No students attained the levels of importation and contextualization. It should, however, be noted that 15 students' responses indicated contextualized elements whereas only five indicated elements of importation (Table 9).

Table 7. The major increases in ICT opportunities and awareness ideas.

Question and Categories	Answers in the Category (Questionnaire 1)	Answers in the Category (Questionnaire 2)
1. Where do you see examples of robotics or ICTs around you in your everyday life?		
Institutions (schools, hospitals, police)	4 (6.1%)	12 (14.1%)
Technical devices including (computers, cars, ships, trains, etc.)	9 (13.6%)	18 (21.2%)
Mechanical aid (carrying, moving etc.)	3 (4.5%)	5 (5.9%)
Others	50 (75.8%)	50 (58.8%)
Total number of answers	66 (100%)	85 (100%)
2. How could you apply programming and robotics skills in your society?		
Education (learning, teaching)	15 (34.1%)	21 (37.5%)
Mechanical aid (carrying, moving etc.)	5 (11.4%)	7 (12.5%)
Others	24 (54.5%)	28 (50%)
Total number of answers	44 (100%)	56 (100%)
3. What opportunities do ICTs give you in your life?		
Getting, delivering information, communication (inc. education)	17 (44.7%)	25 (42.4%)
Poverty reduction, econ. development, competitiveness	3 (7.9%)	10 (16.9%)
Personal development (e.g., creativity)	1 (2.6%)	4 (6.8%)
Others	17 (44.7%)	20 (33.9%)
Total number of answers	38 (99.9%)	59 (100%)
4. What opportunities do ICTs extend to your society?		
Getting, delivering information, communication (inc. education)	10 (27.8%)	13 (24.5%)
Poverty reduction, economic development, competitiveness	10 (27.8%)	18 (34.0%)
Others	16 (44.4%)	22 (41.5%)
Total number of answers	36 (100%)	53 (100%)

Note: 27 students altogether.

Most of the students are at the application level, although many students still functioned only on the transfer level. This indicates that they seem to be stuck on a conceptual level where learning application is still largely based on the uncritical memorization of information.

As half of the coding units reflected ideas on application, we hope that students will develop along these lines and progress toward higher levels. Of course, it may take students a long time to move from memorizing to application.

The following interview responses illustrate how we derived the level of student attainment, and they offer insight into how difficult it sometimes is to derive the correct level from a particular coding units.

Table 8. CATI levels in general.

Level	General CATI Levels Per Person	
	Number	Percentage
Contextualize	—	—
Apply	17	63.0
Transfer	10	37.0
Import	—	—
Total	27	100

Table 9. Distribution of coding units at different CATI levels.

Level	Coding Units at Different CATI Levels	
	No. of Coding Units	Percentage of all Coding Units
Contextualize	15	7.9
Apply	94	50.0
Transfer	75	38.9
Import	5	2.6
Total	189	100.0

> *Yeah, it will benefit if the ... If we have this computer program of Java, the society would benefit very much.* (*import level*)

This comment and its implications do not refer to or address any real-life situations or contexts although the student uses the word society. We derived an *import* level from this coding unit.

> *Yeah, my, my teach in case my school can introduce such a course at my school, I am able to teach, I can teach how to program in Java language because I have a mastered the important* (*course*). (*transfer level*)

The coding unit clearly shows an example of *transfer*. The student explains what he will do; he can teach a similar course without any change.

> *Okay, an important learning experience from the course. I have been interested in geometrical structures because I can transfer the knowledge of Java programming to my mathematics course such as on drawing different structures or graphs and other many things.* (*application level*)

The application of ideas means that one uses one's own understanding and knowledge to *apply* an idea. In this case the student intends to apply it in her mathematics teaching.

> *I can use these programs even now to teach the students because nowadays we have many things to teach the students, let's say for example in Java programming. Always we just deal with how to educate the students on budgeting.*

> *Therefore I can use this program to show them that if you need to do something, before doing it just go to the computer and just see if you can manage. Therefore I think it is very useful.* (*contextualization level*)

The level of contextualization is the most difficult to actualize in practice. This student reported on how he would apply and *contextualize* his teaching by using Java, and applying ICT for budgeting teaching (a new initiative) in his Tanzanian school.

The inter-rater reliability for the analysis of 189 codes by CATI levels was moderate; the value of Kappa was 0.62. The overall inter-rater reliability for students' general level CATI coding (27 codes) was high with a Kappa value of 0.81, giving a positive general insight of the similarity of the coding results and applicability of the content analysis method.

5. Discussion

Our analysis shows how the Introduction to Programming course developed and the role that its learning technologies played in its conceptualization, design, and successful implementation. In addition, the analysis of the design of the course shows how learning to use ICT can be successfully based on community and context-driven principles. This emphasis has become stronger during the development process as teachers and students began to notice the challenging (and often unexpected) opportunities for applying ICT in their local region.

It was encouraging to notice how readily our students devised application ideas. This is reflected in the many constructive comments that students made in the interviews about applications. Their enthusiasm gives us reason to hope that these students will continue to develop along these lines and progress to higher levels of application and contextualization. Of course, it is no easy transition to move from memorization to application. But a start has been made. The ultimate success of this kind of advanced education and the quest for sustainable development requires us to retain our long-term goals and perspective as we continue to strive for results that would eventually lead to development. The level of contextualization is the most difficult to attain. It was vitally important that at least some of our students should break through into this level. That some did is a cause for satisfaction.

Another interesting consequence of our work was that the course development and outcomes created several spin-offs in local communities. One example was that our students created educational robotic devices for some young patients in a local rural hospital. In addition, there was the opening of the first East African Science Park at IUCO, a newly established forum in which university and business collaborate to meet local needs, the TEDC conference, the first international conference to be organized on the premises of IUCO, and, finally, plans for a localized undergraduate program in ICT. It is therefore most important that skilled teacher trainees are encouraged and supported to continue their work in this field with new application ideas and skills.

We are aware of the risks of using the concept of ethnocomputing as both the design and analysis method of a course. But the analysis shows that ethnocomputing can serve as a viable, precise, and creative framework for guiding a design process of this kind so that representation, utilization, and appropriation are taken into account in a way that encourages learners to contextualize their learning process.

6. Conclusion

We applied an ethnocomputing-based approach to the design and analysis of an introductory ICT course at Tumaini University, Iringa University College, Tanzania. The emphasis of the analysis is on the development process of the course, the role of learning technologies in the course, and the contextual learning outcomes of the students. This study indicates that an ethnocomputing-based analysis can be used for designing needs-based ICT education by analyzing its feasibility with various operationalizations of the ethnocomputing concept.

The design process of the contextualized programming course reflected several perspectives of ethnocomputing. The student responses focused especially on the usage of I-BLOCK technology to support learning and understanding ICT. The digital learning material and Jeliot visualization tools did not seem to be so important for the contextual learning experiences. On the other hand, students' application ideas increased, and the digital learning material was designed to support these. There were a couple of students who reached the contextualization level elements based on the CATI model in the learning outcome analysis. The most frequent were application level ideas. Spin-offs in context and students' thoughts and ideas of application in a wider Tanzanian context were promising.

We developed and tested the use of the CATI model for a *community-driven* approach in its use of learning technologies. The model seems to support this approach to development, but the descriptions of level and divisions still need to be elaborated. The division of the application and contextualization levels to a greater level of detail, such as people, environment, and technology, requires further analysis.

The process of creating a contextualized ICT course is time-consuming and requires a great deal of patience and circumspection, as this case study from Tumaini University shows. Our work in Tanzania continues. One goal would be to create a contextualized Bachelor of ICT degree program at Tumaini University. But this task will require further elaboration of the ethnocomputing concept itself, as well as the design and analytical methods that accompany it locally.

Another issue related to the curriculum is that there might be a need for a more general ICT degree curriculum that specifies the content of Computer Science, Computer Engineering, Software Engineering, and Information Systems Science in a developing country. There could also be a need for a mixture with Engineering Sciences. This also refers to general needs of experts with a wide spectrum of understanding and skills in the theory of Computer Science and practical understanding of technology.

Tanzania is currently in the state of using ICT that existed a couple of decades ago in Western countries. Meeting the real needs with novel ICTs can also mean development of new business opportunities. Much good work can yet be done.

Acknowledgments

The authors are indebted to Marcus Duveskog, Minna Naumanen, Justus Randolph and Matti Tedre for their helpful and stimulating comments during the writing process. The financial support of the Academy of Finland (Grant 201682) is greatly appreciated.

References

ACM & IEEE (2001). Computing curricula 2001. Computer science volume. Retrieved September 3, 2006, from http://www.sigcse.org/cc2001/

Ashford, R. (1999). Internet Project Strategic Plan. Tumaini University, Iringa University College.

Bahendwa, F. (2004). Statistics of students who graduated at Tumaini University Iringa from 1998–2004. Tumaini University, Iringa University College.

Brennan, R. L., & Prediger, D. J. (1981). Coefficient Kappa: Some uses, misuses, and alternatives. *Educational and Psychological Measurement*, *41*(3), 687–699.

Eglash, R. (2004). Appropriating technology: An introduction. In R. Eglash, J. L. Croissant, G. Di Chiro & R. Fouché (Eds.), *Appropriating technology: Vernacular science and social power* (pp. vii–xxi). Minneapolis: University of Minnesota Press.

Haataja, A., Suhonen, J., Sutinen, E., & Torvinen, S. (2001). High school students learning computer science over the web. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning (IMEJ), 3*(2). Retrieved September 3, 2006, from http://imej.wfu.edu/articles/2001/2/04/index.asp

Levy, B.-B., Ben-Ari, M., & Uronen, P. (2003). The Jeliot 2000 program animation system. *Computers & Education*, *40*(1), 15–21.

Lund, H. H. (2003). Intelligent Artefacts. In M. Sugisaka & H. Tanaka (Eds.), *Proceedings of the 8th international symposium on artificial life and robotics (ISAROB)* (pp. I11–I14), Oita, Japan. Oita: ISAROB.

Lund, H. H., Nielsen, J., Sutinen, E., & Vesisenaho, M. (2005). In search of the point-of-contact: Contextualized technology refreshes ICT teaching in Tanzania. In P. Goodyear, D. G. Sampson, D. J. Yang, Kinshuk, T. Okamoto, R. Hartley & N. Chen, *Proceedings of the 3th International Workshop on Technology for Education in Developing Countries (TEDC 2006)* (pp. 983–987), Taiwan. Los Alamitos: IEEE.

Neuendorf, K. (2002). *The content analysis guidebook*. London, Thousand Oaks & New Delhi: Saga Publications.

Task Force on Science, Technology, and Innovation. (2005). *Innovation: Applying knowledge in development*. London: Earthscan & UN Millennium Project.

Tedre, M., Sutinen, E., Kähkönen, E., & Kommers, P. (2006). Ethnocomputing: ICT in social and cultural context. *Communications of the ACM 49*(1), 126–130.

The United Republic of Tanzania. (2002). Population and housing census general report. Retrieved September 3, 2006, from http://www.tanzania.go.tz/census/iringa.htm

The United Republic of Tanzania. (2005). The President's Office – Planning and privatization. The Economic Survey 2004. Developments in different sectors of the economy. Education. Retrieved September 3, 2006, from http://www.tanzania.go.tz/economicsurvey1/2004/part4/education.htm

Tumaini University (2006). Iringa University College. Objectives. Retrieved September 3, 2006, from http://www.tumaini.ac.tz/objectives.htm

Tumaini University (2005). Prospectus 2005–2007. Tumaini University, Iringa University College.

United Nations General Assembly. (2000). United nations millennium declaration. Retrieved September 3, 2006, from http://www.un.org/millennium/declaration/ares552e.pdf

Vesisenaho, M. O., & Lund, H. H. (2004). I-BLOCKS for ICT education development, case Iringa, Tanzania. In F. Flückiger, R. Ruprecht & R. Scheurer (Eds.), *Proceedings of the 33rd International Symposium IGIP/IEEE/ASEE* (pp. 364–371), Fribourg, Switzerland. Fribourg: University of Applied Sciences of Western Switzerland.

Vesisenaho, M., Duveskog, M., Laisser, E., & Sutinen, E. (in press). Designing a contextualized programming course in a Tanzanian University. Accepted for Proceedings of 36th Frontiers in Education Conference (FIE 2006), October 28–31, San Diego, California, IEEE.

Vesisenaho, M., Kemppainen, J., Islas Sedano, C., Tedre, M., & Sutinen, E. (2006). How to contextualize ICT in higher education: A case study in Tanzania. *African Journal of Information & Communication Technology (AJICT)*, *2*(2), 88–109.