Mobile Blogs in Education: Case of ViSCoS Mobile

Teemu H. Laine

14.1.2007

University of Joensuu
Department of Computer Science and Statistics
Master's thesis

Abstract
Distance learning programmes often encounter problems with student motivation and performance. In the case of ViSCoS (Virtual Studies of Computer Science) programme at the Department of Computer Science, University of Joensuu, these problems have been recognised to stem partly from lack of strong community. Furthermore, distance learning is not always location- and time-agnostic; studying is often confined to a specific location and time. In this study, we present the idea of a mobile version of ViSCoS where students are granted a true anywhere, anytime learning experience. In this scheme, mobile blogging is recognised to be a part of student support services facilitated by mobile technology. Mobile blog usage scenarios for ViSCoS are discussed and students' motivation considered from moblogging point of view.

Blogging is a powerful learning tool with various usage possibilities. When blogging is ported to a mobile environment, the resulting application can be used efficiently to record events and process learnt knowledge in real time. Several existing solutions for blogging offer a good basis for mobile blog development. Mobile blog gateway is required to transfer messages from a mobile device to the blog system. In this thesis we analyse existing blog system solutions and mobile blog gateways in order to determine the best way to build a mobile blog system for needs of ViSCoS. Prototype of a new mobile blog system based on existing solutions is constructed and evaluated by two usage tests.

Based on the findings, we provide recommendations for mobile blog usage and development for distance learning environments. Results of this work are useful in deployment of mobile blog system for ViSCoS in the future.

The ACM Computing Classification System:
K.3.1, K.3.2, H.4.3

Keywords:
blogs, blogging, mobile blogs, moblogging, mobile learning, distance learning, ViSCoS

Contents

Chapter 1

Introduction

Finland is a country of vast geographical size and a low number of inhabitants. Direct consequence to these facts is that the country has sparsely distributed population. For the past decades, transition from an agricultural society to an urban society has taken place, resulting to the young generations abandoning the countryside and moving to the cities. However, there still remains a large number of sparsely populated rural areas in Finland where maintaining a high level of education can be seen as a major problem for the general development of the area. This is the perfect setting for distance learning to take place. If students living in rural areas could study the same knowledge as the students living in larger cities, we could achieve educational equality. This scenario is not only possible in Finland, but in many other regions, particularly in Africa. As we see it, demand for distance education, particularly in the field of ICT and computing, is growing rapidly.

The Department of Computer Science at University of Joensuu launched ViSCoS project [68] in 2000. ViSCoS is an acronym for Virtual Studies of Computer Science and the idea of ViSCoS programme is to offer first year computer science courses on distance learning basis. At first, the target group was upper secondary school students in Joensuu region who were considering of studying computer science in the future. In the fall of 2004, ViSCoS project started collaboration with the Continuing Education Centre in the University of Joensuu, bringing ViSCoS studies within reach of everybody in Finland. In the fall of 2006, ViSCoS took one step further and launched the whole programme in English, thus offering possibility for anybody from any location to attend the ViSCoS courses.

Students in ViSCoS take control of their own studies, meaning that they can set their own schedule of when to study, when to take the examination, etc. This means that the time is not restricted unlike it often is in contact learning. For example, according to the observations of ViSCoS instructors, some students tend to submit answers for weekly

exercises in the middle of night.

Another unrestricted element in ViSCoS is location - student can study from any location where a computer with connection to the Internet exists. However, the location of computer can also be a restrictive element; if a student is able to use a computer only at school, computer laboratory opening hours effectively limit the time used in studying. Even if a student would have a computer at home, he would not be able to study at *any time, anywhere* as desktop computers are not portable.

From portability arises a novel idea; why not to use portable mobile devices to enable any time, anywhere learning experience in ViSCoS programme? Technical infrastructure is well established as nearly every student in Finland owns a mobile device, some even have several of them. *ViSCoS Mobile*, as the new version of ViSCoS was baptised, consists of several development threads, each having specific problems to be solved [41]. This work concentrates on design, implementation and usage of *mobile blog*, or *moblog*, as a part of student support services in a distance education setting. Shortly put, blog is a powerful tool for reflecting and processing ones thoughts [18]. Mobile blog usage in distance education is a novel idea and there has not been much research regarding this topic. Using mobile devices in education effectively removes spatial and temporal restrictions which are firmly present in ordinary classroom settings. In similar fashion, mobile blogs allow users to update the content whenever and wherever, thus enabling real-time processing of spontaneous, fresh ideas.

In order to establish a moblog environment for ViSCoS Mobile, we first analysed existing solutions. According to the analysis, we chose to use existing blog system and to develop a moblog plugin for that system. The new moblog plugin, named *Advanced Postman*, was based on e-mail gateway solution which, according to our analysis, was the most feasible solution for ViSCoS at the time. After designing and constructing the system, we ran preliminary tests in order to discover errors, missing features, and to gather new ideas of how moblogs could be used in ViSCoS Mobile.

1.1 Motivation

Currently there are large number of distance education programmes around the world. Many of these programmes are based on correspondence, i.e. communication between a student and a teacher is done via traditional mail. For example the Association of British Correspondence Colleges [56] maintains a portal for 23 distance learning colleges offering correspondence-based courses in the UK. In addition to correspondence courses, there are quite many Internet-based distance education programmes established as well, one of them being the ViSCoS programme at the University of Joensuu. One of the major

benefit of using the Internet in studying is the increased communication and interaction between course participants, including the instructors. The Internet not only enables faster message delivery (e-mail, instant messengers) and delivery of rich multimedia content and educational tools, but it also provides an oasis for online communities. Most of these Internet-enabled distance education programmes require students to have access to a computer with connection to the Internet.

Motivation for this study comes from the idea of bringing education to mobile devices. Essentially, this means that students use mobile devices to access educational content anywhere and at any time. Ordinary blogs have been used successfully in a traditional classroom setting (e.g. [7], [33], [62]) and this success of blogs can only be enhanced when the user is given freedom to update the contents of a blog regardless of time and location, thus enriching the learning experience. Furthermore, lack of discourse between students in ViSCoS has been a major problem and a probable cause for the drop outs [50]. This problem can be partly solved by adding tools of social software, such as mobile blogs, within reach of the students.

It is obvious that properties of the current mobile devices are not optimal for learning. This thesis attempts to identify the most crucial incapabilities of mobile devices and suggests methods to overcome them. If the limits of mobile devices in regard to education are successfully solved, then the portability advantage of mobile devices will definitely make mobile learning more powerful.

Prensky [59] has recognised that the following learning processes can be supported through mobile phones: listening, observing, imitating, questioning, reflecting, trying, estimating, predicting, speculating and practising. Traditional classroom environment often supports only a subset of these. In a traditional setting, reflecting and speculating are often forgotten beyond a monologye of the lecturer. While some ViSCoS courses currently use learning logs and discussion forums, they are not popular among students. Mobile blog could be the right tool for breaking the ice and escalate the adaptation process of novel learning technology among the students in distance education.

Mobile blogs could be used in ViSCoS on several courses. For example on programming courses, students could post alternative ways to solve programming problems, to ask help from others and just to write about their experiences and feelings. On Introduction to ICT and Computing course, students could post observations of new technologies and innovations to their blogs. On Introduction to the Ethics of Computing course, students could use mobile blogs to establish active ethical discourses. Finally, on programming project course they could use their mobile blogs to reflect their learning, post fresh ideas of design and implementation of the project, and discuss with other students/instructors about possible solutions to the current problems. New ways to use mobile blogs are likely to emerge as students are provided with technology to study wherever and whenever they like. The main objectives of bringing mobile blogs to ViSCoS are to enrich the interaction

within the ViSCoS community by establishing a blogging culture, and to enhance the learning processes of the students.

1.2 Terms and Concepts

The Instructional Technology Council (ITC) defines *distance learning* as, "the process of extending learning, or delivering instructional resource-sharing opportunities, to locations away from a classroom, building or site, to another classroom, building or site by using video, audio, computer, multimedia communications, or some combination of these with other traditional delivery methods." [13]. In this thesis distance learning is represented by ViSCoS project and particularly Mobile ViSCoS.

Digital Learning Environment (DLE) is a virtual classroom where students can carry out exercises, learn new material, interact with the teacher and each other, and even take examinations. Usually DLE's are accessed over the Internet which makes them available almost anywhere. Digital learning environment is sometimes referred as a *virtual learning environment* (VLE).

IMPIT (International Master Program in Information Technology) is international master program at the Department of Computer Science, University of Joensuu. Studying in IMPIT is implemented as normal lectures and examinations.

Moodle is an open source DLE, thus it is free of charge and available for anyone for further development. Moodle is used in ViSCoS studies as the main learning arena where students retrieve material, return weekly exercises, receive feedback and interact with other members of the course. Single Moodle installation can run multiple courses in multiple course categories, and the system is expandable by a powerful plugin mechanism.

The acronym *ViSCoS* stands for Virtual Studies of Computer Science, a distance learning computer science programme at the Department of Computer Science, University of Joensuu.

Blog, or weblog, is an online diary where entries are typically displayed in reverse chronological order. Blogs are typically accessed with a web browser on the Internet. Blogging is an activity in which a *blogger*, the blog author, writes entries to his/her blog.

Moblog, or *Mobile Blog*, is essentially similar to blog and the only difference is the method of delivering an entry. Sometimes terms moblog and blog are mixed and particularly moblog is often referred as blog. The term moblog is used in this thesis to refer to a weblog where entries are posted by a mobile device. In similar fashion, the term

moblogging is used to refer to action of sending an entry to a weblog by a mobile device. The verb *to blog*, however, is used to refer to any general form of blogging action, including moblogging.

Mobile learning, or *m-learning*, is a type of learning that occurs "on the road". It simply means that the learning process is activated outside the classroom by means of mobile technology. One could consider a traditional book also as a tool to enable m-learning; a book can be carried anywhere and it can be used to learn new theory. In this thesis the term mobile learning is used for technology-assisted learning outside the ordinary classroom.

According to Wikipedia ([24])*Application Programming Interface* (API) is "the interface that a computer system, library or application provides in order to allow requests for services to be made of it by other computer programs, and/or to allow data to be exchanged between them".

Blogging engine is the core of blogging software that stores and retrieves blog entries to and from blog database. *Moblogging engine* is usually implemented on top of ordinary blogging engine with additional gateway to process incoming messages from mobile devices.

Extensible Markup Language (XML) is a general-purpose markup language for describing the data. Unlike more commonly known markup language HTML, XML is not used to describe representation of the data.

Instant Messenger (IM) is a communication tool which enables instant communication by short messages sent between two or more peers. There are several well-known IM protocols, for example MSN and AIM/ICQ.

J2ME is Java platform for mobile devices. It is possible to run a single J2ME program on devices of different manufacturers, if they have Java Virtual Machine for J2ME installed.

JVM is acronym for Java Virtual Machine. It is a piece of software that is used to enable Java programs to run on given operating system. Thanks to JVM, Java programs are nearly platform independent because once compiled program can be executed on any platform having appropriate virtual machine installed.

Mobile device, or handheld device, is a small computing device, typically having a small colour screen, miniature keyboard and some communication facilities. Common example of mobile devices are *mobile phones* and *smart phones* which have become extremely popular around the world. Also *PDA* (Personal Digital Assistant) belongs to the group of mobile devices. Today mobile phones and PDAs are converging, and perhaps in the future there are only mobile devices which facilitate the best features of each. Newest addition to the group of mobile devices is *Ultra-Mobile PC* (UM-PC) which is a device slightly

larger than a mobile phone, yet capable of running a full-scale PC operating system such as Windows XP.

RFID (Radio Frequency IDentification) is a technique for automatic identification based on RFID tags and corresponding readers. RFID tag can be attached to any product. Communication between tags and readers is established by microwaves.

1.3 Structure of the Thesis

This thesis concentrates on mobile blogs which are a part of ViSCoS Mobile concept. Chapter two starts by presenting the essential research questions and methodology used to answer these questions. Chapter three presents the ViSCoS programme and its latest development branch, ViSCoS Mobile. Chapter four conducts a literature review on existing research results and theories on blogs and moblogs in education. General information of blogs and moblogs is also covered. The results are used in planning and implementation of mobile blog system for ViSCoS. Chapter five starts by presenting an overview of existing blog solutions together with moblog gateway solutions. Then the results of design and implementation of Advanced Postman, a moblog plugin for Nucleus CMS blog system, are presented. Results of analysis and evaluation of tests performed on Advanced Postman are presented and discussed in chapter six. Finally, chapter seven concludes the work by answering the research questions and providing information on the future work. Figure 1.1 presents structure of the thesis in graphical format. Arrows denote connections between the chapters. Conclusions and Future Work chapter would be actually connected to most of the previous chapters, but we left out those connections in order to keep the diagram legible.

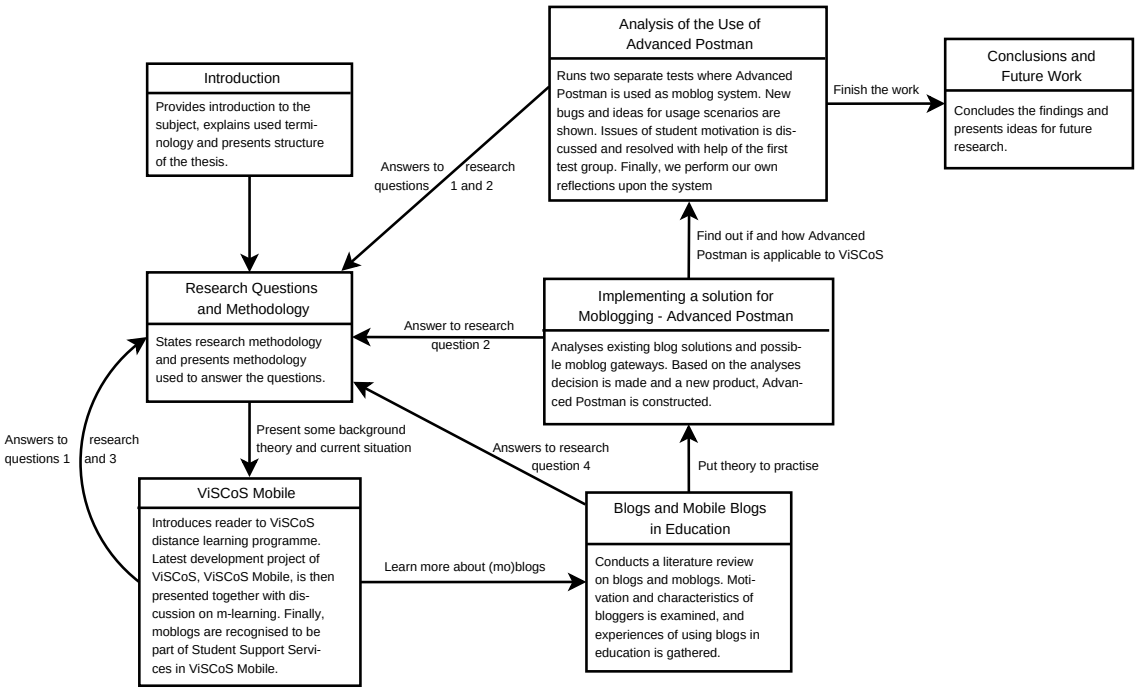

Figure 1.1: Thesis structure

Chapter 2

Research Questions and Methodology

This chapter begins by presenting the essential research questions that this study seeks to answer. Then, the research methodologies used in this thesis are described.

2.1 Research Questions

The first question for this research is how mobile blogs can be used in a computer science distance learning course to increase motivation and collaboration of the students in order to build a stronger learning community? This question requires descriptions of possible usage scenarios and suggestions for instructors and for students. Also preliminary usability results are required to estimate the potential of moblog as a learning tool.

The second question derives from the first one, stating: what is the most feasible way to construct a mobile blogging environment for ViSCoS? The moblogging environment should be easy to use, yet offering possibility for customisation. Existing solutions must be taken into consideration. An important question lays in how blog entries are delivered from a mobile phone to the blogging environment. Incapabilities of mobile devices are apparent, and it is yet unclear how these would affect on moblogging in a virtual classroom, thus this should be cleared out.

The third question is: what are the needs in ViSCoS Mobile and how does this work fulfill these needs? This thesis presents the idea of ViSCoS Mobile, its needs and finally it reveals a solution to one of the presented needs.

The fourth question concerns existing work: how blogs and moblogs have been previously used in education and in research.

2.2 Methodology

This thesis uses three generally approved scientific methods. First of these is literature review. It attempts to discover what already has been researched about usage of blogs, particularly mobile blogs, in education. The literature review section also discusses who bloggers actually are and what their motivation is. After reviewing existing literature, we were able to answer to the third, fourth question, and give a partial answer to the first question.

For the literature review several sources were used to retrieve results of existing research. Digital libraries of ACM and IEEE were the most useful resources, but also article resource lists (e.g. [32], [47]) constructed by other researchers were much used. There were several individual journals having useful articles, e.g. Educause Review, Journal of Online Education and Australasian Journal of Educational Technology. There seem to be a few individual authors that have been working much with blogs in the name of research. Some of the most distinguished names on this area are Will Richardson, Jill Walker, Bonnie Nardi, Lilia Efimova and Rebecca Blood.

The second method is the design and construction of a mobile blog system. To achieve this goal, multiple existing blog system solutions were analysed and the best solution was chosen as the backbone for the new system. The mobile blog system was then constructed by adding necessary features to the existing blog system. While designing and constructing the system, weaknesses and strengths of different technological options were evaluated. By using this method we found the answer to the second question.

The last research method is evaluation and analysis of the constructed mobile blog system prototype. For this purpose two separate usage tests were performed. In both tests a group of students were equipped with mobile devices and accounts to mobile blog system. Students then used the mobile devices to post messages to moblogs. In the first test each student had personal moblog while the second test incorporated a single moblog for the whole group. User feedback was collected in qualitative manner both by analysing the moblog entries and by asking related questions from the participants. The goal of the first test was to map the characteristics and problems of Advanced Postman in order to enhance it further. In addition, the first test was conducted to develop ideas of how moblogs could be used in ViSCoS in order to create stronger study community. This approach was taken in order to establish *participatory design* in which users of the system are involved in the design process. In the second test the goal was to evaluate moblog as an event logging tool for programming project course. The evaluation and analysis method gave a partial answer to the first and the second research questions.

We can depict the connections between different methodologies in a work process illustration presented in figure 2.1. The work process starts from analysis where the litera-

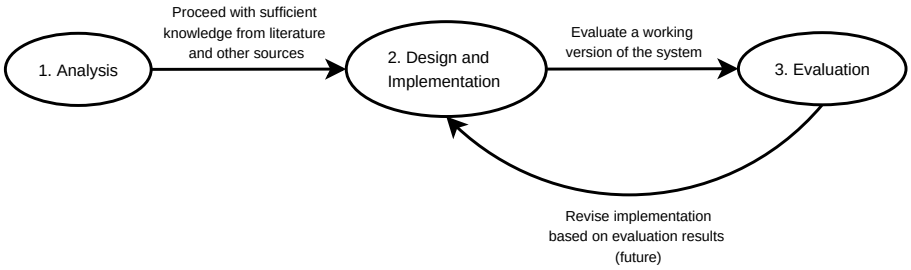

Figure 2.1: Illustration of the work process

ture review and background information are analysed. Based on gathered information and recognised needs, we proceed to the implementation phase where technological options are discussed and finally new moblog system is constructed based on discovered knowledge. Last phase, evaluation, tests the system in order to evaluate its strengths and weaknesses. It is possible to return from the last phase to the implementation phase, if the system requires revision, which is likely to happen in the future.

Chapter 3

ViSCoS Mobile

This chapter presents ViSCoS distance learning programme and the effort of bringing the entire programme into mobile devices. Opportunities and challenges of mobile learning are recognised, and solutions to discovered problems are suggested based on existing work. The implementation plan of ViSCoS Mobile is reviewed from three developmental viewpoints. The main topic of this thesis, mobile blogs, are connected to the viewpoint of student support services.

3.1 ViSCoS Programme

ViSCoS, Virtual Studies of Computer Science, is a distance learning programme in which first year computer science courses are taught to students via the Internet. The target length of the ViSCoS programme is 1.5 years, but individual performance of each students may vary. Courses available in the ViSCoS curriculum in 2005 are presented in Table 3.1. Credits earned from each course are presented in standard ECTS units (European Credit Transfer and Accumulation System). Semester column states the semester on which the course is usually taken. For example, Fall II means the second fall semester.

Table 3.1: ViSCoS Curriculum

Course	Credits	Semester
Introduction to Information and Communication Technology (ICT) and Computing	2 ECTS	Fall I
Programming I	3 ECTS	Fall I
Programming II	4 ECTS	Spring I
Research Fields of Computer Science	2 ECTS	Spring I

Continues on next page...

Table 3.1: ViSCoS Curriculum (continued)

Course	Credits	Semester
Programming Project	4 ECTS	Spring I
Discrete Structures	5 ECTS	Fall II
Hardware, Computer Architecture and Operating Systems	3 ECTS	Fall II
Introduction to the Ethics of Computing	2 ECTS	Any semester
TOTAL	**25 ECTS**	

3.1.1 History

The ViSCoS programme started in the year 2000 at the Department of Computer Science, University of Joensuu. At first the programme was offered to upper secondary school students in the surrounding province of Joensuu, namely North Karelia. The idea was to motivate the students to continue studying computer science at the University of Joensuu after finishing their studies at upper secondary school. In the year 2001, the ViSCoS programme expanded to upper secondary schools in the province of South Savo. The following year ViSCoS was introduced at the Päivölä Adult Education Institute, enabling mathematically talented students to study computer science in distance learning manner. From the year 2004, ViSCoS was integrated into curriculum of the Continuing Education Centre in the University of Joensuu, thus giving possibility for anyone in Finland to study any of the ViSCoS courses.

From the very beginning until today ViSCoS project has been a target for research. Suhonen's FOrmative DEvelopment Model (FODEM) [67] was used to develop the ViSCoS programme. FODEM is a light-weight development method for digital learning environments useful particularly in sparse learning communities. Several research papers (e.g. [28], [68], [41]) concerning ViSCoS have also been published. Furthermore, ViSCoS project has had an effect on development and evaluation of several learning tools such as Woven Stories [25], Ethicsar [34] and Jeliot [46]. Student feedback and performance figures are gathered continuously. Captured data is processed further by means of data mining in order to detect and prevent potential drop outs, for instance [30]. This thesis belongs also to the group of ViSCoS research. The aim of all the research centred around the ViSCoS project is to enhance course material, tools and methods to facilitate better distance learning experience.

3.1.2 Studying in ViSCoS

Learning computer science in ViSCoS relies heavily on independent studying, as is the case with most distance education programmes. There are no traditional lectures nor any other contact teaching, except arbitrary summer schools and seminars. ViSCoS does not provide synchronous online lectures which are sometimes featured in distance learning programmes. Students must therefore study the material on their own and then do exercises for practising the learnt theory. Instructors, who are competent university teachers, evaluate returned exercises and give appropriate feedback. Furthermore, students are encouraged to consult instructors and other students if they encounter difficulties while studying. Examinations are taken remotely. ViSCoS programme has several partner institutions such as upper secondary schools, adult education centres and universities, which arrange examinations simultaneously on an agreed schedule. Questions for examinations are prepared by the course instructors, whose responsibility is also to correct the returned answers.

3.1.3 ViSCoS in English

To address the demand of globalisation in computer science education, ViSCoS programme in English was launched in September 2006. The aim is to provide ViSCoS studies for the English-speaking population in Finland. First year can be seen as a pilot phase and should it proves to be successful, the ViSCoS programme in English is likely to be officially expanded beyond the Finnish borders. It is believed that there is a huge number of potential students in sparsely populated countries particularly in Africa and other developing areas. In the pilot phase of the ViSCoS in English there are five students of whom three are located in Finland and two in Zambia.

3.1.4 Supportive Learning Tools

ViSCoS programme utilises a set of learning tools to facilitate the students' learning processes. These tools are continuously under development and feedback gathered from the students is an important resource for improvements. This section presents the tools that have been used in ViSCoS programme and tools that are under development.

Students and instructors in ViSCoS use the open source *Moodle learning environment* [15] as a virtual classroom. Moodle enables management of multiple courses in multiple categories. Furthermore, Moodle is continuously under development and new features added from time to time based on the feedback received from the users. Moodle's powerful plugin API (Application Programming Interface) makes it easy to modify existing modules

and add new functionality. Moodle's plugin library is relatively large; featuring useful plugins such as questionnaires, activities and administrative tools. Since Moodle is open source, the ViSCoS team has performed some modifications to its source code in order to allow custom authentication and other features concerning integration to other systems.

Jeliot [46] is an open source program visualisation tool for Java programming language. It helps novice students grasp the idea of programming execution by means of animation. Jeliot is used particularly on Programming 1 course where important basic structures of Java programming are taught. Jeliot's newest version is Jeliot 3 and it can be downloaded at no cost from http://cs.joensuu.fi/jeliot.

Ethicsar [34] is a web-based ethical argumentation tool which has been used on Introduction to the Ethics of Computing course in ViSCoS. Students can use Ethicsar to discuss and analyse ethical problems in a collaborative manner. The goal of Ethicsar is to help students initiate personal ethical thinking by gaining new insights from the fellow students.

LEAP (LEArning Process companion) [37] is a digital learning tool which was developed for students to support management of the Programming Project course in ViSCoS. LEAP can be used with a web browser and it has two main functions; it works as a learning portfolio and as a creative problem solving support. LEAP has been used by students of the Programming Project course to manage their projects and to reflect their learning. The latter activity overlaps with the goal of mobile blogs.

In addition to e-mail, for communication between students and instructors, online conversation tools such as *IRC (Internet Relay Chat)* and *Skype* (Internet telephony application) are used. Instant messaging services have been established and students have been encouraged to contact instructors via them, but so far only a few students have utilised this opportunity. Ordinary telephone is also used to give individual guidance during the Programming Project course.

The blogging tool was first introduced for ViSCoS students first time in the Fall of 2006 with the use of Nucleus CMS blogging platform. The aim is to guide students to use blogs as learning journals to reflect their learning processes and write down new ideas. Commenting on other students' blogs is also encouraged. Mobile blogging extension is developed during this thesis and it is expected to be deployed officially by 2007.

3.1.5 Acute Challenges in ViSCoS

Despite many research projects there still remain many issues to be solved in ViSCoS. Probably the most difficult and serious of these is the high rate of drop outs particularly on programming courses. Part of the drop outs can be explained by the fact that many

new students do not know what programming is and they simply attend the course to find out. After learning that programming requires a specific way of thinking, they decide to drop out. Ongoing data mining research attempts to tackle the problem of drop outs. Furthermore, course materials are continuously enhanced according to feedback gathered from students. Every year some improvements have been made, but nevertheless it seems that achieving 100% success is impossible as there will always be some students who drop out for one reason or another. Currently part of the research work concentrates on establishing blogging culture in ViSCoS community in order to help students to be more interactive, thus gaining more motivation for success.

Another problem in ViSCoS distance learning programme is lack of communication between students and instructors as well as among students. Although Moodle learning environment offers tools for collaboration, such as forums and chats, it is very hard to motivate students to start using them. Perhaps it is the characteristics of a Finn that restrain students from being more active. Most of the ViSCoS courses have mechanism for writing and receiving feedback. Students are given the opportunity to write self-evaluations and instructors write personal feedback to these evaluations. Instructors also write comments and feedback for every single exercise submitted by the students. Use of telephone guidance has also been used on Programming Project course. Despite all these communication channels interaction between instructor and students is far from that of a traditional classroom environment.

Distance is probably the most obvious reason for aforementioned problems in a distance learning setting. Nagel et al. [50] perceived lack of community as the most significant single reason as to why unsuccessful students in online study programme adandoned their courses. The same study also suggests that the importance of community is highly appreciated by those who are successful in their online studies. In order to overcome the challenge of belonging to a community, the effect of distance must be dissolved by means of technology so that students and instructors can feel as part of a classroom community. This thesis proposes mobile blogs, or blogs in general, to be used in order to bring the active virtual classroom one step closer.

Thanks to partner institutions of ViSCoS, currently the examination system works well inside Finland. However, the demand for ViSCoS studies is likely to grow on international arena and this will pose problems in arranging examinations simultaneously in every location. The solution to this challenge lays in alternative evaluation methods and online examinations. These possible solutions should be reviewed and implemented before ViSCoS can efficiently provide distance education outside Finland.

3.2 Opportunities and Challenges of M-Learning

This section presents opportunities and challenges of mobile technology in the field of education for the purpose of identifying implementation challenges in ViSCoS Mobile. The challenges are presented together with suggested solutions.

3.2.1 Mobile Technology and Learning

Mobile technologies in context of learning have been under active research since the appearance of the first mobile phones. The first applications were based on early technological innovations such as SMS messaging and simple voice based applications. Today, thanks to rapid technological advancement, mobile devices are much more than phones; they are multimedia computers. In this subsection we illustrate how these technologically powerful mobile devices can be utilised in education, and particularly in distance education. Motivational issues are also discussed.

Java MIDlets, small Java applications runnable on Java-enabled mobile phones, have been used in several mobile learning projects in past few years. For example Cruz Mayorga-Toledano and Fernandez-Morales [44] have done research on using MIDlets as learning tools in higher education. According to their results, Java MIDlets are particularly useful for interactive tests, calendars, glossaries and other small applications. MIDlet technology can also be used to create cross-platform educational software such as games, exercises, collaboration software, and productivity tools.

In addition to MIDlet applications mobile phones have potential for much more. Multimedia capabilities of new mobile phones enable media creation and processing on the road. New powerful graphic processing chipsets are able to render 3D graphics, just like new personal computers were able to do ten years ago. Fast connection to the Internet by means of WiFi, 3G (4G in the future) and WiMAX provide a gateway to endless information flow. Prensky [59] has gathered the capabilities of mobile devices and mapped them with possible usage scenarios in education. Table 3.2 presents these usage scenarios together with the respective mobile device capabilities. Even more usage scenarios emerges, if one mixes these capabilities together. Prensky's list is not exhaustive, but it gives a good impression of what mobile devices are capable of in the field of education. The table does not list the newest features such as light and motion sensors in mobile devices. These futuristic properties could be used for example to create interactive learning games or simply to control the device.

Table 3.2: Usage scenarios for mobile devices in education (Prensky [59])

Mobile device capability	Usage scenarios in education
voice	language training, voice recognition (authentication), voice lectures, voice-activated services
short text messages (SMS)	reminders, motivator messages, polls, games, quizzes, news, training for exam
graphic display	reading text, flash cards, pictures, comics, animation (with sound), 3D content, holograms
downloadable programs	new educational software can be downloaded and used; possibilities are limitless
Internet browser	browse web content, use dictionary, thesaurus, encyclopedia, and gather material (text, image, video) with search engines,
camera	data collection, documentation, visual journalism, remote-operated observations
Global Positioning System (GPS)	learn about locations, orientation, treasure hunting games, research tool especially in geography
video	television journalism, creative movie-making, educational clips

Eronen [21] has discovered five pedagogical models suitable to be used in mobile learning. These models are Situated Learning, Problem-based learning, Project-based learning, Performance-based learning and Tutorial learning. Table 3.3 presents these models and roles of technology respective to each of the models. Last column presents examples for each model of how mobile technology supports learning activities. From these models particularly problem-based learning, project-based learning and tutorial learning are currently used in ViSCoS courses.

Table 3.3: Pedagogical models of mobile learning (Eronen [21])

Pedagogical model	Roles of mobile technology	How mobile technology supports learning
Situated learning	Contextualising and information explication, data collection, referential, interaction and collaboration	- Allows access to materials and exercises in digital learning environment (DLE), such as Moodle. - Supports recordings of daily activities and uploading the material to DLE - Supports moblogging and discussions with other students of the course - Supports collaboration with other students for example in group tasks
Problem-based learning	Collaboration and problem-solving support, interaction, referential, microworlds	- Allows access to problems and their supportive materials - Supports communications within the group and the people outside of it - Facilitates problem solving and hypothesis creation with suitable software - Supports creation of conceptual presentations and multimedia
Project-based learning	Interaction, collaboration, data collection, administration, artefact creation	- Allows access to supporting materials throughout the project work - Supports in mindmapping and concept mapping - Offers tools for artefact creation and project management - Supports groups' and individual learner's reflective actions - Supports moblogging and discussions between the project group
Performance-based learning	Location-awareness, referential, collaboration, administrative and communication support	- Keeps track of the whereabouts of the students - Keeps track of the performance of the students - Allows access to learning materials related to a particular learning task - Offers learning interventions and assistance when necessary - Supports both real-time and forum-based (moblogs) communications between the students
Tutorial learning	Administration, collaboration, communication, interaction, microworlds	- Keeps student logs and follows their progress (e.g. with moblogs) - Supports communication with the system, fellow students and teacher - Offers interactive elements for the learner to support the learning process

Koole's [38] article presents suggestions on how m-learning can be used successfully despite the limitations of mobile devices. Based on Koole's article, we gathered and

refined requirements for successful educational mobile services. These requirements are presented and explained in table 3.4. The second column provides further explanation to the requirements. Actual limitations of mobile devices are presented in the next section.

Table 3.4: Requirements for successful educational mobile services (Koole [38])

Requirement	Explanation
Simplicity	Simple user interface and functionality guarantees good usability.
Automate functionality	By automating functionality as much as possible, we can reduce the steps required to use the service, thus making usage faster and more effective.
Reduce number of actions	This is closely related to the previous requirement. By reducing the number of actions the service becomes faster and more efficient to use, and it will be easier to learn as well.
Reduce cognitive load	By cognitive load we mean the amount of information the user must keep in his short-term memory while using the service. For educational services this factor is crucial as students should put their thoughts on the actual learning material rather than on deciphering the system behaviour.
Prevent disorientation	Navigation should be so clear and simple, that user knows always her location in the service.
Reduce to essentials	By showing irrelevant information, student's cognitive load increases. Therefore we should strive to provide only the essential data to m-learners.
Use summaries, annotations, advanced organisers	These techniques should be used to logically connect *chunks* of course material together. In chunking material is divided into small, compact units, which are then studied from mobile devices. Without appropriate connections the chunks do not form the big picture.
Knowledge navigation	Mobile service should offer tools for knowledge navigation, including social interaction with peers (information exchange). Knowledge navigation feature is important in courses where students are required to gather information from online sources, for example.
Focus switches	Switch focus in order to keep the learner active and learning experience interesting. In pervasive service this would mean changing focus from device to the surrounding environment and back.

One of the major problems in distance education is how to motivate student to learn and how to maintain the motivation. Attewell [4] performed a mobile learning research on a group of young students who were in danger of dropping out of their institutions, or had already dropped out. Attewell's results states that mobile learning can attract young people to learning, to maintain their interest, and to support their learning and development. Perry [57] also noticed boost in motivation while researching effect of PDAs in schools. It seems that new technology is a good motivator particularly with

children and young adults. This is an interesting observation as the majority of ViSCoS students are under 25 years old.

3.2.2 Identified Limits and Solutions of Mobile Devices

Small screen, cumbersome input system, insufficient processor speed and slow network connection are the major obstacles of using mobile devices successfully for learning. In addition, high cost of data transmission rates are problem in many countries world wide. Since the focus of this thesis is on mobile blogs, the identified limits and solutions are discussed from the mobile blogging point of view, but they can be generally applied to any other event of m-learning. This section provides ideas on how to overcome discovered obstacles by means of new innovations and customisation of the moblogging environment.

Tiny Screen Syndrome

The problem of tiny screen comes forth when long texts and large images are viewed or produced. This problem can be efficiently tackled by means of *content adaptation*. When result of content adaptation is insufficient, one can consider using a device akin to Sony VAIO UX50 Ultra Mobile PC (UM-PC) which is a portable version of a personal computer running a full scale Windows XP operating system. With UM-PC moblogging and other mobile learning activities are fast, efficient, portable and most of all enjoyable. Another interesting mini-PC device is Samsung's SPH-P9000, which also supports WiMAX connections.

Input Dilemma

A good input system is an important feature when large amounts of text is produced. From ViSCoS Mobile point of view, this is particularly important on programming courses. One solution to improve cumbersome input systems of mobile devices is to use a dictation software that is able to recognise and parse speech into text. If this is solution is not applicable, it is also possible to set the moblog as audioblog where the user posts short audio messages instead of text. Another solution is to use an external keyboard which is available for several smart phones and PDA devices. According to a research on mobile device data input systems [12], an external keyboard is the preferred input method over stylus and onboard keyboard. The third option is to utilise accessibility applications such as Dasher software [74] for input method that does not require typing. To enhance this solution, onboard motion or tilt sensor can be utilised in Dasher in order to allow the user

to input characters by moving mobile device on the air with his/her hand. Motion and acceleration sensors could be also used to enable three dimensional controlling mechanism as in Nintendo Wii game console. Futuristic input devices such as AlphaGrip [77] could be useful for typing text rapidly and efficiently.

Hardware Resource Issues

Processor speed and size of RAM in current mobile devices is enough for small applications if they are executed one at a time. Although multitasking is enabled in many smart phones (e.g. Nokia Communicator 9500), running several programs at once causes severe shortage of internal memory. Furthermore, processor speed shortage greatly affects the ability of a mobile device to fully utilise available network bandwidth.

To avoid the resource problems of mobile devices it is advised that software used for moblogging (be it e-mail client, web browser or custom tool), or for any other mobile activity, is executed independently without any other concurrent applications. Furthermore, mobile software should be chosen/built so that it does not use computationally expensive operations such as heavy encryption/decryption or compression/decompression. To enhance data transferring speed, image, video and audio files can be reprocessed to decrease their binary size.

One could claim that duration of battery life is a problem with Internet-enabled mobile devices as the wireless connection quickly consumes the power charged in the battery pack. However, if the blogging activity in a mobile environment is examined closely, one can easily see that the nature of moblogging is not continuous. Moblogging is bound to occur as often as any other activity performed on a mobile device; that is, when the user has free time and motivation for it.

Network Problems

Problems in mobile networks vary from connection problems caused by dead spots (i.e. device is out of reach of network antenna) to slow transmission speeds. The latter problem will be solved in the future when faster and more reliable wireless network technologies, such as WiMAX, xMax, 3G and 4G, are widely established. The problem with the dead spots can be solved by using hybrid connectivity, i.e. the device uses the fastest and the cheapest connection that is currently available and is able to change from one connection type to another without disconnecting.

Cost of Connection

Currently the price of a megabyte in mobile networks is very high, even higher than ten years ago when modems where the most common way to connect to the Internet. In Finland the situation is quite good and one can even obtain a subscription with a fairly low monthly fee which allow unlimited mobile data transfer. This is not the case in many other countries. While waiting for cheaper megabyte prices, the high cost of mobile data must be overcome by other means.

One solution to this particular problem is the previously mentioned hybrid connectivity in which a mobile device can hop from one network to another without disconnecting. For example, Nokia and Fujitsu have released a solution that enables mobile devices to switch from WLAN and GSM network seamlessly, thus making use of cheaper telephone connection when available. More network switching solutions are bound to appear as wireless technologies develop and demand for hybrid connectivity increases.

As mentioned before, moblogging is not a continuous activity, thus it does not use the mobile network constantly. Furthermore, if users sends images and text only to his blog, the amount of the transmitted data per entry will probably be less than 200 kilobytes. The amount of required bytes grows drastically, if video clips are added to the post. Therefore, if mobile data is expensive, one should avoid sending video clips and use text with small images only.

3.3 Implementing ViSCoS Mobile

ViSCoS Mobile as a concept was first introduced by Laine et al. [41] in 2005. The goal of ViSCoS Mobile is to develop necessary mobile learning environment and learning tools which can be used by ViSCoS students to learn via mobile devices, thus enriching the learning experience. Mobile technology is a double-edged sword. On one hand it offers high portability, instant access, flexible solutions and affordable price. On the other hand, however, small screen and lack of proper input method makes usability of mobile devices very bad compared to desktop or laptop computers. Unstable data storage and security of personal information are also issues affecting the design of the ViSCoS Mobile [65]. Contents of the ViSCoS courses are not optimised for mobile phones and even though the HTML content is theoretically readable by any web browser, presentation of large HTML tables on a small screen is very frustrating to browse.

To overcome the challenges of Mobile ViSCoS, three development threads have been defined that must be activated in order to enable implementation of the concept. These

threads are *content adaptation*, *programming on mobile device*, and *student support services*. This section briefly describes these three development threads which are aimed to overcome the challenges.

3.3.1 Content Adaptation

Online material in ViSCoS courses is currently implemented with HTML and CSS techniques, and some Flash animation is also included. To enable these materials on mobile devices is one of the main challenges in the ViSCoS Mobile concept. Majority of mobile devices have XHTML browsers capable of showing most common HTML structures, but particularly large table structures can cause problems. According to Jones et al. [35] reading performance may not decrease severely with a small screen, if users simply read or browse a piece of text. Simple navigation procedures, such as scrolling and clicking links, are not troublesome provided that the users are familiar with the control mechanism of the device. Complex interfaces designed to be viewed on desktop computers can be difficult or even impossible to access with mobile devices. Content adaptation techniques are used to alter content of the page according to properties and constraints of the devices, network and other factors.

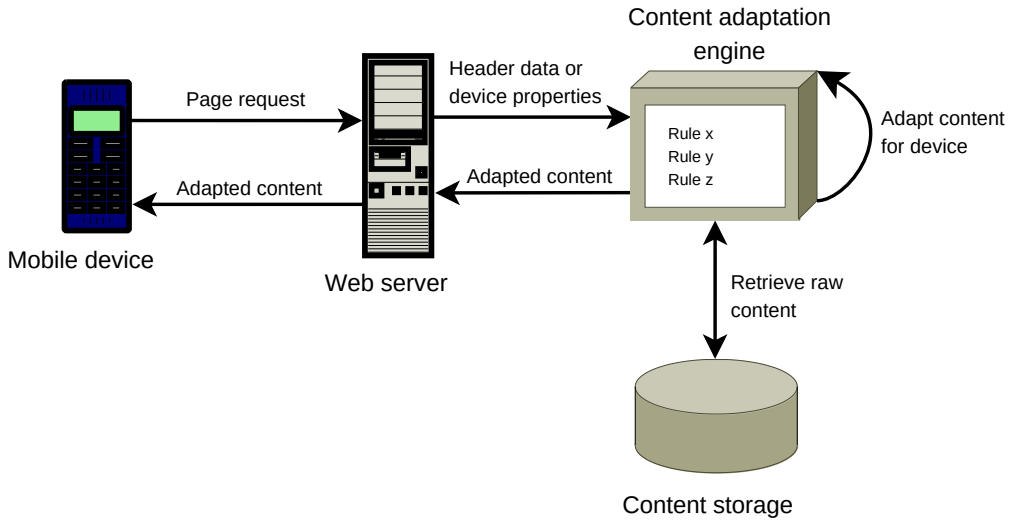

Figure 3.1: Generic idea of content adaptation

Three content adaptation techniques have been identified for potential usage in the ViSCoS Mobile. These techniques are *re-authoring*, *transcoding* and *structure-aware methods* [36]. Figure 3.1 presents the basic idea of content adaptation where the client sends

a request to a web server and an adaptation module generates and returns the appropriate version of the content to the client. It is possible to use these content adaptation techniques to adapt the ViSCoS material a suitable presentation for mobile devices. It is clear, however, that none of these techniques solve all adaptation problems, but their joint use provides the best result. Table 3.5 summarises the advantages and the problems related to the three adaptation techniques, originally reported by Laine et al. [41]. The last column includes possible application areas in the ViSCoS Mobile.

Table 3.5: Overview of Content Adaptation Techniques for ViSCoS Mobile (Laine et al. [41])

Technique	Favorable aspects	Problems	Most applicable in ViSCoS Mobile
Re-authoring	+ easy to implement + content is limited, no need for "on the fly" adaptation + learning materials in ViSCoS have similar structure, and they have already been designed for low bandwidth connections. Easy to implement different versions of the content according to predefined rules.	- non generic - requires different version of the course materials; requires much work	+ adaptation of the digital learning materials such as course theory
Transcoding	+ adaptation of dynamic content + management of adaptation rules + XML-based solution	- poor quality with complex structures such as nested tables	+ adaptation of the digital learning tools used in ViSCoS
Context-aware method	+ re-usable to other situations	- most laborious and difficult to implement - too heavy for needs in ViSCoS	+ adaptation of arbitrary Internet content (i.e. outbound links)

3.3.2 Programming on a Mobile Device

Programming is one of the main topics in the ViSCoS Curriculum and it should therefore be well supported in ViSCoS Mobile. However, programming on a small device with limited input capabilities may seem to be a very difficult and time-consuming challenge even for an experienced programmer. Furthermore there are no suitable tools available to support Java programming on multiple mobile platforms, such as Symbian OS or Windows Mobile. Table 3.6 presents a list of existing programming tools, environments and methods that were discovered by Laine et al. [41]. The table presents name, type, platform, and some pros and cons of each solution.

Table 3.6: Programming solutions for mobile devices

Name	Type	Platform	Pros/Cons
jCompile	Java compiler	P800 smart phone / Psion	+ free + Java - only Personal Java supported - no product support - very limited platform support
Manual installation of Java compiler	Java compiler	Any device supporting Java Personal Edition. Possibly others too.	+ free + java + possibly wide platform range - old Java (1.1.8) - no ready implementation - requires additional tools such as editor
BeanShell	Java source interpreter	Any platform with JVM	+ free + wide platform range - differs from Java syntax - execution on mobile devices uncertain
OnBoard C	C compiler	Palm OS	+ free + includes simple editor - not Java - highly platform dependent
Mobile BASIC MIDlet	BASIC programming environment	Any platform that supports Java MIDlets	+ wide range of platforms - not Java - costly (£19.99)
Python for Series 60	Python interpreter	Any Series 60 phone	+ wide range of platforms + free - not Java - does not contain editor
PythonCE	Python interpreter	Windows CE	+ free - quite limited range of platforms - not Java - documentation very minimal - under development

In fall 2005 two students of the Department of Computer Science at University of Joensuu were given a study project to investigate possibilities of compiling Java source code on

Nokia Communicator 9500 (Symbian OS, Series 80). The students, Hashim and Wafula, were directed to use Java Personal Edition (version 1.1.8) compiler for the project, as it was previously discovered to be a good choice [41]. The result of the project was a simple programmer's editor which utilised the imported Java compiler to compile Java files on mobile device. Hashim has continued the development of the product in his master's thesis and the results should be available in the near future. We believe his work is essential for practising programming on a mobile device in ViSCoS Mobile.

3.3.3 Student Support Services

One of the main challenges to support services in ViSCoS Mobile is to motivate students to study from a tiny screen. Using newest technology can be seen as a motivator at least for the technologically oriented students. However, it is clear that additional services are required to boost the motivation and to create and support a new kind of learning community; a mobile learning community.

Lack of community and interaction is a major element that contributes to many failures of online students [50]. In order to tackle this problem, mobile blogs could be used in ViSCoS Mobile to facilitate students' learning processes and to enrich collaboration between participants. In addition, mobile blogs would strengthen the ViSCoS community and release students from the restraints of location while reflecting the learnt knowledge. In this thesis, the emphasis is placed on the student support services of ViSCoS Mobile, and particularly on how to increase the interaction inside the ViSCoS community with help of mobile blogs.

According to several existing research results (e.g. [58], [16]) blogs can be very useful for education. They can be effectively used for example in organizing and processing learnt knowledge and new ideas. Blogs can in fact be considered as online learning diaries [42]. Not only the freedom of self-expression is granted to the author, but also constructive comments and brainstorming of ideas are a natural part of a collaborative blog system. The idea of using mobile blogs in ViSCoS is that the students could use them as a tool to facilitate their learning processes in a location- and time-agnostic manner. To take this one step further, students could carry a mobile device equipped with a camera. Whenever they meet a situation related to contents of the course, they could take a picture and send it to their mobile blogs with accompanying message. This effect of being ready and motivated to learn anywhere and anytime is a valuable factor for progressive learning. It also expands learning experiences from in front of a desktop computer into new surroundings.

As we see it, blogging would not be a student's privilege, but it could and should be used by the teachers and instructors as well. In this way, students could read their teachers' thoughts and reflect it to their own learning. Conversely, students could express their

ideas and give feedback on the course. This would also enable teachers to develop the course based on the feedback. Mobile blogs also provide a valuable tool for teachers to evaluate the learning process of the students, thus reducing the need for comprehensive examinations.

Woven Stories [25] is a collaborative writing tool which could be used in conjunction with the moblogging system to construct and visualise relationships between moblog entries. This way members of the moblogging community could actually see how their community looks like, thus giving them more motivation to develop the community further by writing. The teacher could utilise the moblogging community map to distinguish patterns of collaboration in order to use the knowledge for evaluation, for example. Woven Stories could be used also as a tool in courses requiring collaborative effort. For example it is an ideal tool for the Introduction to the Ethics of Computing course to support ethical debates and discourses. Furthermore, Woven Stories has been integrated with Jeliot, the program visualisation software, and the result is called JeCo - Jeliot Collaboratively [49]. The idea of JeCo is that programs can be developed collaboratively in Woven Stories and then reviewed in Jeliot. In order to use Woven Stories and Jeliot in mobile context the software must be ported to respective platforms. This is a challenge that yet requires many implementation details to be solved.

Additionally, ViSCoS Mobile could provide a set of additional student services including reminders, calendar and other information helping everyday life of an online student. Space- and time-agnostic queries, quizzes and even examinations would be possible to be organised with current mobile technology. User authentication needed for examination could be based on voice recognition, for example. Internet telephony and conversation tools on mobile devices would offer instant interaction between members of the learning community, thus creating a stronger atmosphere of togetherness.

Chapter 4

Blogs and Mobile Blogs in Education

Blogs are part of social software (e.g. [3], [70]). Others include Wikis, instant messengers, forums and social network services. Social software users interact with each other either in peer-to-peer fashion or as part of a larger online community. The idea behind using social software in education is to boost collaboration and interaction between the students. This is particularly important in distance education where interaction mostly occurs between instructors and students only. This chapter discusses existing research and solutions on blogs and their subset, moblogs. Since moblogs are essentially similar to blogs, the research results on blogs also apply in most moblogs cases.

4.1 About Blogs

The first private homepages started to appear when the Internet started its conquer over the digital world in the early 90's. Many of these pages were static by content and they were updated arbitrarily. There were, however, some pages which resembled much of a traditional private diary — only that it was not a private one, nor did it have a physical form. These web diaries were updated frequently and some even had a commentary feature. Some years after the first web diaries were published, the term *weblog* was introduced. Later this term shrank to *blog*. A random sample of a blog is presented in figure 4.1. This particular blog is established on Google's Blogger service and author's intention is to write three things daily, that have given her pleasure.

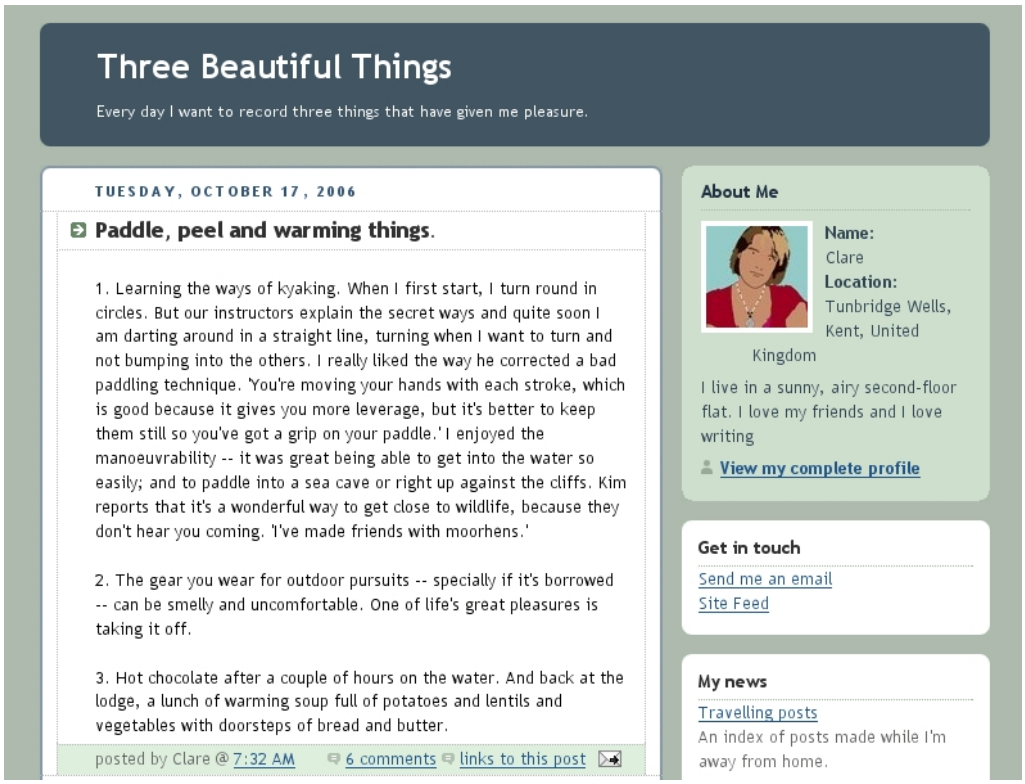

Figure 4.1: Screenshot of a random blog

Krishnamurthy [40] has proposed a classification of blogs into four basic types by using two dimension axes. These axes are: personal vs. topical, and individual vs. community. This classification is reproduced in figure 4.2 which marks dimensional areas as different categories. The first category, online diaries, is a typical blog hosted by several blogging services such as Blogger. In online diaries, users generally write about their lives, ideas and experiences. The second category, support group, is a collaborative effort of a group of friends who write about personal matters togehter. The third category includes individually authored columns, link filters and aggregators, which often have more visibility due to greater external interest compared to that of personal blogs. The fourth category is much similar to the third category by content. The difference between these two categories is that the blogs in the fourth category are created and maintained collaboratively. An example of such collaborative column is a corporate blog where employees write about their ongoing projects and events. Moblogs and blogs in ViSCoS programme belong to the third and fourth categories in this model; both individual learning blogs and collaborative team blogs could be useful for ViSCoS.

Blog consists of typically short *posts* (often referred as *entries*) which can be arranged e.g. by topic, date or any other categorisation criteria set by the author. A common way to

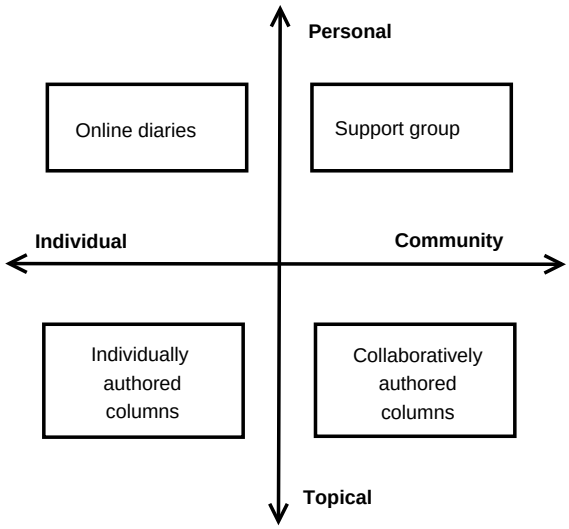

Figure 4.2: Classification of blogs by Krishnamurthy

organise blog entries is to do it chronologically, thus resemblance to old-fashioned diary is obvious. In chronologically ordered blogs, newest posts are generally placed at the top of the page. Many blog systems automatically create *permalinks* for each blog entry. Permalinks are permanent URL links through which individual blog entries are accessible even after they have been archived (i.e. not visible on the blog's main page). Some blog systems also allow user to attach *tags* (i.e. keywords) to each entry, thus making them easier to be found. Blog posts can consists of anything from a poem to an account of actions of an ex-boyfriend. The language used in writing the post depends on the author. English language, however, captures the largest potential audience as it is the common language used on the Internet.

From a technological point of view, a blog is a collection of hypermedia pages consisting of text, colours, links, images and possibly video or other multimedia material. However, blogs incorporate some features that distinguish them from ordinary hypermedia pages, such as a person's homepage. First of all, blog content is updated often, which makes it interesting for readers to follow. Secondly, most blogging systems enable interaction between the author and readers by use of a commentary feature, thus blog can be also seen as a discussion forum. This feature also links blogs together as blog owners tend to be readers (and therefore commentators) of other blogs. As Downes puts it: "writing a weblog appears in the first instance to be a form of publishing, but as time goes by, blogging resembles more and more a conversation" [16]. Lastly, blogs are often equipped with a *blogroll*, i.e. a collection of links to other blogs which interests the author.

Blogosphere is a term used to describe a set of blogs connected together to form a social network or a community. Generally this term is used for all blogs on the Internet, but

we used it here to also describe smaller blog communities. The connections between blogs are established by users commenting, reading and linking each others blogs. This interaction can be seen as a discussion forum within the blog system. Social network consists of users who may know each other in real life, or they can be merely virtual acquaintances. Some common theme might tie up the community together, but it is often just a network of friendships. In an educational setting a blogosphere could consists of students of a single university or a faculty. A blogosphere can be refined further to smaller core units (e.g. students of a single class) where dense discussion takes place. Publishing techniques such as *RSS* and *ATOM* have been developed to facilitate observation of blog entries from different sources, thus making it possible to keep track on several blogs with ease.

4.1.1 Mobilising Blogs

Recent development in mobile communication devices has established a new wave of blogs — mobile blogs, or moblogs — whereby authors are able to update their blogs wirelessly by the means of a mobile device. Moblogs resemble normal blogs, the major difference is the posting method. The advantages of using mobile blogs instead of ordinary blogs are unrestricted location and time. Due to technical limitations of mobile devices, entries sent from them are usually shorter than those sent from a desktop computer (i.e. ordinary blogging). Furthermore, moblogs are often used jointly with normal blogs, thus moblogs can be seen as an extension to ordinary blogs. Figure 4.3 presents a screenshot of a random moblog. The author of this particular moblog uses his mobile device's camera feature to post images; texts posted with images are often very short.

4.1.2 Challenges and Solutions

As any other technology, blogs and moblogs have some challenges to be solved. The first problem is that bloggers often write to a specific group of readers (friends, colleagues, fans), thus they filter out the inappropriate text for the audience. This is a potential threat as it may lead to self-censorship. For example a student might think: "I don't dare to post this idea because someone can consider it ridiculous". To overcome this problem, the author can post anonymously to a specific anonymous blog in which sensitive issues are discussed. Furthermore, blog entries can be protected so that only members of a specific group (e.g. ViSCoS community) are able to access the content.

The second problem concerns the commentary feature of a blog. The comment spam is a phenomenon much like e-mail spam; spam programs find blogs from the Internet and post advertisements as comments to them. There are several solutions established

Figure 4.3: Screenshot of a random moblog

Figure 4.4: Example of Captcha verification on Nucleus CMS

to overcome this problem. The simplest solution is to disable the commentary feature or limit it to blog system users only. Another solution is that the blogs author manually accepts and rejects comments; once comment has been accepted, further comments from that particular reader are accepted automatically. Third solution to fight comment spam is to use the so called *captcha* (Completely Automated Public Turing test to tell Computers and Humans Apart) images. Captcha image presents a graphical (sometimes distorted) representation of text that is very difficult to be read programmatically, but can be deciphered by most human eyes. Figure 4.4 presents an example of a captcha image used by Captcha plugin of Nucleus CMS; the user must rewrite the text correctly in the field below the captcha image in order to post a comment.

Publicity of blog is usually beneficial; the more readers the blog has, the more comments the author receives and the more active the blog is. However, in some cases publicity can be seen as a threat. For example a user who uses his own name in a blog might be queried about his writings in a job interview in the future. Search engines are powerful tools and they can reveal much of a person's identity. Therefore, one should carefully weigh the advantages and disadvantages of publishing ones writings to the entire Internet. Anonymity and restricted access can be used as means to tackle the publicity problem.

4.2 Motivation to Blogging

Motivation to write a blog varies from person to person, but some common characteristics can be distinguished. Here we put emphasis on motivation of blogging in education because of its relevance for this thesis. The discovered factors could be used to establish an interactive blog community in the ViSCoS programme. This could in turn help to overcome lack of the students' motivation.

According to research done by Wang et al. [73] people write blogs in order to share lives with their families and friends, to publish their personal profile online, to share emotional feelings with other bloggers, to influence readers by self-expression, to log events during travelling, and finally to develop new life experiences by new forms of social interaction. It is clear that a single blogger is usually motivated only by a subset of these motivators. Another study [19] suggests that successful blogging is enabled with a context that requires creation, processing, organising, storing and publishing ideas. From a technological point of view, access to the Internet as well as tools supporting blogging and discourse are necessary. Successful blogging also requires correct attitude, open-mindedness and desire to write a blog.

Additional motivators found by Nardi et al. [51] were; to receive feedback and opinions for own ideas, and to "think by writing", meaning that people are brainstorming and processing new ideas during authoring process. These days blogging can be seen as a trend among youngsters and young adults. To be part of this trend is also one motivator that drives people to start blogging. However, blogging should not be enforced, but it should be a voluntary act. Richardson [63], having three years of experience in using blogs in classrooms, recognises the same concern by writing: "By its very nature, assigned blogging in schools cannot be blogging. It's contrived." And he continues: "...my students drop blogging like wet cement when the class is over". In order to avoid this problem, students should be attracted to write their blogs by means other than declaring blogging as a mandatory part of the course. One idea is to have blogging as an alternative way of evaluation; students could choose between writing a blog or doing exercises.

Krause concerns with Richardson by writing in his article [39] about his bad experiences using public blogs in classroom education. According to his experiences, students will not write just because they are given an opportunity to do so. Krause argues that "the writer has to have a reason– and generally, a personal reason", thus forcing students to write without internal motivation is not appropriate. It is therefore necessary to motivate students to be interested in the subject at hand so that they start to discuss the topic on their own and write not only for themselves, or for the teacher, but for everyone; their friends, families and random surfers all over the Internet. The reason for lack of writing can be also because of insufficient feedback from the audience in form of commentary. Another study [51] revealed that commenting can be infrequent or even non-existent especially if

the blog's reader base is small. On the other hand the same research discovered that many bloggers do not wish to have comments; blog is their "refuge from the intense interaction of other forms of communication". While this might be true with personal blogs, the case is different in educational blogs where students ought to have active discourse.

What could trigger and maintain the inspiration and genuine desire for blogging? Many potential bloggers probably do not know much about the art of blogging so they should first be guided to the topic. Another motivator are friends who have already adopted blogging; blogging is a social activity after all. As the first impression is very important, a first time blogger should be provided with easy-to-use tools in order to avoid additional stress that could cause abandonment of blogging. Once blogging has started, it should be encouraged with active commentary between bloggers. Students could be for example provided with extra points on the course if they actively compose meaningful comments to other students' blogs and answer to other comments on their blogs. The blog's author could also rate comments according to their relevancy and usefulness on a simple scale.

4.3 Characteristics of Bloggers

Bloggers can be roughly divided into individuals and groups. Individual bloggers maintain their personal blogs while group bloggers contribute to a common blog, such as work blog, research blog or some other blog of common interest. Individual bloggers can be further divided according to the contents of their blogs or their motivation.

Today anyone can become a blogger without knowing anything about web programming, thanks to free blogging services offered by several companies. National Institute for Technology and Liberal Education (NITLE) is a non-profit consortium which maintains the Weblog Cencus [54] service which keeps count of the number of active weblogs in different languages. Table 4.1 lists the number of weblogs in different languages. The last row of the table shows some exceptions and overall figures. Although these numbers do not show the exact amount of blogs on the Internet, it can be used to understand the distribution of blogs between languages. One must keep in mind, however, that the proportion of blogs written in English includes the blogs written by non-native speakers. Another source, Technorati, claims to have tracked 35.3 Million weblogs by April 2006 [69]. Since the difference between Technorati's and NITLE's figures is so huge, the real amount of active blogs is probably somewhere between them.

Table 4.1: NITLE Weblog Cencus Statistics (October 24, 2006)

Language	**Count**
English	1958443
Catalan	123320
French	83950
Spanish	80509
Portuguese	71561
German	35870
Italian	26659
Chinese-big5	25123
Farsi	19730
Chinese-gb2312	19324
Japanese	18576
Dutch	13133
Danish	9870
Indonesian	8831
Malay	6658
Japanese-euc_jp	5413
Swedish	5267
Czech	5089
Icelandic	3776
Tagalog	3608
Finnish	3326
Turkish	2817
Esperanto	2803
Slovak-ascii	2592
Too_short (not analysed reliably)	284749
Total amount of blogs	2869632

Politicians are one group of users who have adopted blogging as tool to gain more visibility and to become closer with their voters, both existing and potential ones. Particularly young audience can be reached better via blogs than through more traditional media such as newspapers. For example the prime minister of Finland, Matti Vanhanen, writes in his blog [72] not only about his daily work, but also about his holidays. By including his private life in his writings, Vanhanen gives a more intimate feeling of himself to the readers. In addition to Finland, blogs in Spain and Portugal [60], United States [17], [1] and United Kingdom [66] have been used in politics too.

Newspaper and TV-reporters have also realised the usefulness of blogs in their work. Blogs offer them a means to express their own experiences and thoughts of the incidents they encounter. Blogs have proved to be a powerful tool for reporting news from war zones, catastrophe sites and other locations that are difficult to be reached by the means of ordinary media. For example, after the September 11 terrorist attack in New York, many web sites of the news media were down while several blogs provided valuable up-to-date information of the latest events on site [40]. Reporter blogs are not only written by individual reporters, but also a group of reporters who are working for the same news agency. For example, BBC News has a concept called *Reporters' Log* in which several BBC's reporters write short reports from different locations, such as war zone in Iraq and from London after the terror attack against the London public transport system on July 7, 2005. These are the stories are not usually published in the news, or at least they are published online before they reach the news.

Blogging is also used in the corporate world to empower decision making, to circulate new ideas, to enhance communication, to deliver information and to improve collaboration. Tacit knowledge can be captured and redistributed with the help of the blogging. For example new employees can refer to project's blog archives in order to understand the current situation and the organisation better [70]. Corporates can also take advantage of blogging technology as a form of *distributed apprenticeship* [20], aka *legitimised theft*, where employees of the organisation learn from each other via blog entries. Lastly, companies can use blogs for public relations in the same way as politicians use blogs to get closer to their voters.

A good example of a corporate blog is Google's Official Google Blog in which employees of Google report product and technology news. Google hosts also a large number of other blogs dedicated for different activities inside the company. Another example of using blogs in the business world is Oracle's blogging community in which "Oracle executives, employees, and non-employees alike exchange views about customer requirements and best practices for using Oracle and industry-standard technologies". More corporate blogs and blog communities can be viewed through link lists collected by the TheNewPRWiki web site [53]. Majority of these blogs are established by information technology corporations, but there are also some exceptions such as General Motors' corporate blogs and Guinness blog.

4.4 Blogs and Education

Majority of the blogs are written by individuals or communities for their own purposes. However, there exists some efforts to incorporate blogs in education. The term *edublog* is often used to refer to a blog used for educational purposes. It is not a surprise that

many of the forums for discussion in edublogs are in fact built upon blogs. This section provides references to research results on blogs in education. The material presented here is not meant to be comprehensive; it merely attempts to show that blogs have already been adapted successfully as part of education for some time.

Du and Wagner [18] have completed a study on using blogs as learning logs. Their results suggests that blog usage during the course can be used to predict student's overall success; if a student writes long entries regularly into a blog, she/he is likely to perform better in the course. This knowledge could be used in ViSCoS courses where one of the biggest problems is a high number of drop outs. The drop out detection has already been researched by means of data mining (e.g. [30], [31]), but there has not been a complete breakthrough yet. Some might argue, however, that the blogging activity distracts students from the actual learning. It is therefore imperative that educational practise of blogging is clearly defined before introducing blogs to students.

The problem of motivation in distance education can be tackled partly with the help of blogs. Williams and Jacobs [76] noticed in that higher education students are particularly positive towards using blogs as a tool in their studies. According to the results, a great majority of students preferred to continue using blogs as an effective aid to teaching and learning.

Farrell [22] has distinguished several different ways to use blogs in education. Table 4.2 presents Farrell's findings together with examples; the first column describes the application of blog while the second and third column present the roles of teacher and student, respectively, on the particular scenario. The last row of the table was not included in Farrell's article. Some of these usage scenarios can be used simultaneously; for example course homepage and linkdump can co-exist in one blog.

Table 4.2: Uses for blogs in education (Farrell [22])

Application	Teacher's role	Student's role
Standard web page	Teacher maintains the course web page as a blog; it is easier to update than ordinary web page. Blog can for example include schedules, notifications, rules and links to resources.	Students read the blog regularly in order to keep updated.
Linkdump	Teacher adds interesting links to blog.	Student visits the links in order to receive deeper understanding of the matter at hand. In addition to studying the linked resources, student can send comments on them.
Discussion and debate	Teacher harnesses a blog to foster in-class discourse between the students.	Student uses the blog to post new discussion topics and/or comment on existing topics.
Seminar publications	Teacher uses blog to organise intensive seminars	Student posts weekly summaries of course readings and have active discussion about the topics.
Student's blog	Teacher merely observe students' own blogs and uses them as part of course evaluation.	Student writes her own blog during the course. Contents of the blog can vary, but should be connected to the subject of the course.
Administrative blog	Teachers are writing blogs in order to enrich the communication between themselves and institutions.	Students do not participate.

Mortensen and Walker [48] argue that using weblogs as a tool in tertiary education helps students to develop and sustain a confident and clear voice of their own, and the ability to formulate and stand by opinions. These indeed are vital skills of scientific work across different fields. Mortensen and Walker continue by stating that when one is writing a weblog, he is forced to confront his own writing and opinions, and to see them reflected in the words of others [48]. This is particularly true if the blogging community is active. Downes [16] witnessed the power of blogs as a supportive tool in education at Institut St-Joseph, Quebec City. According to his findings both teachers and students felt the benefits of blogs, and also that using blogs encouraged students to write more. Another paper [58] notes that blogging has the potential to be a transformational technology for teaching and learning in tertiary education. Huffacker has noticed the usefulness of blogs to promote literacy in the classroom [33]. Placing et al. [58] mentions the i-lecture project at the University of Western Australia, in which learning material, including text, audio and

video, is delivered to students' mobile devices. After studying the material students reflect the learnt theory in their blogs and comment on the writings of other students. These examples prove that blogs have already been successfully established as an effective tool in education.

Blogs are also part of a distance learning setting in Dickey's work where blogs' impact on isolation and alienation of students are investigated [14]. Dickey concluded that students' experiences with blogging in a distance learning setting were positive and use of blogs reduced their feeling of isolation. Students also preferred blogs over traditional discussion forums because they had a chance to manage their own blogs instead of posting to teacher-moderated forums. Blogs also offer more interaction in form of commentary. This result is suggests that blogs, and mobile blogs, could be used in ViSCoS programme to improve voluntary collaborative effort, which is currently minimal.

A few projects and blogs have been established to research and develop blog usage on the educational arena. Weblogg-ed [61] is a website discussing and reflecting usage of blogs and other related technologies in K-12 education. Weblogg-ed also contains a frequently updated blog with recent information of ongoing events on the arena of edublogging. Another project promoting educational blogging is HighEd BlogCon online conference which "seeks to engage the Higher Education community in a conversation on the use of blogs, wikis, RSS, audio and video podcasts, social networks, and other digital tools in a range of areas in academe" [8]. The third example of an effort to bring blogs into education is Educational Bloggers Network [52] which is a community of teachers and education professionals and supporters who are using weblogs for learning and teaching.

Majava [43] suggests that blogs are especially suitable for students writing their thesis, as well as for different research courses where students are working rather independently on a larger project. Here we would add courses where the education method is problem-based learning because students using this method particularly benefits from reflection of their learning processes. According to the experiences on a theoretical computer science course at the University of Joensuu, problem-based learning is an effective tool that fulfills the needs of active and collaborative learning [29].

4.5 Moblogging in Black and White

While usage of ordinary blogs have been under a vast number of research projects, mobile blogging, or moblogging, has not been a popular topic particularly in the education field. This section summarises the key findings of existing moblog research.

According to Goggin [26], mobile blogging is a rather new phenomenon in large scale and it is still finding its shape. The same paper states that the difficulty to embrace moblogging

as part of a wider set of blogging is because the lack of interest from mobile phone carriers and vendors. This means that in order to start moblogging, one must know how to set up a moblogging environment with necessary tools. Nokia Lifeblog [55] offers a solution to this problem, but it is suitable only for a few proprietary device models. A case study of mobile blogs [75] suggests that "individual mobile media artefacts derived from quotidian, transitory and insignificant events have only passing interest". Nature of moblogging is therefore very current and meaningfulness of the presented information decreases fast.

Several research results (e.g. [73], [2]) have revealed that the longer people keep their mobile blogs, the fewer photos they transmit. Reasons mentioned for this behaviour is due to incentives, bad image quality, lack of services and operational difficulties of the mobile device. Lack of good client-end software may also have an effect on this trend. In order to make moblogging with camera phone more appealing, camera feature should be equipped with a link to moblogging tool which would enable access to mobile blog with ease. Furthermore, the camera quality should be higher and the file size smaller in order to increase the attractiveness of this activity. Lastly, the cost of data transfer from mobile device to network should be much lower for consumers to be interested in mobile blogging. Today many camera-enabled mobile devices also support video capturing. This feature can be used for example in mobile videoblogs [27] where users send video clips with accompanying text to their blogs. In case of video moblogging, the cost of data transmission becomes an even more crucial factor that hinders adoption of the technology.

Mobile blogs have been used in education as part of several research projects. A recent study by Berth [6] introduced moblogging to high school students as a tool of reflection, resulting to the statement: "when students communicate through mobile media, social knowledge is produced which in turn require reflectiveness to be exercised". Berth also noticed that students prefer to send images to moblog than share them by MMS technology. Russell [5] presented a SmartBlog tool for mobile blogging and discovered that it supports the immediate, ad-hoc nature of mobile learning experiences more effectively. Experiences of another project, RAMBLE (Remote Authoring of Mobile Blogs for Learning Environments), have shown that mobile blogs are very useful to support personal reflections in an educational setting [71]. In addition to these examples, mobile blogs have also been used in language education [23].

As new features are added to mobile devices, they become more versatile in terms of usage scenarios. Pictures, audio and video are already conventional media for blogging. From newer technology, an integrated GPS navigator can be used to add geographical information to moblog entries [27]. RFID reader embedded on a mobile device has already been used to capture context-sensitive information of places and everyday objects in order to pass this information into mobile blogs [9]. This kind of information can for example help people with common interest to find each other, thus establishing relationships in a

virtual community.

What we do not know about mobile blogs is how well they suit in computer science distance education. There are some research done in usage of moblogs in distance learning, e.g. [11] and [10]. None of these, however, concern the field of computer science. Our literature review did not provide results concerning research on moblog usage in computer science distance education. However other research results presented in this chapter suggests that mobile blogging could work in this field as well. After all, computer science students, being particularly interested in new gadgets, are likely to adapt new technologies rapidly.

Chapter 5

Implementing a Solution for Moblogging - Advanced Postman

This chapter starts by presenting existing solutions for blogging and moblogging. Then we select the best method to establish a moblog system to be used in ViSCoS, and finally implement a moblog system with the chosen method. As it will be soon revealed, the chosen method was to build a moblogging system based on an existing open source blogging system. Therefore we analyse different solutions to implement a *moblog gateway* that delivers messages from a mobile device to a blog. Based on the analysis, the best entry delivery method is chosen as the model of new moblog plugin which is then implemented based on existing moblog plugin.

5.1 Analysis of Existing Blog Solutions

This section describes the basic structure of a blog system and presents different solutions to establish one. At their simplest, moblog systems are merely extensions of existing blog systems. The information presented here applies generally to any blog system, including moblog systems.

5.1.1 Blog System Architecture

Most blogging systems are based on *client-server architecture* where the server part is running on a web-server, such as Apache, which offers a standard HTTP access to the blog system via hypermedia interface. The reader of the blog uses a web browser or some

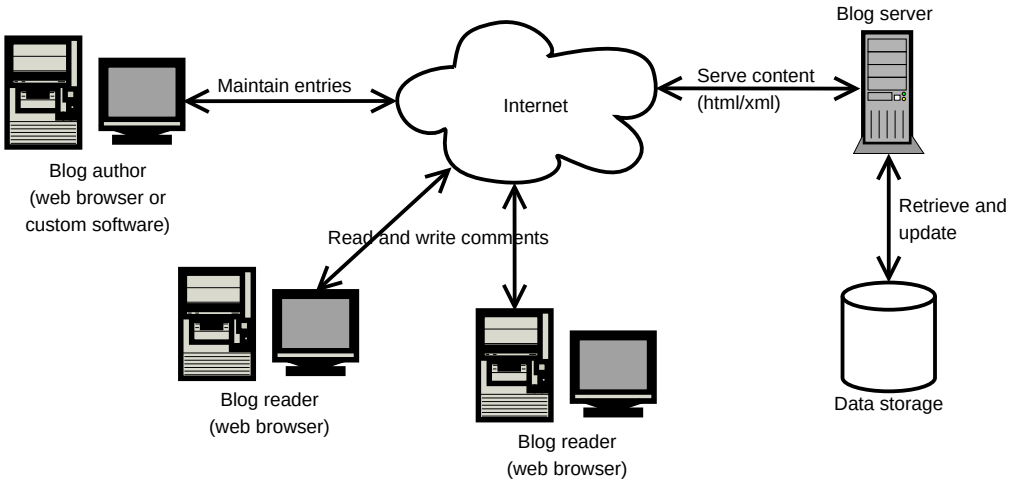

Figure 5.1: Client-server blog system architecture

other appropriate client to access the content on the server. In such a model, the server part is the core of the blogging system and clients merely connect to the server in order to update, read or maintain contents of the blogs. Figure 5.1 presents the architecture of a typical client-server blog system.

Data of the blog system is saved in a database. In the case of an open source blog systems, MySQL database is often used in conjunction with PHP server-side scripting language. PHP scripts manage saving, updating and removing blog entries, as well as maintaining user accounts and other blog properties. When a user connects to the blog system by web browser, the PHP interpreter on the server executes the requested script and produces the appropriate HTML-page to be sent back to the client. Some blogging systems also offer a custom tool for maintaining the blog.

User authentication in client-server blogging system is often accomplished by the user accounts saved in the database of the blog system. Each user has a unique username and a password set for it. The password is stored in an encrypted format on the database in order to reduce the risk of identity theft. Powerful blog systems are able to run several blogs concurrently and it is not a rare phenomenon to witness several authors writing the same blog.

Many blog systems offer a searching feature which can be used by readers to search for past blog entries by filling in an arbitrary keyword. Searching is usually performed by a simple SQL query to the underlying database. An alternative way to perform a search is to search not only the local databases (i.e. local users), but also remote blog systems sharing the same characteristics. In this case, an open interface must be defined by the blog system to enable access from a remote host. Remote connection requires usage of

appropriate credentials.

RSS and ATOM techniques are often used to provide XML-based feeds of blog entries and comments. These feeds help to keep the blog readers updated of modifications on the blog in a uniform way despite the underlying structure of the blog system. The technical implementation of such features can be done for example by a PHP script which updates the respective XML feed files whenever modifications are performed on the blog in question. XML-based feeds can also be generated on-demand, but this is a less favourable option if the number of readers using the feeds is large.

Several blog systems allow administrators to expand the functionality by adding new plugin modules to the system. A talented administrator can also write custom plugins if the public plugin library does not offer any suitable solutions to the problem at hand. Plugins are usually attached to the system via *plugin API* (Application Programming Interface) which is a set of methods, structures and rules that must be followed in plugin development.

5.1.2 Establishing a Blog

There are two ways to set up a new blog: by using one of the various blog services or by manually installing and configuring a customisable blog system. This section will present, in a non-exhaustive manner, the most well-known and interesting blogging services and software available today, each having their advantages and disadvantages. It must be noted, however, that new services and software are introduced all the time and therefore in order to get a fresh look on the topic, some Internet searching is advised. All presented solutions are evaluated for their applicability to ViSCoS.

Service-Oriented Blog System

Creating a new blog by using a service-oriented blog system, i.e. *blog service*, is easy and does not usually require any programming skill. In such service, the blog is hosted on a server maintained by a service provider who sets the terms of usage and may limit some features to paying customers only. Additionally, blogs based on free accounts may bear permanently attached advertisement banners. Some services may offer possibility to enhance and modify blogs by means of scripting, but this is merely an additional feature and by no means necessary to use. Blogging services are ideal for individual users who do not want to spend time configuring the blog software on their possibly non-existent web server. For technically advanced users, groups (e.g. study groups) and closed communities service-oriented blog system is not a good option.

The most known free (or partially free) blog services include, but are not limited to, Google's Blogger, Six Apart's LiveJournal, MSN Spaces, Yahoo's 360°, MySpace, DiaryLand and Xanga. A common feature for all these services is that they support communities and social interaction in one form or another. Each of them has a huge international user base thus the main language is English. As blogging is trendy today, these blogging services are growing rapidly. For example, Rohan et al. has predicted that the market value of MySpace will be 10-20 billion USD in 3 years from 2006 [64]. The free use of these services is compensated by advertisement banners and videos on users' blogs and homepages.

Although using free blog services is trouble-free in terms of maintenance, they do not give means for instructors to moderate inappropriate contents. Observation of blogging is also difficult both from research and the evaluation point of view. Furthermore, the means of customisation are very limited and services are bound to change and even cease to exist in the course of time. Lack of support for Finnish language is also a problem as currently majority of students are attending the ViSCoS courses in Finnish.

Custom Blog Systems

There are various software packages available allowing bloggers to set up their own blog systems. The interesting fact here is that many of the customisable blog systems are open source and can therefore be modified at will. Some of these blog systems offer a plugin API which allows blog system administrator to easily add more features to the core system. Plugins can be either downloaded or even created from scratch, if person is talented enough. This section briefly presents three popular blog systems in regard to moblogging purposes in ViSCoS.

Nucleus CMS

Nucleus CMS (Content Management System) is an open source publishing environment for weblogs. What makes Nucleus CMS stand out is the ability to host multiple blogs written by multiple authors, allowing each author to contribute to several blogs. This is a necessary feature in an educational setting where each student might be writing a personal learning blog and students might additionally share a common class blog. Administration of the blogs, users and other resources can be done via an easy-to-use administration interface. The team publishing feature together with multiple blog support and easy administration makes Nucleus a strong candidate for educational setting. Thanks to its open source license, the product is free of cost.

There are a few prerequisites that must be met before Nucleus can be successfully installed. First of all, a web server with PHP scripting support needed as the system is written completely in PHP language. Secondly, a MySQL database is required as Nucleus

uses it to store all information such as blogs, authors and plugin data. After these requirements have been fulfilled, the actual installation process of Nucleus is very straightforward and no programming skills are required. If the user does not have account to a web server, he/she can buy such an account with a reasonable monthly cost.

Nucleus offers a powerful and easy-to-understand plugin API and a large library of existing plugins for those who are not comfortable with developing their own plugins. For example, one of the Nucleus plugins, namely PostMan, can be used to send entries from mobile devices via e-mail. However, it was discovered that PostMan is not suitable to be used in ViSCoS Mobile, thus another plugin, namely Advanced Postman, was developed. The technical details and motivation for the Advanced Postman plugin are discussed in Chapter 5.

Posting API defines a necessary interface to send a new entry to a blog system from an external application. Different blog systems use different posting APIs. In addition to its own posting API, Nucleus supports Blogger API, MetaWeblog API and MovableType API. By supporting different posting APIs, a blog system can be accessed from other blog systems as well as from custom blogging tools.

WordPress
WordPress is very similar to Nucleus CMS. It also is based on an open source license, thus it is free of charge. Like Nucleus CMS, WordPress requires a web server supporting PHP and MySQL. WordPress is also easy to install and does not require any programming skills. Currently WordPress does not support multiple blogs on single installation. There has been some effort to create a multiuser version of WordPress, but the development work is currently in progress. WordPress does not offer proprietary posting API, but it utilises several existing ones: Blogger, MetaWeblog and MovableType. Figure 5.2 presents a screenshot of a blog running on WordPress.

Expandability of WordPress has been implemented as plugin API. Due to popularity of this software, it has a larger plugin library than Nucleus CMS. For the same reason developer network is also larger, thus support for problem situations is easier to get. WordPress has a moblogging plugin called Postie. It works much the same way as the PostMan plugin for Nucleus CMS does, but since WordPress does not offer reliable support for multiuser blog environment, Postie was not evaluated in this thesis.

Movable Type
Movable Type is a commercial blog system from Six Apart Ltd. Like Nucleus CMS, it is capable of running several blogs for multiple users. Movable Type is licensed according to use of it; for personal, unsupported use, the license is free, but any other type of usage bears a price tag. For example, educational license of Movable Type is priced per classroom and student basis, starting from $39.95 USD (October 2006).

Figure 5.2: Screenshot of a blog running on WordPress

From a technological point of view, Movable Type utilises Perl server-side scripting language. The database can be established by using one of the following database engines: MySQL, Berkeley DB, PostgreSQL or SQLite. The basic architecture of Movable Type is therefore very similar to Nucleus and WordPress: a script fetches data from the database and renders it suitable for a web browser. Movable Type also offers a plugin API together with a large plugin library which can be used to extend the functionality of the system. There are at least two e-mail blogging plugins available for Movable Type which can be utilised for mobile blogging. In addition to its proprietary posting API, Movable Type supports Atom, Blogger and MetaWeblog APIs.

Comparison of Blog Solutions

Existing blogging services are too restricted for ViSCoS; they do not offer expandability nor can they be customised. Furthermore, incompleteness of multiuser support on WordPress, and MovableType price tag solidified the decision to use Nucleus CMS as the core bloggins system for moblogging on ViSCoS. Table 5.1 presents the overview of discovered blog solutions with their features. Movable Type features covers the version with the education license.

Table 5.1: Comparison of blog solutions

	Blog services	**Nucleus CMS**	**WordPress**	**Movable Type**
Price	Varies greatly	Free	Free	$39.95-$299.95
Support	Varies greatly. Paid accounts yield better support	Community support	Community support	No support mentioned in license details
Plugin API	No	Yes	Yes	Yes
Multiple blogs	No	Yes	No	Yes
Multiple users	No	Yes	Yes	Yes
Moblogs	Partly	Yes, by plugin	Yes, by plugin	Yes, by plugin
Extensibility	Cannot be extended	Open source - can be extended indefinitely	Open source - can be extended indefinitely	Cannot be extended
Server requirements	None	Web server with PHP and MySQL	Web server with PHP and MySQL	Web server, Perl and one of the following databases: MySQL, Berkeley DB, PostgreSQL, SQLite
Blog moderation	No	Yes	Yes	Yes

Continues on next page...

Table 5.1: Comparison of blog solutions (continued)

	Blog services	Nucleus CMS	WordPress	Movable Type
Posting APIs	Depends on the service	Nucleus, Blogger, MetaWeblog, MovableType	Blogger, MetaWeblog, MovableType API	MovableType, Atom, Blogger, MetaWeblog
Finnish language support	Mostly not supported	Yes	No	Yes
Applicability for ViSCoS	Not applicable, because blog moderation is essential in a virtual classroom environment. Lack of extensibility is also a disadvantage.	Very well applicable	Applicable as soon as multiple blogs feature is supported properly	Applicable, but pricing is a disadvantage

5.2 Moblog Gateway

In most cases moblogging functionality can be attached to an ordinary blogging environment software as a plugin that can handle posts received from a mobile device. This approach is good because moblogging is essentially very similar to blogging, thus it would be a waste of resources to copy the same code to create a separate moblogging software. The only thing that is needed for moblogging is a gateway functionality that can receive incoming moblog entries and direct them to the blogging engine, which in turn handles the entries as ordinary blog posts. There are several technical possibilities for implementing the gateway functionality. This section explains three discovered moblog gateway options in detail.

5.2.1 Post by E-mail

Today, almost every mobile device supports e-mail functionality, thus it is very convenient to utilise this feature in order to send an entry to a moblog. For this approach an e-mail account dedicated for moblogging must be created. The gateway for posting by e-mail consists of a mobile e-mail client and a script that polls a specific e-mail account every given interval of time. If new messages are found in the e-mail account inbox, the script either directs the message to a blogging engine in the appropriate format or uses public functions of the blogging engine to perform storing of a new entry. In both cases the e-mail must be parsed properly and transformed into a format understood by the blog

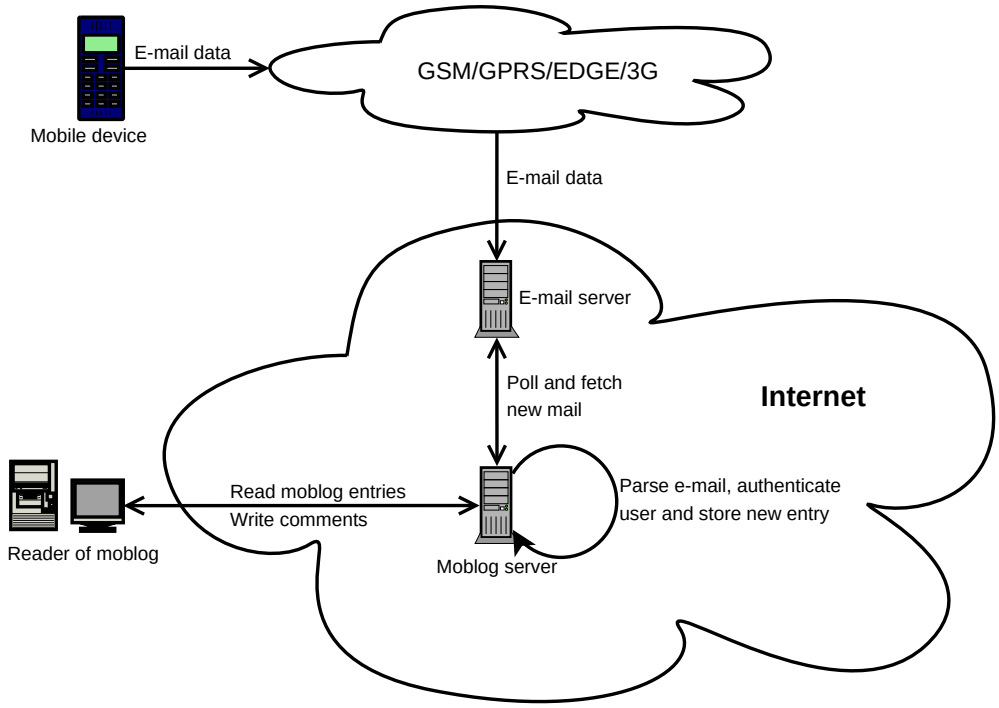

Figure 5.3: E-mail gateway

software. E-mail posting offers existing infrastructure and protocols for message delivery so development work can be concentrated on message parsing. Figure 5.3 presents the generic architecture of an e-mail gateway system.

In order to filter out inappropriate incoming e-mail messages, a scheme of author identification can be introduced. In case of a plain e-mail, this poses a problem since e-mail header information can be easily forged for misuse. However, if the moblog mailbox address is kept secret and changed every now and then, user identification can be done simply by detecting the user from the sender field of the e-mail header. This method is secure enough for most of the blogs. If a spam e-mail finds its way to the secret e-mail account, it can be easily removed by manual moderation performed by the moblog author. In addition to sender field verification, other parts of the e-mail header can be also used in message filtering.

More powerful user identification schemes utilise secret key phrases and digital signatures. In the key phrase method, a secret key phrase is checked by the moblog script. Needless to say, this key phrase must be explicitly defined in blog settings and it is a good idea to change the key phrase regularly. The longer and the more random the key phrase is, the more secure the identification becomes. Digital signature methods such as PGP are very secure and based on usage of public and private keys. Signing (by the

sender) and verification (by the recipient, that is, the moblog script) are fairly expensive operations in terms of processor time. Therefore one should carefully estimate whether the benefits of the digital signature exceeds the delay of message signing and verification. It should also be kept in mind that if the moblog is maintained by a group of individuals, then each of them should have their own respective signature and the moblog script should be able to validate all these signatures.

It is also possible to combine different user identification methods in order to obtain a more secure method. For example, verification of sender field can be used together with a secret key phrase check. This method is fast and quite powerful. The identification system should be flexible enough to allow the moblog owner to easily add and modify rules according to her/his needs.

Pictures, video and audio captured with mobile devices can be sent to a gateway as e-mail attachments. In order for the gateway to recognise types of attachments and parse them accordingly, *MIME types* (Multipurpose Internet Mail Extensions) are used. This way received images can be posted by using IMG-tags of HTML. Should the image be too large, thumbnail images can be automatically created with image processing tools such as ImageMagick. Due to MIME types, captured videos can be embedded properly, and other file types can be inserted as hyperlinks pointing at respective files. As for mobile phone video, the 3GPP and 3GPP2 formats are generally used. Both of these are supported by Apple QuickTime software, thus they can easily be embedded on the moblog and then opened with the QuickTime player. It is also possible to convert video recorded with a mobile device to a more conventional format such as AVI. For this purpose, an open source software *FFmpeg* can be utilised via a PHP script to perform the conversion.

One drawback of using e-mail as gateway is the message delivery delay. There are at least two factors that contribute to the message delivery in case of e-mail posting. Firstly, when a message is sent from a mobile device to operator's SMTP server, the SMTP server forwards it to the mail server of the destination address. Some e-mail servers automatically reject the first attempts of any incoming e-mails in order to block spam, thus the operator's SMTP server must re-send the message after some period of time. The time between the first delivery and the second delivery contributes to the total message delivery time. The second factor affecting to the total delivery time is the polling interval of moblog plugin script. The script must poll the mailbox to which the moblog messages are sent. Polling is usually scheduled by *cron* or similar task scheduling software to occur every few minutes. While sending a moblog entry by e-mail can take several minutes, it does not have harmful effects on learning experience because moblogging is not a real-time event in the sense of interaction.

Another disadvantage is complexity of creating a new message. For each new message, one must fill in the receiver, subject, message body and possibly attachment. Since there are so many fields to fill, composing one message can take a long time especially if

the input method is cumbersome. To overcome this problem, a skeleton of a message including receiver information and possible key phrase can be saved as a draft. This way the blogger can create new messages simply by editing the draft.

5.2.2 Post by Custom Application

While nearly every mobile device supports e-mail these days, it is rather cumbersome to construct a new e-mail every time one wants to send a moblog entry. This is especially true when there are multiple attachments (pictures, videos and other files) to be sent. A more convenient way would be use a custom application with a powerful user interface that enables posting a moblog entry with minimum effort. For example such an application could be integrated as part of the camera application of the mobile device for enabling marking of newly taken pictures to be sent to moblog. Although the idea sounds great, custom application for moblogging poses several technical problems to be solved.

As for the programming language, for a custom moblogging application Java is the most feasible choice as most of the mobile devices have Java environment enabled by default. The gateway on the server can be written in Java, PHP or any other language that is appropriate in the server environment. The only requirement is that the server should listen to connections from the client (i.e. the mobile device) and process incoming posts properly. The client in turn must be able to establish a connection to the server in order to send the moblog entry.

User identification issue is easier to overcome in case of a custom application than it is in e-mail posting. For example the conventional username and password method can be utilised. Here, username and password can be the very same that are used when the user posts via web interface (i.e. normal blogging). One must keep in mind, however, that the username and password must be protected by encryption against *the man in the middle* attacks.

One option is to use the e-mail gateway described in the previous subsection. This, however, poses the same problems with user identification as in the e-mail posting scheme. The advantage of using direct posting over e-mail posting is that there are no additional delays in message delivery.

The major disadvantage of using custom software in moblog posting is the vast number of different mobile device models and different operating systems running on them. Even though Java is supported in most of the modern mobile devices, it is clear that developing a general application that would integrate with the device's features (e.g. camera) is extremely difficult. This is because different platforms use different, even proprietary (such as Windows Mobile's Camera Capture API), interfaces to access the device's media

properties (e.g. camera, audio recorder). If the used set of mobile devices is limited to a certain platform, say Symbian OS, developing such application would be much easier.

We were not able to find such client-end moblogging tool that would support access to any given mobile device's media properties in a uniform manner. However, in case of J2ME-enabled mobile phones, the situation looks brighter due to *Mobile Media API* (MMAPI), released by Sun Microsystems together with an expert group consisting of members of several major corporations of the mobile industry. MMAPI provides additional tools for J2ME developers in order to harness audio, video and other time-based multimedia capabilities of the device. Sun Microsystems maintains a list of J2ME devices together with information of MMAPI support of the devices [45]. The list, according to its assembler, is not exhaustive, thus more devices supporting MMAPI may exist. It must be also noted that MMAPI implementations may not fully support all features available in a mobile phone. For example Nokia 3650's implementation of MMAPI only supports capturing images with camera of size 160 x 120 while native camera application can capture images up to size 640x460.

Another problem of using a custom independent application is related to the lack of standardisation in posting APIs of different blog systems. Posting APIs are used to grant third party applications access to blogs' content. For example, Blogger has its own *Blogger API*, Livejournal uses the *Livejournal API* and TypePad has its version of Atom API, namely *TypePad Atom API*. Having so many different posting APIs means that writing a general moblogging application that would support any given posting API is troublesome. Some standardisation efforts for posting APIs have been established since Atom 1.0 standard was published in 2005. Perhaps in the future all popular blogging systems will conform to the Atom standard, thus making development of general moblogging software easier.

There already exist some mobile device applications for moblogging. One such software is Kablog and it has several versions available for different platforms. Furthermore, Kablog supports many of the most used blog software including TypePad, Movable Type, Blogger, WordPress, B2, Blog-City, UserLand, Roller and SnipSnap. The downside of Kablog is that it is licensed as shareware, thus it must be purchased after a trial period. Furthermore, a test usage of Kablog's J2ME version on Nokia 6270 revealed that the photo attaching feature is non-functional.

Nokia Corporation has released its own software giving a chance to Nokia mobile phone owners to start moblogging. Their software is called Lifeblog and it is supported only by a subset of devices manufactured by Nokia. It is expected, however, that the amount of mobile devices supported by Lifeblog will increase in the future. Lifeblog is a client-end application and as for the actual blogging platform, only Six Apart's TypePad is supported, thus using other blogging APIs with Lifeblog is not feasible.

BlogPlanet is yet another software enabling moblogging on mobile devices and it comes with two flavours. The first one is the full version which runs only on devices supporting Mobile Media API (MMAPI). The second version is the light version, running on Siemens 65 series phones. The light version excludes camera support and picture upload. BlogPlanet is a shareware program akin to Kablog, thus acquiring licenses will pose additional costs.

5.2.3 Post by Web Browser

When a moblog post is sent to blog system with a web browser, no additional software components are required in addition to a web browser at the client end. However, as input systems of mobile devices are often very limited, posting by web browser requires more actions from the user which makes the process complicated and cumbersome. In order to send an entry with a web browser user must first navigate to the correct URL and log on to the blog system. After this the user navigates to appropriate page for adding a new entry. The new entry form often embeds several fields to be filled and with a tiny screen of a mobile device, the view is rather inconvenient. It is possible that the web interface of the blogging software is not developed for mobile devices, thus it may be completely unusable. This problem can be partly solved by developing a specific *skin* for the blog system that adapts the page to fit the requirements of mobile web browsers.

Posting by web browser is expensive compared to other methods because the web browser downloads many unnecessary hypermedia objects such as text and images. This will also cause the process to be slower than other gateway methods; with a GPRS connection, it may take several seconds to have one page loaded and if sending a single message requires loading several pages, the total amount of seconds spent in waiting is great.

5.3 Implementation of Advanced Postman

While it is relatively easy to implement a gateway for a single mobile device, it is very hard to find a gateway solution that would work with every mobile device now and in the future. In order to cover as many devices as possible, a common feature of mobile devices must be recognised and then build the gateway based on this feature. According to previous discussion, such a feature is capability of sending e-mails; it is built-in on nearly all mobile devices and fairly easy to use. E-mail also offers existing infrastructure and protocols for actual gateway implementation. Possible usability issues of vendors' e-mail clients can be tackled by using a third party e-mail client, or by creating an easy-to-use blogging application which uses e-mail as a message delivery method. Posting by a

custom application is a fairly good alternative for e-mail gateway, but currently available products and development tools are vendor and platform specific, thus e-mail gateway was chosen to be implemented. Table 5.2 summarises features of discovered moblog gateway solutions depicted in the previous section. The gateway solutions are presented on columns and their respective features are presented on rows.

Table 5.2: Overview of moblog gateway solutions

	E-mail	**Custom application**	**Web browser**
Price	Free	Depends on license; usually costs some money	Free
Protocol	SMTP	Usually XML-RPC	HTTP
Suitable mobile platforms and devices	Any device and platform that supports e-mail	Depends on the application. Applications are usually platform and device specific.	Any device and platform that has appropriate web browser
Suitable blog systems	Any system where e-mail posting is enabled	Depends on supported posting APIs. Usually only a subset of all possible posting APIs are supported.	Any system providing web interface suitable for mobile web browsers
Message parsing and filtering	Messages can be parsed and different attachments handled in custom fashion (according to the plugin). Filters can be set to reject inappropriate content.	Message filtering and parsing is performed by the blog system. Customisation requires changing the core system.	Message filtering and parsing is performed by the blog system. Customisation requires changing the core system.
Existing software	Mobile e-mail clients are widely available. E-mail posting plugins are available for many blog systems. Some proprietary e-mail clients may limit size of the message.	There are some existing applications for blogging, but they are platform specific or otherwise limited. J2ME devices are not inter-compatible.	Mobile web browsers are widely available, but not all of them are capable of viewing web interface of blog systems. Cost of connection may also set limits to usage.
User identification	Sender field, subject prefix, digital signature	Username and password (are possibly sent unencrypted)	Username and password (are possibly sent unencrypted)
Bandwidth usage	Only bytes for new entry	Only bytes for new entry	Bytes for new entry and other contents of the blog site
Delivery delay	Up to few minutes	Instant	Instant, but accessing the blog system can be slow

Continues on next page...

Table 5.2: Overview of gateway solutions (continued)

	E-mail	**Custom application**	**Web browser**
Ease of posting	Requires some effort, but using draft messages makes posting fast	Requires some effort	Requires much effort; takes much time, if connection is slow

As stated in section 5.1.2, Nucleus CMS was chosen as the core blogging solution for ViSCoS. In search for an existing e-mail gateway solution, PostMan plugin for Nucleus was discovered. By default PostMan plugin only allows configuration of one e-mail account to retrieve posts for all blog users. It is clear that using only one e-mail account makes moblogging in a multiuser blogging environment inconvenient and difficult. A better option would be to specify an e-mail account per each blog in the system. To achieve this improvement, a modified version of PostMan plugin was created. A new plugin was baptised as *Advanced Postman* to emphasise its origin and enhanced functionality.

The heart of Advanced Postman is a script that retrieves e-mails from POP3 e-mail accounts specified on blogs' properties. The script is timed to run every N minutes by a UNIX cron program (or similar task scheduler). Messages sent to these e-mail accounts must be tagged with a special subject prefix which can be set on each blog's properties. Legal sender e-mail addresses can also be defined in order to avoid inappropriate mail to appearing in the blog. All messages without correct subject prefix or without correct sender address are ignored. Information of all approved and rejected posts can be stored in log files. This option can be turned on and off from the script. Table 5.3 presents options that can be defined for each blog established on Nucleus when Advanced Postman plugin is installed. These options must be correctly set before Advanced Postman can fetch and process messages for the blog. Each option is presented on blog's administration area with appropriate instruction message. The plugin has been designed so that adding a new language file (e.g. Finnish) for the options is easy.

Table 5.3: Options for blogs added by Advanced Postman plugin

Option	**Type**	**Description**
MailUser	Text	Username of the e-mail account
MailPassword	Text	Password of the e-mail account
MailURL	Text	Mail server address
MailPort	Text	Mail server port
MailRemove	yes/no	Flag of message removal after fetching it. This is strongly suggested to be 'yes', if the intention is not to debug the system.
DefaultUser	text	Default author of the entry. Must be a valid user on Nucleus

Continues on next page...

Table 5.3: Options for blogs added by Advanced Postman plugin (continued)

Option	Type	Description
DefaultCategory	text	Default category for the entries sent via e-mail
SubjectPrefix	text	Prefix string that must be included in the subject of the e-mail
AuthorisedEmails	textarea	Addresses of authorised e-mail senders mapped together with corresponding users on Nucleus
DefaultPublish	yes/no	Flag of immediate publishing

It is important to use a dedicated e-mail account to avoid unnecessary fetching of e-mails without proper credentials (spam and other e-mail messages). Furthermore, blog options, including e-mail account password, are saved to the Nucleus database unencrypted. Using encryption functions such as MD5 is not feasible because e-mail username and password must be sent in unencrypted format to the e-mail server. It may be possible to retrieve unencrypted password from the Nucleus database, if one has access to the server as a local user. Therefore it is highly recommended to use a dedicated, secondary e-mail account for moblogging with Advanced Postman.

Figure 5.4 illustrates the architecture of Advanced Postman. The figure shows components of the plugin, and connections between these components and the Nucleus CMS. The numbers depict the sequence in which the actions are taken. The actions are simplified to reduce complexity of the diagram. The process runs repetitively in intervals set in the task schedulers configuration. On each iteration the program retrieves options of a blog (presented in table 5.3), fetches new e-mail messages from respective mail servers, and parses new messages. Successfully processed messages are removed from the mail account if the mail removal flag is enabled in the blog's options. To avoid possible loss of important data, ignored messages must be removed manually.

One moblog entry posted by e-mail can consist several different elements. Usually these elements are subject, message body and some attachment files. The Advanced Postman has been built to process each of these elements separately. The subject field is processed first because it should contains a legal subject prefix that is defined in the blog's options. If the proper subject prefix is not found, the rest of the message is ignored and a new entry is not added to the blog. If the subject prefix and sender of the message are validated, the message body is processed together with the attachments. Advanced postman separately handles text, images, videos and applications according to MIME type embedded in the header information of the file attachments. Table 5.4 presents the attachment types and respective processing methods performed by Advanced Postman.

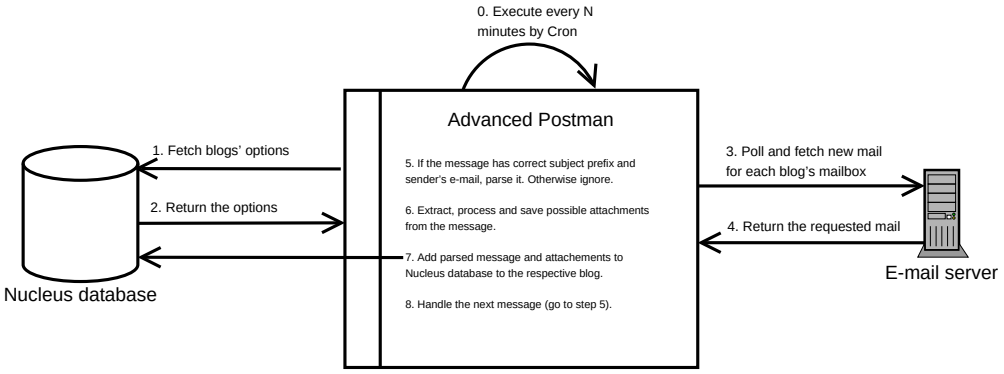

Figure 5.4: Architecture of Advanced Postman

Table 5.4: Attachment types handled by Advanced Postman

Attachment type	**Processing method**
Text	Text is processed as enriched, HTML or plain format and added directly to the new blog entry on the database.
Image	Images are saved to media directory on the server. Thumbnail images are automatically created from large images. Appropriate image tags are added to the new blog entry.
Application	Application files are saved to media directory on the server. Appropriate HTML links are added to the new blog entry.
Mobile video	3GPP file types and its derivatives are saved on the media directory on the server, and then embedded to the blog entry as QuickTime objects (QuickTime is able to play these files). Embedding can be also disabled from the script, after which the video file is simply added as an HTML link to the new blog entry.
Shockwave Flash	Flash file is saved to media directory on the server, and then embedded to the new blog entry as Flash object.
Other attachments	File is stored to media directory on the server and a simple HTML link pointing to the file is added to the new blog entry.

Advanced Postman utilises PEAR library to enable POP3 protocol and MIME decoding operations. It also uses ImageMagick software or PHP's GD library for image processing purposes. Video processing software FFmpeg is used by the Advanced Postman to capture a frame for a preview image from a mobile video file. However, on the server where Advanced Postman was installed for testing, FFmpeg was not installed, thus usage of FFmpeg was disabled during testing.

Chapter 6

Analysis of the Use of Advanced Postman

After development of the first prototype version of Advanced Postman was finished, we provided mobile devices and moblog accounts to test users in two different test scenarios during the Fall of 2006. In this chapter, we analyse the two usage tests. In addition, we reflect on the technical and usability issues of the Advanced Postman. Despite the partial lack of motivation of the test subjects, the tests were successful because they resulted to discoveries of deficiencies and bugs from the system, and suggestions for new features emerged as well. Furthermore, students provided good ideas on how to improve motivation and collaborative community with moblogs. It must be noted that the term *blogging* is used here to refer to both ordinary blogging and moblogging.

6.1 First Test Scenario - Technical Test of Moblog Prototype

In the first phase of evaluation, seven international IMPIT (International Master Program in Information Technology) students (2 male, 5 female) were provided with mobile devices and individual moblog accounts. The test had two primary goals: to generate ideas on how moblogs could be used to establish a collaborative community in ViSCoS, and to find out bugs, deficiencies and missing features in the system. The test was not part of any course and the student participation was completely voluntary. This kind of test is part of participatory design approach where the intention is to gather tacit knowledge from users of the system which might reveal important details that would not otherwise be discovered.

Figure 6.1 presents a screenshot of a web interface of the moblog system used in the first usage test. The moblog system hosts moblogs of several authors. The screenshot presents the main research moblog where all questions were posted for participants to answer. Some of the questions were directly answered by commenting on these posts, but users discussed these questions in their personal moblogs as well. Other moblogs of the system are accessible under the *Links* menu.

Figure 6.1: Test 1 - The main research blog

6.1.1 Course of the Test

The students were given Nokia Communicator 9500 smart phones equipped with camera, QWERTY keyboard, access to the Internet, 1GB memory card, and an onboard e-mail client. A prototype of Advanced Postman was then installed to Nucleus CMS and individual moblog accounts created for each student. In the experiment kickoff session, the students were given instructions on how to use the system and they were helped to

configure the mobile devices. They were then asked to read a paper written by Nagel et al. [50]. The paper makes analysis of ViSCoS programme and comes to conclusion that ViSCoS lacks collaborative community. Based on the paper and a short introduction to current challenges in ViSCoS, the students were asked to discuss in their moblogs about the contents of the paper. Other questions were presented regularly as well. Since discourse took place in a moblogging environment, it provided preliminary information of usefulness of Advanced Postman as a moblog technology.

The students were not able to browse and comment on each other's moblogs from mobile devices because a web interface of Nucleus CMS was not built for mobile access. However, they could read and commented on each other's moblogs from their desktop computers, thus making comments often longer than actual entries posted from the mobile device.

During the first week, each student posted in average 4.14 messages (including about 2 test entries per person), but within two weeks the posting frequency decreased to 1.29. One reason was the examination period; the students simply did not have time for moblogging. Another reason for this is the lack of interest in the subject. Many of the students were thrilled to get their hands on Nokia Communicator devices, but after a while the excitement faded and their motivation to post decreased. It happened just like Krause [39] and Richardson [63] had warned; the students did not have internal motivation to continue moblogging after the excitement of new technology was gone.

The test period lasted about three weeks in October 2006. There were a few students who were active in the discussion while other students did not send many posts during the test. Table 6.1 presents questions to which the students were asked to post answers to during the test. First column of the table presents the question, second column explains relevance of the question, and last column shows the answering frequency for each question. Frequencies are on three level: low (1), medium (2) and high (3). There were other questions in addition to those presented in the table, but they merely refined previously started discussions.

Table 6.1: Questions posted to students

Question	**Relevance**	***f***
1. What is your opinion over the article (by Nagel et al.)?	This question strove to seek answers to general intelligibility and opinions of the article that students were asked to read before starting moblogging. Relevance of this question was not very big for this thesis; it merely tried to measure whether students understood what they read.	2

Continues on next page...

Table 6.1: Questions posted to students (continued)

Question	Relevance	*f*
2. How could you use mobile blogs to enrich every day life of a student?	This question was asked not from educational point of view, but from every day life point of view. Intention was to discover ideas on how moblogs could help students' every day life.	1
3. How to create and maintain a powerful learning community with help of mobile blogs?	This question was directly targeted against one of the results of the article; ViSCoS lacks collaborative community and students were asked to generate ideas how moblogs could be used to establish such community.	1
4. Is it possible to establish a friendship within a blogging community? If yes, how is it done? If no, why not?	By issuing this question we hoped to get ideas from students on how friendships could be established in a blogging community. Friendships would improve motivation of the bloggers.	3
5. Blogs and moblogs can be used as personal learning tools to reflect ones learning. Can you think of any other ways to utilise blogs as a technology in (distance) education?	Intention of this question was to gather new ideas on (mo)blog usage scenarios.	2
6. In order to create a development community (or any other community), the potential members of the community must be motivated first. Novelty of a technology is a good motivator to someone, but after a while it is not enough; people need more motivation. How would you provide motivation to blog/moblog to them?	The intention was to get answers to profound question: how to motivate mobloggers (in ViSCoS)?	3
7. As you know, most of the camera-equipped phones have also video shooting capability. Any ideas how mobile videos could be used in education?	Although sending video from a mobile phone to moblog is currently expensive, in the near future it might be very common practise. Therefore we posed this question to students in order to gather ideas on how mobile videos in general could be used in education (both distance and traditional education).	2
8. Is it difficult or too slow to write a new blog entry with your mobile device? If yes, what's particularly difficult and how would you make it better?	This is one of the questions trying to find out how students perceived usability of the moblog system and the devices.	3
9. What are the annoyances of the moblog system (i.e. things that nearly drive you crazy when you are using the system)?	This is another question attempting to discover usability issues of the moblog system.	3

Continues on next page...

Table 6.1: Questions posted to students (continued)

Question	Relevance	*f*
10. Your experiences of the test. What was good? What was bad?	This question was the first of the final questions posted for the students to answer. Intention of this question was to find out students' opinions on the test.	2
11. Could you imagine using moblogs later? If yes, how would you use them? If not, why not?	This question was the second of the final set of questions posted for the users to answer. By this question we wanted to find out the outcome of the moblog experience of the students. Answers are useful for the future.	2
12. Give any ideas how the system could be improved. Also crazy ideas are welcome!	This question was the third of the final questions posted for the users to answer, thus being the last chance for the students to give suggestions for improvements. This question was discussed in other forms earlier, but we hoped that perhaps new ideas would emerge at the end of the test.	2

Questions that were not directly answered at all were 2 and 3. The reason might have been bad disposition of the question or wrong period of time (e.g. just before the exam). Questions that received only a few answers were 1, 5, 7, 10, 11 and 12. Popular questions for discussion were 4, 6, 8 and 9; they resulted to discovery of interesting ideas and remarks. Overall, the students' response to the questions varied, as some students were eager to discuss the questions at hand while others were not, therefore not answering any of the questions.

6.1.2 Suggestions for Motivation

One of the major concerns in mobilising blogs to ViSCoS is how to make students interested in blogging so much that they would carry on after the first external push. Students taking part in this test were asked to discuss possible motivators for blogging and reasons for lack of motivation. They were also asked to discuss motivation issues in distance learning in general. Table 6.2 presents findings together with appropriate comments. The first column presents actual ideas and remarks proposed by the students and the second column consists of relevant commentary.

Table 6.2: Ideas for motivation

Idea/Remark	Additional comment
Make moblogging community stronger by establishing acquaintanceships and friendships.	If people of a blogging community share common interests, they are more likely to form acquaintanceships and friendships, thus making the community stronger. In ViSCoS the common interest is naturally studying and computer science in general. If two persons become friends during ViSCoS studies, they are likely to nurture the friendship by reading and commenting each others blogs. If two persons are already friends before joining ViSCoS, they are likely to extend their friendship to blogosphere. There are different ways to establish a friendships, but they depend much on individuals. A general solution therefore cannot be provided. Probably the best way is to encourage casual discussion and sharing own interests to others.
Increase motivation by making moblogging fun	A student asked: *Is it realistic to think that there is a way of getting people to blog without them taking it up as a hobby on their own?* Another student approvingly answered: *I think that they (ViSCoS) should make blogging more fun and interesting, even if the original idea is to use blogs in learning it will do no harm to have a little fun now and then.* These students are referring to motivation from the enjoyment point of view. This makes sense because if learning is made fun, motivation is high. However, the students did not comment on the most important aspect of this matter: how could we make educational moblogging fun?
Motivation of high school students may suffer from participation to student parties and other courses.	Based on this remark, the "fun to blog"-idea could be incorporated to parties and to other courses students are attending; ViSCoS students could be given free hands to write to their moblogs anything about their lives, including ViSCoS studies. This way motivation will keep up and students will have always something to write. Not only high school students would have the advantage on this, but older students could also learn to author moblogs about their lives and about ViSCoS studies.
Moblogging could be helpful for shy students to present their thoughts.	In addition to moblogging, this applies to all activities of online learning. Anonymity would improve this situation even more, but that would remove possibility for grading. Alternatively entries could be anonymous to other students, but teachers could be able to distinguish the author.
Arrange ViSCoS student meetings quarterly or semiannually.	Currently in ViSCoS there are no such meetings. These meetings could improve motivation of students because they would meet their teachers face-to-face and they could form connections to other course participants as well. However, blogs and moblogs could compensate the lack of teacher-student and student-student meetings. In this modern age moblogs and social software form the technology that shrinks the world rapidly, so perhaps in the future most of the social interaction is performed via social software.

Continues on next page...

Table 6.2: Ideas for motivation (continued)

Idea/Remark	Additional comment
Lack of teacher-student interaction has negative effect on evaluation and motivation.	This is particularly true, if course evaluation is based on subjective reasoning. Moblogs and blogs could enhance evaluation process because then teacher could actually become aware of problem solving process of the student. This result implies that the moblogs are written diligently.

6.1.3 Discovered Bugs, Annoyances and Missing Features

One of the goals of this test was to discover bugs and deficiencies of Advanced Postman and of the entire moblog system in general. According to several moblog entries posted by the students, the technical aspects of moblogging could be improved. One student even stated that the effort of the test should be placed in improving the technical structure of moblog rather than emphasising the possible usage scenarios. Table 6.3 presents bugs, annoyances and missing features reported by students or otherwise discovered during the test. Each discovered element is listed together with a solution proposal.

Table 6.3: Bugs, annoyances and missing features

Discovered element	Solution
E-mail polling did not work because the Net_POP3 PHP module's login() function used APOP feature by default and the POP3 server used in testing did not support it.	APOP parameter was changed to FALSE
Image and video attachments were not added to the post	Bug was tracked to Advanced Postman code and was eliminated
Embedded video caused some web browsers to freeze	Video embedding feature was disabled and video was added as a hyperlink
Complexity of moblog settings caused problems; for instance messages were sometimes sent with wrong subject prefix	Once settings are correct, they work well. Some settings could perhaps be automated by using default values. Client-end improvements can be achieved by using draft messages containing necessary details (i.e. subject prefix, sender, recipient) when constructing a new entry.
Delivery time (a few minutes) was too long; students wanted to see the post immediately	Delivery time can be decreased by adjusting spam filters and cron task scheduler. Unfortunately in e-mail gateway method the delay cannot be completely removed. One idea to help users in this situation would be to establish a delivery report system in which user would get delivery report by SMS as soon as the message has been added to the moblog.

Continues on next page...

Table 6.3: Bugs, annoyances and missing features (continued)

Discovered element	Solution
Used devices (Nokia Communicator 9500) were too heavy and too large.	This problem can be tackled by equipping mobloggers with lighter and smaller devices that are capable of sending e-mail and taking pictures
Keys on the keyboard were too small and too close to each other, thus text input was cumbersome and slow	Problem can be solved by using external keyboard or alternative text input method. Predictive text entry (e.g. T9) would also make text entry faster. One could also consider using voice recognition software, as suggested by one of the students.

6.1.4 Suggested Usage Scenarios

Students were asked to develop ideas on how moblogs could be used in a distance learning course such as ViSCoS. Many ideas emerged, but some of them were similar to those presented earlier in this thesis. Table 6.4 presents the ideas captured from moblog entries of the students together with additional comments. All these ideas can be utilised in some ways in ViSCoS courses.

Table 6.4: Moblog usage ideas

Idea	Additional comment
Take instant notes	A student who presented this idea stated: *Most ideas, especially in programming normally pop up when least expected and it will be good to get them noted down as quick as possible before one forgets*, and she continued: *I remember there was a time I had to wake up in the middle of the night to type a piece of code that had just came to my mind*. This case illustrates the usefulness of mobile technology in education in general. Time- and location-agnostic note taking could be useful in ViSCoS context where learning is bound to occur anytime and anywhere.
Use moblogs as Q&A (Questions and Answers) discussion forums in order to create a developer community.	This could be particularly useful for programming courses where students are often struggling with technical problems. Currently some ViSCoS courses have Q&A forums established on Moodle DLE. However, using moblogs as Q&A forums would enable better commenting, linking and interaction.
Use moblogs as learning diaries as part of the curriculum.	If the moblog content is graded, students are likely to write more. Alternatively, moblogging could be optional for students; the course could be evaluated either by moblogging activity or by exercises, for example.

Continues on next page...

Table 6.4: Moblog usage ideas (continued)

Idea	Additional comment
Moblogs could be used in preparation for examination.	This way course participants could discuss about problems and solutions, thus gaining better understanding together. The scenario is familiar from situation where group of friends are preparing for examination together. When correctly used, this method is quite powerful. One could even consider using exam preparation moblogs as part of evaluation; students who contribute much are given extra points added to the examination score.
Enable rating of entries and comments posted to moblogs.	ViSCoS students could rate both entries of other students and comments posted by others to their own moblog. Rating could be then used as part of evaluation, but some moderation is required to ensure trustworthiness of the ratings. Rating system would be very useful for example when students are asking help with solving a problem from other students.

In addition to aforementioned moblog usage scenario suggestions, the students' moblogging process could be facilitated by establishing a set of moblogging rules. For example setting a minimum amount of moblog entries per week will ensure that a student writes regularly to his moblog. Announcing mandatory questions to be discussed is also possible in educational setting. Although these methods may cause reduced quality of moblog entries, it is still better than no entries at all. In case of a sensitive topic, students could be granted anonymity so they could post entries and write comment without self-censorship.

6.1.5 Usability Experiences

Some students first had some problems with remembering to include proper subject prefix to the e-mails that were sent from their mobile devices. The sender's e-mail address was always correct because once it is configured, there is no need to change it later. The delay of message delivery caused some students to send messages several times. Since this test was not the actual usability test, students were assisted while configuring the devices during the kickoff session. However, it is imperative, for the future account, that users are provided with very clear instructions on how to configure both the mobile device and the moblog account so they work seamlessly.

According to the students, other parts of the moblog system were satisfying. The web interface of the system was considered easy to use and the students had no troubles with the e-mail client. From the administration point of view, Advanced Postman could provide some global options that could be operated by the administrator of the entire system. These useful options could include for example debug flags, logging settings and options for different features such as image resizing and video embedding.

The usability issues of mobile devices are vendor-specific and they cannot be affected by Advanced Postman. Many discovered usability problems of mobile devices were derived from slow and uncomfortable input method. The screen size or other features of the devices were not criticised. In order to determine the best possible mobile device for moblogging, another investigational study covering usability on a set of selected devices should be performed. It is clear that all possible devices cannot be tested, thus those with the most promising features should be included in the study.

6.1.6 Other Observations

Some students obviously felt tempted to use the credit on the mobile device for other activities than moblogging although they were asked to use the devices primarily for moblogging only. According to logs and messages stored on the device, some students for example called and sent messages to their friends. The web browser was also used for reading web pages. This phenomenon was expected because students did not use their own mobile devices. In order to avoid secondary use of test devices, some limits could be set on telephony and Internet access in order to allow moblogging only. However, this would set restraints to users and could decrease their motivation. The best way to deal with this problem would be to introduce limitless data access which is based on a monthly fee.

In addition to sending answers to posed questions and discussing about moblogs in ViSCoS, the students used the mobile devices to record interesting events in their lives. For example two students posted pictures of the first snowfall because they had not seen snow before. Examination of one difficult course participated by the students was also a popular topic of discussion. This behaviour hints that if students have similar interests concerning studies, they are likely to write about it in addition to mandatory, enforced discussion. The third example of irrelevant moblog discussion occurred when some students had an idea of having a blog system or some other community tool for themselves in order to improve the study process and get help from each other. The students clearly discovered the advantages of asynchronous discourse enabled by blogs and moblogs. These examples prove that in addition to mandatory content, students are interested in writing about things that are more meaningful for them.

6.1.7 Conclusions of the First Test

Despite the problems with lack of motivation of test subjects at the end of the test, the results provide important information on how Advanced Postman can be improved

and ideas on how moblogs can be effectively established in a distance learning programme such as ViSCoS. The motivation problem suggests that voluntary moblogging on a strange topic is bound to diminish in the long run. Therefore we have two challenges in order to make moblogging work in ViSCoS. Firstly, the students must be pushed to write with some kind of carrot (e.g. extra points, part of evaluation). Secondly, once writing has started, the writing must be kept fun and interesting to the students. By establishing these two aspects in moblogging, students are more likely to reflect their learning in moblogs and therefore improve their learning processes.

Concerning motivation of ViSCoS students, several ideas and remarks were authored by the test subjects. This information is useful when moblogs will be officially introduced in ViSCoS. For example, one remark was that if a student is shy, then moblogging could help him to write to the audience. This is even more true if the moblogging system could offer anonymous authoring. Another example of a good remark was that in addition to motivation, lack of teacher-student interaction also reflects upon evaluation. Moblogs could be used to boost teacher-student interaction which would be useful both for motivation and evaluation.

The test yielded plenty of interesting ideas on how moblogging could be used in ViSCoS. The most interesting and useful ideas were to use moblog as an instant logging tool, Q&A forum, learning diary and exam preparation tool. Another good idea was that moblog entries and comments could be rated by the users of the moblog community, and the ratings could then be used as part of evaluation.

Usability experiences of Advanced Postman were generally good. However, the delay in posting the message was noticed by several students. Mobile devices, however, were criticised much; particularly their large size and inconvenient input mechanism were targets of discontent.

6.2 Second Test Scenario - Moblog in Programming Project Course

In the second usage test six Finnish students (6 males) of a programming project course (not ViSCoS) were given mobile devices to be used in logging software development activity on and off university campus. While users of the first test scenario used individual blogs, the second test scenario consisted of a single project blog in which each worker of the project posted entries. The intention of the second test scenario was to validate usefulness of Advanced Postman as well as to find out how well it can be used in the programming project course as a logging tool. Since this test is still in progress during writing this thesis, we present only the preliminary results.

6.2.1 Course of the Test

Four of the students were equipped with HP iPAQ 6515hw PDA device and two were handed Nokia Communicator 9500 smart phones. The basic properties of both of these devices are similar, but the keyboards are slightly different; in the Nokia Communicator, the keyboard is placed on a larger area, thus making it easier to use. In addition to a built-in keyboard, HP iPAQ has a Stylus pen and a software for recognising handwritten notes. Deployment of these devices did not yield any problems; the only difference was that for HP iPAQ the Internet settings had to be set manually while for the Nokia Communicator the settings were received from the network operator automatically.

Moblog configuration was performed by the moblog administrator and therefore the students were not required to change the settings of the moblog. This course of action was chosen because if all the students would edit the moblog settings concurrently, the settings could become corrupted. Due to this fact, analysis of usability of Advanced Postman configuration is unfeasible in this test scenario.

After the students independently configured the devices, they started sending messages to the project's moblog. During the three weeks of observation, new messages appeared in the moblog almost daily, excluding weekends. Some exceptions to this rule occurred as a few entries were posted on saturdays and one even on sunday. The total amount of posted entries was 30 and the average amount of messages per day was 2. Weekends were excluded from average calculation and posts from weekends were added to previous Friday because the official project working days were from Monday to Friday. The amount of entries for the first three weeks of the test were 13, 7 and 10, respectively. Entries consisted mostly of daily activities, events, opinions, suggestions and links to resources. The length of moblog entries varied from 1 to 118 words. The shortest messages were merely used for testing the device configuration. The average length of the entries was 47.03 words. Subject fields of the new entries were excluded from the word count.

Only a few entries had comments and those usually consisted of additional data that the author wanted to add to the original entry. Figure 6.2 presents a screenshot of the programming project moblog authored during the test. Finnish language was used in writing the entries because that was the language of the programming project course. Figure 6.3 presents a translated entry from the second test which quite well depicts the nature of the moblog. A person outside the project would probably not understand much of the entry while it may be perfectly clear for the members of the project team.

Figure 6.2: Test 2 - The programming project moblog

If we try to relate this kind of moblog usage to applications presented in table 4.2, we notice that the *Student's blog* application is the closest match; students are regularly posting events and activities of their programming project, and teacher's role is an observer. In this particular course, moblog will not be used as part of course evaluation. According to Krishnamurthy's blog dimensions presented in figure 4.2, moblog usage in programming project like this fits best in *Collaborative authored columns* quarter of the diagram. Even though the students were not actually authoring a column, they contributed to a common moblog in order to save the history of the project's activities.

All right, some things were clarified in the meeting, but focusing of builds and components to be implemented on iteration remained partly unclear.

Things will probably get clearer after I manage to contact Antti, in the meanwhile I'll work on implementation diagram.

There was nothing new and conclusive in the meeting, tomorrow we will have a meeting after Wrap-up where everybody are present.

2h meeting
8h --> 19:00 Implementation model quite ok, KHS updated, MP updated.

Figure 6.3: Translated entry from the second test

6.2.2 Discovered Bugs, Annoyances and Missing Features

Due to requirement of multiple authors of a single moblog, Advanced Postman script had to be enhanced because its first version was built to host moblogs of individual authors only. After a few changes to the Advanced Postman code, the moblog system was able to distinguish each user by their e-mail addresses and by blog system usernames, and therefore new messages were posted under correct usernames. The second test did not reveal any other limits or deficiencies in Advanced Postman.

The input system of the Nokia Communicator 9500 was considered rather slow, but otherwise functional. The HP iPAQ device was not criticised at all, though we did not perform a query on device usability during the first three weeks. A detailed usability results of the devices will be reported in the next publication.

6.2.3 Conclusions of the Second Test

After the three weeks of observation in the programming project's moblog, we noticed that it had been used more actively than the moblogs in the first test. Particularly interesting was the observation that the frequency of posting did not decrease as it did in the first test. The reason for this phenomenon is probably the fact that moblog was integrated to their work process, thus students were obligated to write new entries regularly. We noticed that the length of entries slightly increased during the three weeks of moblogging; average of entry lengths for each week were 37.85, 41.71 and 62.7, respectively. This could be explained by the fact that the students became more familiar with the devices and with moblogging in general. However, we do not know if the students attained internal motivation for moblogging, or if it was just due to the course requirements. Interesting question revealing the true state of internal motivation for the future is: will the

students use moblogs or blogs in their studies or work after the project has finished?

Due to the nature of the moblog (activity log), the students did not have much discussion on the entries as form of commentary. Most of the comments were written by authors themselves in order to add details to the entry. The usage of commenting could be boosted, if the moblog web interface would be accessible with mobile devices. The users would also need mobile Internet connection with adequate speed (e.g. 3G network), thus making web surfing experience bearable.

The test continued and was completed after this thesis and therefore the final results were not presented. However, we plan to release the results in another article in the future.

6.3 Reflections on the Advanced Postman

Facts of the previous sections were results of usage tests analyses. In addition to previously presented information, we noticed other aspects of Advanced Postman which are hereby reflected and analysed. The established moblog system is considered both from a technical and usability point of views.

6.3.1 Technical Aspects

During the testing, performance-related operational limits of Advanced Postman were not discovered. The system was able to run eight moblogs simultaneously without any performance problems. However, suitability of Advanced Postman for large scale systems is still an open question. From a technical perspective, Advanced Postman does not require much processing power, but data communications must be well established due to the e-mail polling feature; if there are tens of e-mail accounts to poll and each of them host tens of messages, the polling process is likely to be timewise expensive. Furthermore, since Advanced Postman does not automatically remove ignored messages, this should be taken care of manually; otherwise e-mail accounts with large numbers of irrelevant e-mail messages may have a slowing effect on the system. Based on these facts, performance tests should be executed before deployment of the product in a large scale environment.

Support for e-mail protocols other than POP3 has not yet been implemented. Other protocols, however, are fairly easy to add by using available libraries such as PEAR. Different message encryption schemes are also currently missing and should be developed, if Advanced Postman is to be used to process confidential data. Digital signatures such as PGP

are not yet supported, and this could be something to be developed in the future in order to make user authentication even more secure.

Moblog sites used in this evaluation could not be effectively viewed by the mobile phone browsers for two reasons. Firstly, the web interface of Nucleus CMS is not optimised for mobile devices, thus tables and other structures are distorted on the screen of the mobile device. Secondly, students of both test scenarios were given credit worth 10 euros and they were told to use it only for posting new entries to their moblogs. This restriction was set because otherwise the credit would run out fast. Because moblogs could not be viewed well with mobile devices, commenting was mostly done with normal computers. In order to enable commentary feature on mobile devices, they could be equipped with software that merges functions of RSS/ATOM-reader and commentary tool; commenting would then be fast, efficient and would not require much data traffic. Another solution to this problem is to use content adaptation to make the entire blog site accessible for mobile devices. The commentary problem must be solved before true bidirectional moblogging can be established.

6.3.2 Usability

Advanced Postman is a fairly usable tool, but it has some usability weaknesses. The amount of moblog options set by Advanced Postman makes configuration of the system quite complex. Once the user learns how to tweak the options, there is no problem. However, learning takes time and if the users are not assisted by the administrator or by someone else, they might even give up on it. In order to improve this aspect of usability, necessity of options of Advanced Postman should be evaluated and user's manual written. Another problem of usability is the delay time in message delivery that was also noticed by the students in the first test. This problem cannot be completely removed, but users can instead be informed on existence of the delay, thus making the delay more acceptable. Before Advanced Postman is deployed, usability of it should be evaluated by a group of ViSCoS students.

From the administrator point of view Advanced Postman is easy to use. Installation of Advanced Postman on top of Nucleus CMS can be executed easily by standard Nucleus plugin installation method. After installation, cron task (or similar) must be set to launch the polling script for any desired amount of minutes. Then everything is ready and new moblogs can be configured by filling in moblog options; although this is mostly done by users, an administrator can perform it as well. Removal of Advanced Postman is just as easy as the installing procedure. If the administrator has adequate skills in PHP programming, he/she can easily modify the Advanced Postman code to fit it better to a particular moblog system.

Chapter 7

Conclusions and Future Work

Distance learning has become an important branch of education since the past few years and the trend is likely to grow. ViSCoS distance learning programme was developed at Department of Computer Science, University of Joensuu, to provide basic level computer science courses for those who do not live close to universities or who do not have the possibility to enroll to ordinary computer science courses due to other studies or work. As any distance learning programme, ViSCoS has also met problems related to students' motivation and performance. In this thesis we provided an overview of the latest scheme of ViSCoS, ViSCoS Mobile, which has a clear focus on bringing ViSCoS studies into mobile devices. One of the three development threads of ViSCoS Mobile is *student support services* and within this thread we recognised the potential of mobile blogs as a learning tool. The primary aim of this thesis was to determine the most feasible methods of establishing and utilising a mobile blog system for ViSCoS Mobile based on background theory and analysis of existing solutions.

As the first research question we asked: *How can mobile blogs be used in a computer science distance learning course to increase motivation and collaboration of the students in order to build a stronger learning community?* Based on the existing theory on blogs in general and brand new ideas for context of ViSCoS, we concluded several scenarios of using moblogs in ViSCoS. The most important findings are listed on table 7.1 together with usage examples in ViSCoS and estimation on feasibility. Even without concrete usage experiments in ViSCoS, we can see that moblogs have much potential in distance learning environments.

Table 7.1: Moblog usage scenarios concluded

Use moblog as	Example of usage in ViSCoS	Feasibility
Problem solving tool	Use moblog for example on programming courses to solve problems with peers.	Very feasible. Problems can be recorded as they appear and other students can contribute by commenting the original post.
Discussion tool	Discuss, debate and publish own thoughts on any course.	Very feasible. Structure of blog is optimal for discussion and publishing ideas. For longer discussions alternative input method should be implemented as input system on current mobile devices are cumbersome.
Reporting tool	Report regularly progress of the programming project in Programming Project course.	Very feasible. Reporting can be for example done at the end of the day when student is on her way home by bus.
Learning diary	Use moblog as learning diary on Introduction to Ethics of Computing course to note down thoughts, ideas and to reflect learnt material.	Feasible, but slow input mechanism of mobile devices sets limit on length of the entries.
Preparation tool	Prepare for examination of ViSCoS courses with moblogs; share the knowledge with peers.	Feasible, but adaptation might be difficult because traditional preparation method is study alone for examination.
Linkdump	On Introduction to ICT and Computing course students could post observations of new technologies and innovations to their moblogs.	Very feasible. This way all students can benefit from interesting links discovered by one student or instructor.
Collaboration tool	Moblog is potentially powerful collaboration tool for courses like Research Fields of Computer Science, where students are co-authoring a scientific report.	Feasible. Again, long entries might be difficult due to slow input system of the devices.
Bonding tool	Moblogging students who share the same interest are likely to become acquainted and even form friendships. Bonding of students is the first step towards a a stronger learning community. This also applies to teacher-student relationships.	Very feasible. This is important for the sake of internal motivation and should be promoted.

Continues on next page...

Table 7.1: Moblog usage scenarios concluded (continued)

Use moblog as	Example of usage in ViSCoS	Feasibility
Instructor's tool	Instructors of ViSCoS can utilise moblogs to deliver information to students, to start discourses, and to evaluate and even predict students' performance.	Very feasible. Moblog is particularly suitable to be used for communication between instructors and students. Instructors should show example on how to use moblogs in order to encourage students to follow their example.
Fun factory	Make moblogging more fun by allowing students to write about their every day life besides actual ViSCoS courses.	Feasible, but not everybody want to publish their lives online.
Shield against shyness	A shy student in ViSCoS might be afraid to discuss about course contents on traditional discussion forums incorporated in Moodle DLE. Therefore moblogs could be useful for the student in order to express his thoughts under an alias or even anonymously.	Very feasible. Many Finnish students are shy by their nature, thus moblogs can facilitate their learning processes.

Students also require motivation in order to start and continue moblogging. We discussed motivational issues in several chapters and concluded that without internal and external motivation, blogging in general is a lost cause. External motivation can be achieved for example by means of rewarding or even obligation. Internal motivation is more tricky to handle. We discovered that friendship with other bloggers could be a trigger for internal motivation. In general, bloggers need active an audience in order to write actively, thus commentary authoring should be encouraged among ViSCoS students.

The second question stated: *what is the most feasible way to construct a mobile blogging environment for ViSCoS?* To address this challenge we searched through existing solutions and concluded that the most feasible way for ViSCoS is to create a moblogging system on top of Nucleus CMS blogging system. This path was chosen because moblogging is very similar to ordinary blogging; only the method of delivery differs. Furthermore, Nucleus CMS provides multiple blogs for multiple users, thus being the optimal choice for a distance learning environment where multiple students can contribute to multiple blogs simultaneously. As for moblog message delivery, we covered several alternative gateways and ended up choosing e-mail gateway as the most appropriate solution in terms of maintainability and a wide range of supported platforms. The e-mail gateway was implemented as a Nucleus CMS plugin and it was given the name Advanced Postman. In addition to design and implementation of the moblog system, incapabilities of mobile devices were analysed and several solutions were proposed to correct them.

Advanced Postman was evaluated and analysed in two different usage tests. The first test was also used to generate ideas for moblogging in ViSCoS. Both tests provided important data on usability of the system, and some errors were discovered as well. In addition to individual errors in the source code, the tests revealed that Advanced Postman has only one major disadvantage: delay of message delivery. Several users complained that the few minutes of message delivery delay sometimes resulted to duplicate entries as the users retried until the new entry arrived to the moblog. Many users also mentioned the bad usability of the mobile devices, but that is not directly related to Advanced Postman. In our own reflections, we concluded that more tests should be executed, particularly on usability of configuration and performance. Moblog browsing from mobile devices could also be improved. Additionally, browsing and commentary of moblogs was recognised to be done mainly from desktop computers because Nucleus CMS system did not offer a web interface that would look good on mobile devices. Commentary from the mobile devices must be enabled in the system to make it completely independent from desktop access in the future. Overall expressions of Advanced Postman were positive, but there were some aspects that require improving and further investigation. The test results suggest that Advanced Postman could be used in ViSCoS after making enhancements on usability and executing performance tests successfully.

As the third research question we wrote: *what are the needs in ViSCoS Mobile and how does this work fulfill the needs?* In chapter 3, we presented among other information the idea of ViSCoS Mobile, its challenges and requirements. Three development threads were identified as: content adaptation, programming on mobile device, and student support services. This research contributed on the need of student support services and particularly on establishing moblogs in ViSCoS programme. We have managed to provide the reader with information on suitability of moblogging for distance education courses such as ViSCoS. Our conclusion is that moblogs can be used to support students' learning process in various ways, as presented previously in table 7.1. The next step is to confirm these findings with deployment of moblogs and appropriate testing within the ViSCoS community.

The fourth and last question was formulated: *how blogs and moblogs have been previously used in education and in research?* In order to gain good understanding of existing work on blogs and moblogs, a literature review was performed and interesting ideas were recorded. We discovered that blogging had been used in several different educational environments from traditional classrooms to distance learning classes. Several cases of successful implementation of blogs in education were found, but also reports of bad experiences on educational blogging were presented. The existing research on moblogging in education resulted to several findings, but none of them concerned moblog usage in a computer science distance learning course. Moblog research reports showed that new technologies such as GPS and RFID can be used to enable pervasive learning experience where the real world is mixed with virtual reality.

If we try to relate Advanced Postman and moblogs to the requirements for successful educational mobile services presented in table 3.4, we can notice that not all requirements were fulfilled. *Simplicity* could yet be improved in terms of usability. Advanced postman offers some *automatic functionality*; for example the messages are automatically parsed and the user does not have to mind about embedding different media types to a new post. *The Numbers of actions required by the user* are not many, but they could still be reduced by allowing instant moblogging on the press of one button. The *amount of cognitive load* is low because moblogging mostly concerns reflection and communication at location and time that are most suitable for the learner. The moblogging system constructed with Advanced Postman partly fails to meet the requirement of *knowledge navigation*; it does not ensure the users' ability to browse and comment on each others moblogs with their mobile devices. Other requirements presented in the table 3.4 are not applicable to this project as they concern more applications with actual learning content and tasks.

7.1 Future Directions

Blogging was introduced to ViSCoS students in September 2006, but due to voluntary participation, none of the students signed up for a blog. By the Fall of 2007, moblogging will be introduced to ViSCoS students by attaching it as part of the curriculum. Actual deployment details have not been planned yet, but moblogs will probably be first used as reporting tools and learning diaries. Blogs and moblogs will also be used as part of evaluation, thus good participation will pay off at the end of the course. We intend to further run tests on usability of Advanced Postman within the ViSCoS community in order to improve the system. The question that must be answered is: how well implementation of Advanced Postman can facilitate students learning processes in a real distance learning environment?

Establishment of Advanced Postman generated another project with a purpose to develop a moblog client for Java-enabled mobile devices. Two students at the Department of Computer Science, University of Joensuu, were given a task to develop a moblogging client that supports multiple blogging APIs. The design guidelines were set to aim for good usability and a wide range of platforms. The first version of the software was finished at the end of 2006. In the future, we intend to run comparative tests on Advanced Postman and the new client in order to find out the superior system.

A prediction tool based on data mining of ViSCoS data is currently under development. When the tool is finished, it can be used by instructors to predict possible drop outs and good performers. Other predictions such as pitfall predictions can possibly be added later. The results of predictions can be used to guide students better and to enhance course contents further. Students themselves can use the tool to predict their own future, thus

motivating them to work harder, if necessary. Adaptive learning can also be established with help of prediction models.

As input method of mobile devices greatly affect the usability of mobile applications, an effort must be put on developing alternative input mechanism. In November 2006 three students at the Department of Computer Science, University of Joensuu, were given a project to port Dasher input software to mobile devices. Should this project succeed, it would be fairly easy to attach Dasher to motion or tilt sensors in order to input text by merely moving the mobile device. The project is expected to end by June 2007 and the results will be used to solve the text input challenge in ViSCoS Mobile.

In the future we will concentrate on other development threads of ViSCoS Mobile. Theories of content adaptation work well on paper, but we must design and implement efficient content adaptation engine that is able to effectively process the course material used in ViSCoS. Programming on mobile devices is also an issue and we must overcome incapabilities of mobile devices. As mobile technologies evolve and new innovations appear, more interesting tools are available to be utilised in projects like ViSCoS Mobile. We see ViSCoS Mobile as a promising concept for developing countries where physical cable connections to Internet are rare, but mobile technology is advancing fast. As a conclusion we can say that the future of ViSCoS Mobile looks bright, but much work is still required.

References

[1] R. Ackland. Mapping the U.S. Political Blogosphere: Are Conservative Bloggers More Prominent? In *Proceedings of BlogTalk DownUnder*, 2005.

[2] E. Adar. Usage Patterns for Cameraphone Driven Moblogs. Technical report, HP Labs, 2004.

[3] T. Anderson. Social Software Applications in Formal Online Education. In *Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies*, pages 909– 911, 2006.

[4] J. Attewell. Mobile Technologies and Learning: A Technology Update and M-Learning Project Summary. http://www.lsda.org.uk/files/PDF/041923RS.pdf, 2005. (Retrieved October 23, 2006).

[5] R. Beale. Mobile Blogging: Supporting Informal Mobile Learning. In *Proceedings of MLEARN 2005*, 2005.

[6] M. Berth. Informal Learning with Mobile Devices - Moblogging as Learning Resource? In *Proceedings of The Informal Learning and Digital Media: Constructions, Contexts, Consequences*, September 21-23, 2006.

[7] J. D. Betts and S. J. Glogoff. Instructional Models for Using Weblogs In eLearning: A Case Study from a Virtual and Hybrid Course. In *Proceedings of Syllabus 2004 Conference*, San Francisco, July, 2004.

[8] HigherEd BlogCon. HigherEd BlogCon Website. http://www.higheredblogcon.com/. (Retrieved October 21, 2006).

[9] Y.-M. Cheng, W. Yu, and T.-C. Chou. Life Is Sharable: Blogging Life Experience with RFID Embedded Mobile Phones. In *MobileHCI '05: Proceedings of the 7th International Conference on Human Computer Interaction with Mobile Devices & Services*, pages 295–298, New York, NY, USA, 2005. ACM Press.

[10] T. Chung. Mobile Learning: Acquisition of Chinese Culture, Moral Values, and Language via Moblogs. In *Proceedings of The International Conference on Language, Communication and Culture*, October 19-21, 2005.

[11] T. Chung. Mobile Learning: Acquisition of Foundation Chinese Literacy Skills via Moblogs. *International Journal of Learning*, 12(8):1–10, 2005.

[12] G. M. Copas and L. Elder. Comparing Data Input Methods on Handheld Computers. *Usability News*, 6(2), 2004.

[13] Instructional Technology Council. ICT's Definition for Distance Learning. http://www.itcnetwork.org/definition.htm. (Retrieved October 21, 2006).

[14] M. Dickey. The Impact of Web-Logs (Blogs) on Student Perceptions of Isolation and Alienation in a Web-Based Distance-Learning Environment. *Open Learning*, 19(3):279–291, 2004.

[15] M. Dougiamas and P.C. Taylor. Moodle: Using Learning Communities to Create an Open Source Course Management System. In *Proceedings of the EDMEDIA 2003 Conference*, Honolulu, Hawaii, 2003.

[16] S. Downes. Educational Blogging. *Educause Review*, 39(5):14–26, 2004.

[17] D. W. Drezner and H. Farrell. The Power and Politics of Blogs. In *Proceedings of the Annual Meeting of American Political Science Association*, 2004.

[18] H. S. Du and C. Wagner. Learning with Weblogs: An empirical Investigation. In *Proceedings of the 38th Hawaii International Conference on System Sciences*, pages 7b–7b, 2005.

[19] L. Efimova. Blogs: The Stickiness Factor. In Thomas N. Burg, editor, *BlogTalks: First European Conference on Weblogs*, pages 109–125, 2003.

[20] L. Efimova, S. Fiedler, C. Verwijs, and A. Boyd. Legitimised Theft: Distributed Apprenticeship in Weblog Networks. In *Proceedings of I-KNOW '04*, Graz, Austria, June 30 - July 2, 2004.

[21] P. J. Eronen. Mobile Learning Concretized - Five Pedagogical Models. Bachelor's thesis, University of Joensuu, 2006.

[22] H. Farrell. The Street Finds Its Own Use for Things. http://crookedtimber.org/2003/09/15/the-street-finds-its-own-use-for-things/, September 2003. (Retrieved September 15, 2006).

[23] R. Fischer. Microlearning with Mobile Weblogs. In Theo Hug, Martin Lindner, and Peter A. Bruck, editors, *Proceedings of Microlearning*, 2005.

[24] Wikipedia Foundation. Wikipedia - The free encyclopedia. http://www.wikipedia.org. (Retrieved October 21, 2006).

[25] P. Gerdt, P. Kommers, C.-K. Looi, and E. Sutinen. Woven Stories as a Cognitive Tool. In *Cognitive Technology*, pages 233–247, 2001.

[26] G. Goggin. "Have Fun and Change The World": Moblogging, Mobile Phone Culture and The Internet. In *Proceedings of BlogTalk Downunder*, May 20-21, 2005.

[27] T. Gross and M. Kleppe. FrameDrops: A Mobile VideoBlog for Workgroups and Virtual Communities. In *GROUP '05: Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work*, pages 128–131, New York, NY, USA, 2005. ACM Press.

[28] A. Haataja, J. Suhonen, and E. Sutinen. How to Learn Introductory Programming Over Web. In *Proceedings of the 7th International Conference of European University Information Systems (Eunis 2001)*, pages 309–312, Berlin, Germany, March 28-30, 2001.

[29] W. Hämäläinen. Problem-Based Learning of Theoretical Computer Science. In *Koli Calling - Proceedings of the Third Annual Baltic Conference on Computer Science Education*, pages 57–65, Koli, Finland, October 3-5, 2003.

[30] W. Hämäläinen, T. H. Laine, and E. Sutinen. Data Mining in Personalizing Distance Education Courses. In C. Romero and S. Ventura, editors, *Data Mining in E-Learning*, pages 157–171. WIT Press, Southhampton, Boston, 2006.

[31] W. Hämäläinen and M. Vinni. Comparison of machine learning methods for intelligent tutoring systems. In *Proceedings of the 8th International Conference on Intelligent Tutoring Systems*, volume 4053 of *Lecture Notes in Computer Science*, pages 525–534. Springer-Verlag, 2006.

[32] D. L. Holbrook. The Master List of Blog Articles. http://www.ourboldhero.com/research/masterlist.html. (Retrieved October 14, 2006).

[33] D. Huffacker. The Educated Blogger: Using Weblogs to Promote Literacy in The Classroom. *AACE Journal*, 13(2):91–98, 2005.

[34] I. Jetsu, V. Meisalo, N. Myller, and E. Sutinen. Ethical Argumentation with Ethicsar. In *Proceedings of the IADIS International Conference on Web Based Communities*, pages 255–261, Lisbon, Portugal, March 24-26, 2004.

[35] M. Jones, G. Marsden, N. Mohd-Nasir, K. Boone, and G. Buchanan. Improving Web Interaction on Small Displays. In *Proceedings of the 8th International World Wide Web Conference*, Toronto, Canada, May 11-14, 1999.

[36] Kinshuk and T.-T. Goh. Mobile Adaptation with Multiple Representation Approach as Educational Pedagogy. In *Wirtschaftsinformatik*, pages 747–764, 2003.

[37] Kinshuk, J. Suhonen, and E. Sutinen. Mobile Adaptation Extension of a Digital Learning Tool. In *Proceedings of the International Conference on Computers in Education (ICCE2004)*, Melbourne, Australia, November 30-December 3, 2004.

[38] M. L. Koole. Practical Issues in Mobile Education. In *Proceedings of The 4th International Workshop on Wireless, Mobile and Ubiquitous Technologies in Education*, November 16-17, 2006.

[39] S. D. Krause. When Blogging Goes Bad: A Cautionary Tale About Blogs, Email Lists, Discussion, and Interaction. *Kairos*, 9(1), July 2004.

[40] S Krishnamurthy. The Multidimensionality of Blog Conversations: The Virtual Enactment of September 11. In *AOIR Internet Research 3.0: Net/Work/Theory*, 2002.

[41] T. H. Laine, N. Myller, and J. Suhonen. ViSCoS Mobile: Learning Computer Science on the Road. In *Proceedings of The 5th Koli Calling Conference*, pages 143–148, Koli, Finland, November 17-20, 2005.

[42] S. Lindblom-Ylänne and A. Nevgi. *Yliopisto- ja Korkeakouluopettajan Käsikirja*. WSOY, Vantaa, 2003.

[43] J. Majava. Kohti Avointa Oppimista: Henkilõkohtaiset Weblogit Opetuksessa. *Piirtoheitin*, 2, 2005.

[44] M. Mayorga-Toledano and A. Fernandez-Morales. Learning with Mobile Devices - Research and Development. In J. Attewell and C. Savill-Smith, editors, *Learning Tools for Java-Enabled Phones: An Application for Actuarial Studies*, pages 95–98. Learning and Skills Development Agency, 2004.

[45] Sun Microsystems. The Java ME Device Table. http://developers.sun.com/techtopics/mobility/device/device. (Retrieved October 21, 2006).

[46] A. Moreno, N. Myller, E.Sutinen, and M. Ben-Ari. Visualizing Programs with Jeliot 3. In *Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI2004)*, pages 373–376, Gallipolli, Italy, May 25-28 2004. ACM Press, New York.

[47] T. Mortensen. Bibliography on Blog Literature. http://tilsett.hivolda.no/tm/draftbibliography.htm. (Retrieved October 10, 2006).

[48] T. Mortensen and J. Walker. Blogging Thoughts: Personal Publication as an Online Research Tool. In Andrew Morrison, editor, *Researching ICTs in Context*, pages 249–279. Intermedia, University of Oslo, April 2002. http://www.intermedia.uio.no/konferanser/skikt-02/docs/Researching_ICTs_in_context.pdf.

[49] N. Myller and J. Nuutinen. JeCo: Combining Program Visualization and Story Weaving. *Informatics in Education*, 5:255–263, 2006.

[50] L. Nagel, J. Suhonen, and E. Sutinen. Towards a ViSCoS Community: Analysis of an Online Computer Science Programme. *Advanced Technology for Learning*, 3(4), 2006.

[51] B. A. Nardi, D. J. Schiano, and M. Gumbrecht. Blogging as Social Activity, or, Would You Let 900 Million People Read Your Diary? In *CSCW '04: Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*, pages 222–231. ACM Press, 2004.

[52] Educational Blogger Network. ENB @ BAWP. http://www.ebn.weblogger.com/. (Retrieved October 21, 2006).

[53] The NewPR. NewPR Wiki - Resources.CorporateBlogsList. http://www.thenewpr.com/wiki/pmwiki.php?pagename=Resources.CorporateBlogsList. (Retrieved October 21, 2006).

[54] NITLE. NITLE Weblog Cencus. http://www.blogcensus.net. (Retrieved October 21, 2006).

[55] Nokia. Nokia Lifeblog. http://www.nokia.com/lifeblog. (Retrieved September 18, 2006).

[56] The Association of British Correspondence Colleges. ABCC - Study at Home Website. http://www.homestudy.org.uk/. (Retrieved October 21, 2006).

[57] D. Perry. Handheld Computers (PDAs) in Schools. Technical report, British Educational Communications and Technology Agency (BECTA), 2003.

[58] K. Placing, M.-H. Ward, M. Peat, and P. T. Teixeira. Blogging in Science and Science Education. In *UniServer Science - Proceedings of the Blended Learning in Science Teaching and Learning*, 2005.

[59] M. Prensky. What Can You Learn from a Cell Phone? Almost Anything! *Journal of Online Education*, 1(5), June/July, 2005.

[60] I. Ramos. Political Blogs and Participatory Democracy: Evidence from Portugal and Spain. *Technology, Humanities, Education, & Narrative (THEN)*, 3, 2006.

[61] W. Richardson. Weblogg-ed Web Site. http://www.weblogg-ed.com/. (Retrieved October 21, 2006).

[62] W. Richardson. Weblogs in the English Classroom: More Than Just Chat. *English Journal*, 93(1):39–43, September 2003.

[63] W. Richardson. The Blogging in Schools Question. http://weblogg-ed.com/2004/the-blogging-in-schools-question/, April 2004. (Retrieved October 21, 2006).

[64] J. Rohan, S. Ju, and R. Sandler. Could MySpace Be Worth $15 Billion in Three Years? http://www.paidcontent.org/spacegasm.pdf, September 2006. (Retrieved October 13, 2006).

[65] T. Smith. Personal Assistants (PDAs) in Further and Higher Education. Technical report, Joint information Systems Committee (JISC), 2003.

[66] Hansard Society. Political Blogs - Craze or Convention? http://www.hansardsociety.org.uk/assets/Final_Blog_Report_.pdf, July 2004. (Retrieved October 23, 2006).

[67] J. Suhonen. *A Formative Development Method for Digital Learning Environments in Sparse Learning Communities*. PhD thesis, University of Joensuu, 2005.

[68] E. Sutinen and S. Torvinen. The Candle Scheme for Creating an On-Line Computer Science Program - Experiences and Vision. *Informatics in Education*, 2:93–102, 2003.

[69] Technorati. State of the Blogosphere, April 2006 Part 1: On Blogosphere Growth. http://technorati.com/weblog/2006/04/96.html, April 2006. (Retrieved October 21, 2006).

[70] M. Tepper. The Rise of Social Software. *netWorker*, 7(3):18–23, 2003.

[71] P. Trafford. Mobile Blogs, Personal Reflections and Learning Environments. *Ardiane*, 48, July 2005.

[72] M. Vanhanen. Matti Vanhanen - Tasavallan Mittainen. http://www.mattivanhanen.net/. (Retrieved October 21, 2006).

[73] H.-C. Wang, Y.-S. Deng, and S. Chiu. Beyond Photoblogging: New Directions of Mobile Communication. In *MobileHCI '05: Proceedings of the 7th international Conference on Human Computer Interaction with Mobile Devices & Services*, pages 341–342, New York, NY, USA, 2005. ACM Press.

[74] D. J. Ward, A. F. Blackwell, and D. J. C. MacKay. Dasher - a Data Entry Interface Using Continuous Gestures and Language Models. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, 2000.

[75] J. Weight. "Faster, neater, sharper!": How Different Models of Communication Intersect. In *Proceedings of BlogTalk Downunder*, May 11-22, 2005.

[76] B. Williams and J. Jacobs. Exploring the Use of Blogs as Learning Spaces in the Higher Education Sector. *Australasian Journal of Educational Technology*, 20(2):232–247, 2004.

[77] M Willner. AlphaGrip White Paper: Handheld High Speed Text Entry Will Power True "Anywhere Computing". http://www.alphagrips.com/WhitePaper.html. (Retrieved October 21, 2006).