Cultural applications for mobile devices: Issues and requirements for authoring tools and development platforms

Daphne Economou *d.economou@ct.aegean.gr*
Damianos Gavalas *dgavalas@aegean.gr*
Michael Kenteris *mkederis@otenet.gr*
George E. Tsekouras *gtsek@ct.aegean.gr*

Cultural Heritage Management Lab (CHMLab)
Department of Cultural Technology and Communication, University of the Aegean
5 Sapfous, Mytilene, Lesvos, GR - 81100, Greece

This paper explores requirements that authoring tools and development platforms should satisfy for the development of cultural applications tailored for deployment on Personal Digital Assistants (PDAs) and mobile phones. To effectively determine such requirements the paper reviews the use of mobile technologies in the context of cultural organizations and tourism and examines three 'real world' case studies that focus on the use of PDAs and mobile phones for providing cultural and tourist information, keeping the visitors' interest and attention, as well as promoting various cultural organizations and tourist facilities. This approach allows the extraction of a set of PDA and mobile phone application requirements, the implementation of which is based on the apparatus offered by authoring tools and development platforms. The paper reviews and evaluates the design and development facilities provided by state-of-the-art multimedia application development tools for PDAs and mobile phones: Macromedia Flash Lite, Navipocket, Java 2 Micro Edition and Microsoft .Net platform for the Mobile Web. The paper concludes with a set of recommendations related to the way authoring tools and development platforms should be exploited in order to gratify application and designer needs for developing cultural and tourist applications.

I. Introduction

There has been a considerable amount of research on the use of multimedia technologies in the fields of cultural organizations and tourism for the provision of cultural interpretational information [15], [25], [30], [34], [35]. Mobile devices have also been gaining increasing acceptance as a means to provision cultural multimedia applications due to their physical characteristics and suitability in these fields. This is evident by a number of research prototypes and commercial projects that have been reported in literature [3], [5], [10], [15], [26], [30], [31]. Currently, the tools used to develop multimedia applications for mobile devices are light versions of state-of-the-art multimedia authoring tools, which are not tailored to adequately satisfy user, designer and mobile device applications requirements.

This type of applications for PDAs and mobile phones are distinct from other domains in respect of the type of content to be provided, the way this content is made available to the user and the way these types of applications are developed. Cultural and tourist applications require rich multimedia content in highly interactive interface, customized for a wide range of devices. To minimize user hesitance and enable affordable usage, the access to content and services should not require constant network connection. Also, cultural applications pose a requirement for multi-modal media provision (e.g. graphics, audio, video) which are not always applicable to other application fields of mobile computing (e.g. mobile commerce, calendars, news, etc); some of these requirements though, may be projected to other fields, such as mobile education. The aim of this paper is to identify a coherent set of requirements that authoring tools and application development platforms should satisfy in order to allow the effective, efficient and economical development of cultural applications on PDAs and mobile phones. This exercise builds upon the evaluation of current state-of-the-art authoring tools and on experience gained by using such tools in three case studies developed in our laboratory to gather user and application requirements that arise in different application states of affairs. The development process of multimedia applications comprises of tasks for: multimedia content creation,

or optimization, which is a particularly time consuming process, interface design, interaction design and service development or provision. The study deals with two types of stakeholders, the cultural application "end user" and the "cultural application developer". The former determines the mobile cultural application needs, while the latter specifies authoring tools and application development platform requirements.

The first two case studies focus on the use of handheld devices for the provision of interpretative cultural information in museum environments, whilst the third is a mobile tourist guide for the city of Mytilene, Greece. The case studies have been developed using Navipocket and Flash Lite authoring tools and J2ME development platform, respectively.

The remainder of the paper is structured as follows. Section II reviews projects related to the use of mobile devices in cultural organizations and tourism. Section III discusses issues related to the design of PDA and mobile phone cultural and tourist applications. Section IV describes the main features of typical tools for multimedia application development for PDAs and mobile phones. Section V presents the three case studies and Section VII identifies a set of requirements for cultural application authoring tools and development platforms. Finally, Section VIII draws conclusions about this work.

II. A review of mobile devices use in Cultural organizations and Tourism

II.A. The use of mobile devices in Cultural organizations

Multimedia and new technologies provide unique opportunities to museums as they bring new ways of communication and interpretation. Technological solutions known by now, like projection systems and info-kiosks, successfully connect artefacts to information. However, these solutions are limited to the museum physical space. Audio-guides are successful examples of connecting artefacts to information in a portable way, which is not limited by the physical environment, but it is restricted to only using audio.

PDAs technology allows the dynamic presentation of information, without being limited or encroached by the aesthetics of the galleries. The introduction of handheld computers in museums for enhancing visitor experience was inspired by Zaurus, Psion, and Newton [2], that provided the museum visitor the opportunity to access multimedia, text and audio while walking through the exhibitions. Moreover, in contrast to audio-guides, users could now follow a non-linear itinerary exploring the information provided.

In 1997 the Smithsonian Institute launched a traveling exhibition named "America's Smithsonian", which used handheld computers iGo to lead an interactive tour providing information on 90 objects of the exhibition using narration and text graphics. In 1999 the Smithsonian Institute included the handheld computer Rocket e-Book at the "On Time" exhibition, which supported hypertext files, sound and B&W graphics. Museum futurists showed great interest in wireless networks and location-aware technologies. The project Hyper Interaction with Physical Space (HIPS) was one of the first projects that experimented with mobile computing and location awareness in museums [15]. The Hippie/Hips project was based on the use of infrared beacons strategically installed to track location of visitors of an exhibition [28]. The visitor choices were traced in the form of a user profile, from which more exhibition suggestions were recommended. Also, the MIT lab is known for its interest in wearable computing through the work of Sparacino in interactive exhibit design [31]. Tate Gallery gives emphasis on museum interpretation. Tate Modern Gallery launched two pilot multimedia tours whose mission was to "to test both applications of wireless technology in the gallery, and to access a wide range of approaches to content design" [30]. The PEACH project was one of the first museum guide projects to embed mobile devices into the existing infrastructure of the museum [36]. Through this research, the term "intelligent museum" surfaced denoting different devices (mobile or fixed) that interact with a variety of services provided in a user-adapted and context-adapted way.

PDAs afford to be used as guides that provide enriched information in outdoor archaeological sites. PDAs on site fill the gap left by conventional paper guidebooks, info kiosks and audio guides and provide truly mobile devices with navigation, personalization, and interactivity features [14]. Archeoguide [33] is an example project based on advanced mobile IT to provide new approaches for accessing information at cultural heritage sites.

Although the audience's feedback was positive towards the use of PDAs in exhibitions, the museums did not incorporate them because of

practical concerns. Issues included: the fragility of the devices; the frequent need to recharge batteries; the current high cost of the devices. However, the interest for enhancing the museum experience by providing information and interpretation, as well as promoting various museum facilities with the use of handheld technologies has expanded lately and there has been a considerable amount of research activity on this subject.

II.B. The use of mobile devices in Tourism

Tourism is a worldwide industry which involves the propagation of large amounts of information. As in most industries, Information Technology has penetrated the field of tourism for the manipulation of such information. The convergence of IT and communications technologies and the rapid evolution of the Internet has been one of the most influential factors in tourism that change travellers' behavior.

At the same time, the rapid growth of mobile devices (mobile phones, palmtops, PDAs, etc) user base raises a demand for using such devices to access Internet resources. Wireless access through mobile devices facilitates "portable" Internet connectivity, i.e. connectivity with no time or geographical constraints, by devices with high penetration to the public; tourists are amongst this technology-oriented public (for instance, the official online tourist office of Dublin [12] supports the provision of rich multimedia content, like podcast audio guides, to multiple mobile device platforms (including PDAs, WAP/i-mode - compatible terminals, i-pods, etc)). Hence, a growing body of commercial and research initiatives that incorporate electronic tourist guide functionality into mobile devices have been reported [23]. However these technologies in general have had a limited success; this is due to the lack of an in-depth study of the special characteristics of tourism, which can draw implications for the design of mobile tourist applications [7]. In the following paragraphs, we review several commercial and research projects with respect to mobile tourist guides. Client-server interactions and location-based services are the key features shared among all the reviewed projects.

The GUIDE system provides guidance, information and communication to tourists visiting Lancaster, UK by means of mobile devices [10]. The system integrates a cell-based wireless communications infrastructure based on a distributed and dynamic information model to locate users in which case derives context-aware adaptive hypermedia that is displayed via a customised web-browser interface.

Similarly, the LoL@ (LOcal Location assistant) project [3] was designed to offer terminal positioning, incorporating an automatic tour guide diary system and the provision of multimedia information related to sights of the city of Vienna. The Lol@ system was based on the next generation mobile telecommunication technology (UMTS Mobile Telecommunication system) and was developed using the Java 2 Micro Edition (J2ME) Connected Limited Device Configuration (CLDC) with the Mobile Information Device Profile (MIDP). It showed a concept to present city sight information with special regard to porting challenges to different mobile devices with different capabilities (e.g. small display size), frequent disconnection or low bandwidth. The REAL project was a hybrid (combination of client-server with application-based architecture) pedestrian navigation system, which helped the user to find information by generating a graphical route description [5]. Its location tracking system was based on GPS/compass signals outdoors and infrared beacons indoors. The REAL project uses the Pocket PC C/C++ development platform and incorporates the IBM Embedded ViaVoice format based speech synthesizer and dynamic rule grammar based recognizer. The 2D- and 3D-graphics are generated via the embedded Cortona VRML1-browser. The project uses GPS, which provides sensor information, such as velocity and direction, and is also required to locate the user when outdoors. Infrared beacons are used to locate a user when indoors. PDA communication with the server is via a standard HTTP connection, for example through a bluetooth capable GPRS/UMTS mobile phone, WLAN, or a USB desktop connection.

The Guide and the LoL@ projects rely on the client-server model to offer location based services via GPS. In situations of no network coverage these projects do offer limited information support to users. However, they were based on client-server technology, which offers multiple client support but relies on the network connection for service provision to those distributed clients. The TellMaris project enables a decentralized approach, which offers advantages in comparison to constant network connection, but represents a device/platform-dependent system resulting in limited device support. The REAL project was one of the first research projects to incorporate a hybrid system inheriting both advantages and the disadvantages of

the application-based and client-server technologies.

The choice of the type of architecture to use for implementing a tourist guide has a major effect on usability issues. Client-server architectures typically provide platform-independent services, however, they raise many concerns, e.g. connection cost, roaming agreements, loss of network connectivity, etc. On the other hand, standalone applications do not pose network connectivity requirement, yet, they may be platform-dependent resulting in complications at the deployment stage for multi-device systems let alone the non-dynamic creation of content, which is offered to the user. The myMytileneCity Guide (developed at our laboratory and presented in [22]) addresses these issues by allowing the user to select their content via a pc-based web interface and then allowing for the dynamically deployment of the created application across most available mobile devices platforms (including PDAs, smart phones and i-Pod units). Once installed, the application operates on both standalone and client-server based modes. Thus incorporating a pull model for content dynamically chosen by the user and having a push model for the creation of context aware content via a network connection.

The following section discusses issues related to the design of PDA and mobile phone applications, with emphasis on their role in cultural organizations and tourism.

III. PDA and mobile phone application requirements

The section initially expresses the general requirements regarding the design of PDA and mobile phone applications (see Section III.A) and then it focuses on specific application-related issues for cultural organizations and tourism (see Section III.B).

III.A. Issues related to the design of PDA and mobile phone applications

According to Dunlop and Brewster [13], for the design of successful PDA applications various factors should be considered mainly related to the device technical characteristics and the use of the application. These factors augment PDA application designers with new challenges, such as:

- mobile context, which can be defined as “any information that characterizes a situation related to the interaction between users, applications and the surrounding environment” [12]
- design for mobility
- design for multimodality (input with voice and touch with relevant spoken output and onscreen visual displays) and user multitasking at levels unfamiliar to most desktop users
- design for a wide user community with various skills and expertise
- design for limited input/output facilities, which means restrictive data entry methods and small screen size with different display resolutions
- limited processing capability and power
- design for providing information based on the user location

However, several issues differentiate the requirements for PDAs and mobile phone applications. The major difference between PDAs and mobile phones is currently screen size. For the design of mobile applications, one must consider the limited screen space the mobile phone has to offer [16]. More recent releases of high-end mobile devices have larger screen sizes, however their owners still represent a minority in the market. Designers of mobile phone applications have to consider three main categories of screen sizes when designing applications [16], unlike PDAs where standards in screen sizes do not vary considerably.

Another issue regarding designing applications for mobile devices is limitations of computational power and memory capacity. This is an immense problem particularly for multimedia applications that require fast processing speed and a large amount of memory for graphic support. Due to the limited processing capability of mobile devices, developers may have to disable some functions (e.g., high resolution images and dynamic frame movement). In addition, most available mobile phones are not yet fully multimedia enabled. The computational constraints of mobile phones have been addressed by the advent of the Ultra-Mobile PC (UMPC) [32], which offers consumers a small, thin and light ultra-mobile device with full PC capabilities, Internet access and anytime connectivity. UMPCs have enough processing power to support audio, video, and gaming, in addition to having rich support for browsing the internet as well as for other communication and networking applications. Ultra-Mobile PCs can also feature GPS devices, webcams, Bluetooth, Wi-Fi, etc. UMPCs are expected to be supported by a wider range of development tools as they do not hold the typical constraints of phones.

Furthermore, the number of dedicated mobile phone applications are still few. Thus, users can not easily draw on past experiences in the use of such

applications [13]. Unlike PDAs, most of the mobile phone applications are downloaded over-the-air (OTA), which implies lack of user manuals, long installation time and -possibly- high download cost. Mobile phone devices in general have less processing capabilities than PDAs. Hence, applications targeting mobile phones need to be highly optimized and customized to meet a broad range of user and devices requirements.

III.B. Issues related to the design of PDA and mobile phone applications for cultural organizations and tourism

The design of PDA applications for cultural organizations (e.g. museums) should address the organizations' requirements and provide the visitors a pleasant experience [11]. Woodruff et al. [35] studied the visitor behavior using PDAs in museums and they identified certain issues related to the user interaction with the PDA application:

- the visitors should be provided with visual feedback for their selection
- the information presented should be short and the system should support audio presentation of the information
- the audio information should not interfere with the interaction between the visitors in the museum

In terms of the interface design, the PDA applications should pursue criteria similar to web sites development [11]. In addition, the Canadian Heritage Information Network adds some practical guidelines [8] for the graphic design of the interface:

- each screen node of the PDA application should fit the size of the PDA screen
- the navigation should be structured hierarchically
- backtrack and easy access to the home page should be supported

The design of an aesthetically pleasing interface is important, however, the success of the system is based on accessing information in an intuitive and easy way [29].

Requirements gathering studies performed at CHMLab in close collaboration with cultural organizations identified a list of design factors that need to be taken into consideration in order to develop effective and efficient cultural mobile applications that enhance the visiting experience [14]:

- provide a means of promoting the cultural content (cultural artifacts) and connecting it with related information (e.g. for an artist that might have created the exhibited artifacts, the techniques used to create the artifacts, the artist's inspirations or even information for parts of an exhibition that are presented in another museum, etc.) without bounding the user to a cultural organization's/exhibition's physical space
- apply multimedia techniques to provide interpretative information about a collection in order to be able to effectively assign meaning related to them
- guide the user by implying translucently a way that the user should follow to view an exhibition
- focus the visitors attention
- personalize the provided information to the visitors interests and experience
- promote services and receive visitors' feedback in order to improve services
- support pre- and post-visit interaction with the visitors
- support the formation of interest groups

As far as mobile phone applications are concerned, most of research work and prototypes are restricted to tourist applications. This is mainly due to the physical restrictions of this class of devices as explained in section III.A.

Kray and Baus in 2003 completed a survey of available tour guide applications [23]. Many issues were retrieved relating to mobile phone applications design:

- the use of location-based services (LBS) is recommended to take into account user and context-related information
- adaptation capabilities should be enabled (e.g. the system should adapt to the lack of network connectivity)
- the user interface should support multimodal communication, natural language and multilingualism.

IV. Authoring tools and application development platform

This section presents the development and design facilities provided by some typical tools for multimedia application development and services provision for PDAs and mobile phones, which are classified in:

- authoring tools
- application development platforms.

This classification has been determined by the objective to evaluate strengths and weaknesses characterizing authoring tools and application development platforms and to identify application requirements from the end user and the developer's

viewpoint. Authoring tools are widely valued by designers and developers as key for rapid development process. Authoring tools accelerate the interface design and the content creation process, which are very time consuming, especially in multimedia application development. However, authoring tools usually lack in facilities for the development of services or for linking applications to services. On the other hand, application development platforms sufficiently accommodate the latter needs, but require more time, skills and programming effort.

IV.A. Authoring tools

Authoring tools development for the implementation of multimedia applications for Cultural Heritage and tourism has been attracting a lot of commercial attention. There are a number of companies focusing on this area like Macromedia, OPHRYS SYSTEMS, Antenna Audio, ATS Heritage and others. In this section two authoring tools are reviewed, namely, Flash Lite by Macromedia and NaviPocket v. 2.4 by OPHRYS SYSTEMS. Both tools have been evaluated in case studies examined later on in the paper.

IV.A.1 Macromedia Flash Lite

Macromedia Flash Lite [1] is referred to in this section, although not used in the case studies examined later on in the paper, as one of the most commonly used multimedia authoring tools that enables companies to easily and rapidly deploy content to mobile devices. The explosive adoption of Flash Lite was driven by a variety of causes. The Flash Lite authoring environment provides the designers and developers a new level of expressiveness, efficiency and interactivity for content creation. In addition, the Flash Lite rendering engine (Flash Player SDK 7 to date) is optimized for consumer electronic devices, enabling consumer electronics manufacturers, system integrators and browser companies to create high impact products and services, with full web browsing capabilities. In addition, developers already skilled in working with Flash MX can easily switch into using Flash Lite to design applications for mobile devices.

IV.A.2 Navipocket

NaviPocket v. 2.4 by OPHRYS SYSTEMS has been designed to meet the demands of Theme and Leisure Parks, Museums and Cultural sectors in developing multimedia guides. Navipocket allows the creation of multimedia applications on electronic message minders of PDA type. It is a software unit aimed for portable systems (PDA or TabletPC-type) supporting an embedded OS (Version 1 functions under Microsoft Windows EC 2.xx and PocketPC). The current version works with Microsoft PocketPCTM 2002 and Windows Mobile 2003. A PalmOS version will be available soon. The product is a complete set of an “Editor”, a “Simulator” and a “Run-time”. Within the Editor Module, the user creates a set of pages. These pages are in text format and are built according to an object-oriented model. Navipocket supports the following objects: page, button, text area, bitmap and video. Each object has properties and can be linked with another object.

Authoring tools like Navipocket and Flash Lite accelerate the delivery of advanced applications and content services. However, they are not open source, they do not support dynamic content maintenance and they require MS Windows compatible devices for the development of multimedia projects, and for the run-time.

IV.B. Application development platforms

In this section two application development platforms for mobile devices are reviewed, namely, Java 2 Micro Edition by Sun Microsystems and the .Net platform for the Mobile Web by Microsoft. The former has been evaluated in a case study examined in Section V.C..

IV.B.1 Java 2 Micro Edition

Java 2 Micro Edition (J2ME) [18], released by Sun Microsystems, is a Java-based framework for developing applications executed on resource-constrained devices. J2ME has achieved a remarkable penetration and is currently supported by virtually all mobile devices. J2ME applications are called MIDlets; MIDlets are usually packaged in *.jar files, downloaded on-the-fly from a web server and executed as standalone applications with no requirement for constant connection to a wireless network.

J2ME inherits the main assets of Java language, i.e. the capacity to develop powerful applications, platform independence, etc. Hence, developers are not restricted by the limitations of an authoring tool’s functionality and may implement full-fledged innovative applications that either execute standalone or communicate with their peers or service providers, taking advantage of the J2ME’s strong wireless networking support.

On the other hand, MIDlets programming is not straightforward as it requires Java development

skills. The development of J2ME applications is far more complex compared to creating content using developer-friendly authoring tools like Flash Lite or Navipocket.

IV.B.2 Microsoft .Net platform for the Mobile Web and the ASP.NET mobile controls

Microsoft's entering the mobile market has been characterized by the release of a proprietary operating system, namely, Microsoft Windows Mobile 2003 [27] and the provision of developer support to program mobile devices. Specifically, a subset of the rich .NET Framework, called Microsoft .NET Compact Framework, provides a runtime engine preloaded in the device's memory in order to facilitate mobile application deployment.

The ASP.NET mobile controls [4] represent a mobile application development platform, recently released by Microsoft. In particular, the ASP.NET mobile controls provide an easy way to build mobile web applications that generate the appropriate markup language (WML, xHTML, HTML or cHTML) and rendering for web-enabled cell phones, WAP phones, PDAs, Pocket PCs and pagers. The programming of ASP.NET mobile controls is enabled by the Mobile Internet Toolkit (MIT) development environment. The main asset of MIT is that it provides server-side mobile controls (including user interface elements such as list, command, call, calendar, etc.) with rich device identification mechanisms; developers simply utilize ASP.NET pages (for no particular target device) which automatically identify the device that posted a request1 and render the appropriate content.

Summarizing, the main strengths of ASP.NET mobile controls are: no need to perform browser checks and deliver the appropriate content based on the target device (this makes an application faster to develop and easier to maintain); developers only need to learn ASP.NET and .NET mobile controls (no need for markup language authoring skills); easy to use programming model and drag-and-drop application development with Visual Studio.NET.

In contrast, the main limitations of this technology are: the target devices are limited to Microsoft products and operating systems (unlike the J2ME platform-independent applications); when a new version of WML or HTML is released, developers need to wait until Microsoft announces support for the new version within its .NET mobile controls.

[1] Accurate information about the display capabilities of the target device is essential for the successful rendering of mobile controls. At a minimum, mobile controls need the following information about a device: markup language (HTML, WML, cHTML), browser, number of display lines, cookie support, screen size.

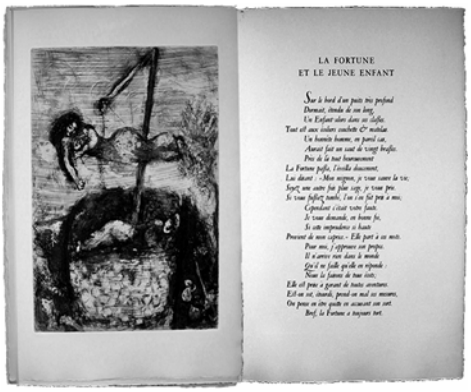
V. Case studies

In this section three case studies are presented. The first evaluates Navipocket as an authoring tool for the development of a cultural multimedia application on a PDA, the second describes a Flash Lite-based news reader mobile phone application and the third evaluates J2ME as a development platform for the implementation of an electronic tourist guide executing on mobile phones.

V.A. The use of a PDA to provide interpretative information for a set of illustrations

In this section, we present a PDA case study, the "Fables on pocket PC". This project focuses on the study of real requirements for the development of museum multimedia applications for PDAs that enhance visitor's experience and provide information and interpretations about the museum collections. The project uses the Museum/Library Stratis Eleftheriadis Teriade in Lesvos, Greece, as a case study and in particular the collection of "Fables" (the Fables were inspired by the tales of Greek, Hindu, Persian, Arabic and Chinese writers) by Jean de La Fontaine, which have been illustrated by Marc Chagall and are exhibited in part of the Teriade Museum. The Teriade Museum exhibits the editorial work of Stratis Eleutheriadis Teriade, which includes copies of the "Grands Livres" and "Verve" that have been illustrated by some of the greatest artists of the 20th century like Chagall, Matisse, Picasso, Braque and others, as well as copies of medieval manuscripts.

In its galleries Teriade Museum exhibits the only illustrations of the "Grant Livres", the "Verve" and medieval manuscripts (see Figure 1b). The illustrations are disconnected from the story they supplement (see Figure 1a). This makes it difficult for the visitor to understand the collection and appreciate its importance. Then again the Teriade Museum appreciates the visitors' requirements of being provided with more information about the collection, and connecting the illustrations to the stories, however, this should not superimpose the physical site.



(a)

(b)

Figure 1. (a) A snapshot of two pages of the "Fables" that contains the story and the illustration by Marc Chagall. (b) One of the exhibition walls of the "Fables" collection at the Teriade Museum exactly the way that Teriade set it up. The stories are disconnected from their illustrations and no interpretational information about the collection is presented.

The use of PDAs technology suits adequately the Teriade Museum requirements, as it offers a very rich way of:

- connecting the stories to the illustrations,
- providing general and interpretative information about the Fables' illustrations that allows the visitor to understand the collection,
- helping visitors to focus their attention on specific items of the collection,
- at the same time allowing the visitors to create their own personal opinion about the collection

without encroaching on the aesthetics of the gallery space. The application helped the users to orientate themselves in the gallery by including videos with instructions about the layout of the exhibition and the location of the collections and specific objects of the collections. In addition, the system allows the visitors to select content (text and images) that meets their personal interest and take it with them after their visit at the museum in print (as a kind of souvenir). These design solutions match most of the cultural organizations requirements as shown in Section III.B.

The content included in the application was especially selected to reflect information of value to certain stakeholders. This set of stakeholders consists of: the Teriade Museum curators, museum management, art historians who might provide various interpretations for the collection and visitors. This stakeholder-oriented method of content requirements analysis evolved out of the Soft Systems approach [9].

For the design of the application's graphical user interface (GUI) and navigation system various issues of usability have been taken into consideration [20] in terms of:

- the physical interaction: menus have been used to select sections of information, control buttons have been used to control the videos (play, pause, rewind), while direct selection of illustrations has been used in order to get related information
- the organization of the information in sections and subsections
- the layout, colour and screen fonts to aid visual perception

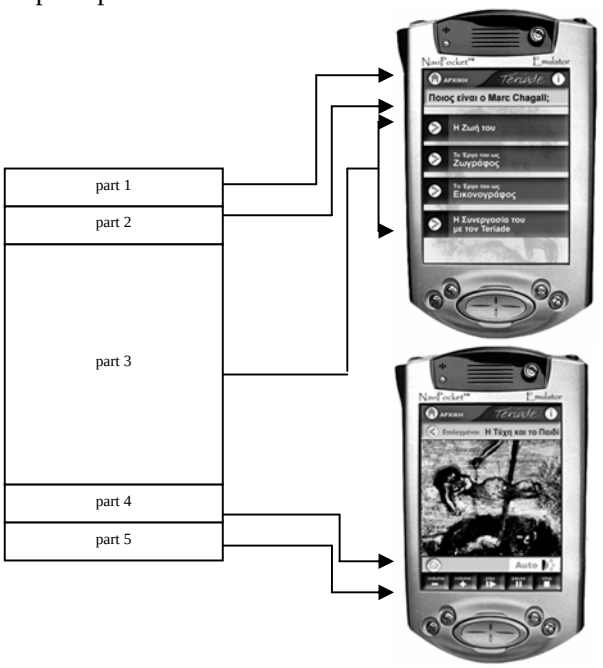


Figure 2. The layout of the "Fables" prototype application.

The application has been designed for mobility by providing an interface with simple menus (see Figure 2) and concise information so that interaction with the application requires minimal effort and does not distract the user's attention from other activities (moving in the exhibition, talking with the etc.). The user can navigate through the content by choosing manually the sections of interest with a PDA stylus. Alternatively, the content can be automatically

selected by the software, since it makes use of infrared based technology which determines the visitors' position in the gallery. The user can choose to switch between manual and auto guidance at any time while using the system. Auto guidance helps the user orientation in the gallery. In addition, in order to avoid user distraction and to address the stakeholders' request for incorporating multimedia techniques to effectively provide interpretative information about a collection (see Section III.B.), videos made of narration and animated pictures were included in the application.

Text was not used as visitors cannot afford the time to read the text while moving through the gallery [34]. Furthermore, PDA devices are too small to provide long textual descriptions, something already recognized by others working with small devices in cultural context [30], therefore audio was chosen as the preferred medium. The videos which were included did not exceed 1 minute duration, since long descriptions are tiring for the user, as already acclaimed by Proctor and Tellis [30], and Woodruff et. al. [35], who have been investigating various ways of including multimedia content in cultural applications for small devices to provide information and focus visitors' attention.

The application interface consists of five parts as it appears in Figure 2:

- part 1 remains static in all pages and includes two buttons, the left one leads to the home page, and the right provides information for the Teriade Museum
- part 2 describes the current section the visitors view and includes a button that returns to the previous page
- part 3 includes either navigation buttons that lead to the various sections and subsections of the application, or a video related to the subsections selected
- part 4 includes buttons that allow switching between auto or manual guide
- part 5 includes control buttons for the video and audio

One hundred illustrations of the Fables were digitized and edited for the creation of the videos. For this prototype only 5 videos have been created. For the narration, a male and a female voice have been used, in order not to tire the visitor. Appropriate background sound has been incorporated into the videos. The videos have been created with the use of Flash MX 2004 (by Macromedia) and then imported as MPEG in NaviPocket v. 2.4 by OPHRYS SYSTEMS, which was used to implement the application (see Section IV.A.2.) The final application is 104 MB. These files are stored locally on the PDA storage or an SD card and executed by the Navipocket "Run-time".

The "Fables" prototype has been evaluated in site with real users [14]. The evaluation revealed various issues regarding information comprehension, interpretation and usability and showed that the application helped the visitors appreciate better the meaning of the gallery, improved the museum experience and served the user requirements effectively and efficiently. The use of multimedia elements like the video, the narration and the music provided a pleasant way of receiving information and helped focussing the visitors' attention.

The development process of the "Fables" prototype demonstrated several aspects regarding the Navipocket authoring tool functionality. In particular, Navipocket seems to offer a number of advantages:

- low purchase cost (the software is available free of charge) and application development cost since it involves rapid implantation with minimal effort
- low storage and memory overhead of the run-time environment
- rapid multimedia content and UI creation

However, several drawbacks of Navipocket have also been revealed:

- restrictiveness of the tools provided for the design of expressive UI
- typically large-sized applications, mainly due to the lack of support for a variety of multimedia formats
- dynamic content update is not supported
- dependency on specific operating systems and devices hardware
- lack of specialized libraries for implementing extra functionality (custom solutions may be supported by OPHRYS on demand).

V.B. The mobile phone used as museum guide and news reader

Presented here is a prototype mobile application used to promote museum services via a mobile phone. The natural history museum of the Petrified forest situated in Lesvos, Greece, was chosen as a case study. The original concept of this project, as it evolved from discussions with the museum's administration, was to build a lightweight, robust application to promote the museum including its location, descriptive content of its collections and a latest news section to mobile devices. Flash Lite was

chosen as the development platform, mainly because the developers were already skilled in working with Flash MX Professional and therefore could easily switch to Flash Lite. That reduced the production curve dramatically.

Flash allows the designer to produce rich content and customized user interfaces integrating interactivity and controlling certain behaviors via scripting tools. However, Flash Lite 1.1 and Flash Lite 2.2 do not offer support for location-based capabilities via GPS. Hence, at this stage, the museum case study does not incorporate any location-based services; this does not though represent a critical omission since this application is aimed at mass deployment for promotional use away from the museum and not for use as a guide system during a museum visit.

Several usability aspects have been taken into account in the design of user interfaces (see Section III.B.):

- no scroll bars are used since they were very difficult to handle on small screens;
- menu buttons are selected via navigational phone buttons and not via users softkeys; this gives the user an added feel of a multimedia presentation and not of a software application;
- the main functions buttons incorporate the same design for all navigational functions, e.g. moving between pages, clip gallery and video control.

The use of audio was limited in video clip in order to reduce the memory requirement. The overall screen size is 176×176, which is a standard screen size for most readily available mobile phones. When installed on devices with larger screen sizes the background color prevails throughout the screen. Figure 3 illustrates several screenshots of the application developed in Flash Lite.

Figure 3. Screenshots of the museum guide application taken from a mobile phone emulator.

As the project unfolded, the idea of reading the museum's latest news was thought of. The main aspect of the latest news section is to connect via HTTP to the already existing news feeder of the museum website to update the news section of the mobile application. Initially, there are two news feeds stored locally in the application, which may be updated through pressing a button. Once an HTTP connection is established, an XML file including the latest news is downloaded from the museum's web server and parsed from a flash object; the file contains three pieces of data: a headline, a date and a brief news description. The size of the application is directly related to the content dimensions. The final application size is 454 KB when incorporating images and a short video and 256 KB when omitting the clip gallery and the video.

V.C. The use of a mobile phone as a city tourist guide

In this section, we present a mobile tourist guide case study, the myMytileneCity guide (an electronic guide for the city of Mytilene, Lesvos Island, Greece) [22]. The prototype implementation is entirely based on Java, on both the web server and the client tier, in order to take advantage of its inherent platform-independence and suitability for web applications. Regarding the supported format of tourist content, XML-family technologies have been chosen to enable compatibility with web standards and interoperability (current trends promote XML-based languages for content development and data interchange on the Internet).

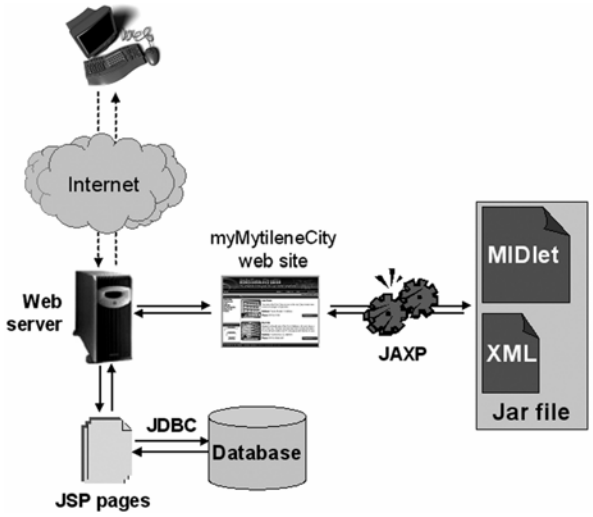


Figure 4. Dynamic generation of a J2ME application through remote interaction with myMytileneCity web site.

The design of the tourist guide application follows a two-step approach. Firstly, as illustrated in Figure 4, the user interested in a particular tourist destination visits a web site including information about restaurants, accommodation, sights, events,

night life, etc. The user selects the application's content using the same approach followed by e-commerce sites. That is, the user clicks on a "Add to content" button to append information of interest to his/her 'web suitcase' (abstraction of personal account) and the 'Remove from' to cancel a selection. The chosen content may be stored and retrieved upon a future visit. When the user 'checks out', the suitcase's content is transformed to XML format, using the powerful Java API for XML Processing (JAXP) [19]. Following that, the system automatically generates a jar file bundle that includes the MIDlet application (to be executed on the user's mobile phone) and the selected XML-based tourist content.

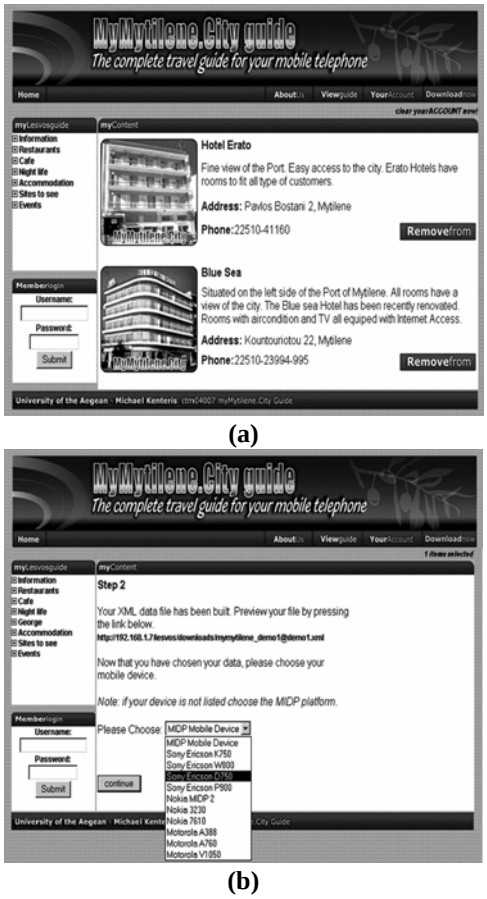
On the second step, the user downloads the generated jar file (temporarily saved on the web server) to their mobile device. Herein, the user is provided two options:

1. directly download the jar file to their mobile device (through end-to-end HTTP); depending on the jar file size this may be a time-consuming and costly solution as it engages the wireless channel for relatively long time (although downloading a J2ME application may be expensive, there exist commercial products that push J2ME applications to mobile devices through dedicated Bluetooth hotspots at no costs for the customer)
2. download the jar file in two phases: first, to a PC/laptop (through HTTP) and then to the mobile device (through Bluetooth, or infrared); although separated in two phases, this method is usually more cost-effective and fast as it takes advantage of the higher transfer rates of the wired Internet and the wireless Bluetooth protocol.

Upon completion of the jar file download to the J2ME-compatible device, the latter is installed and loaded by the local Application Management Software (AMS) module integrated within the J2ME platform (AMS runs on the same device as MIDlets and is typically provided by the device's manufacturer). The MIDlet application is thereafter executed in standalone mode with no wireless connectivity requirement (the user later synchronizes to the backend system only to update the originally selected tourist content). On the client tier, a user-friendly MIDlet menu allows easy browsing of selected content; the latter is included within the downloaded jar file (in XML format) and rendered for display by the 'lightweight' kXML parser [24].

myMytileneCity application has been designed taking into account several usability guidelines associated to three main characteristics of mobile applications, namely, small screens, limited input and mobility, as well as requirements gathering results performed in our labs (see Section III.B.):

- the interface is appealing to a wide range of users with various skills and expertise
- short and concise textual descriptions accompanied by pictures is used to provide tourist information (e.g. hotels, sites to see, etc.)
- the application's presentation follows a hierarchical multi-level structure that helps users to easily search for/browse/understand specific information of their interest.
- menus have been designed in order to help the user to easily reach the desired information. Menus and buttons are clearly labeled and consistent to help the user navigation, learnability and memorability. To minimize cognitive load, long lists of choices have been avoided and support has been added for backtracking and easy accessing of earlier pages/home page [6], [8].
- content page information is typically fitted on one screen to avoid scrolling.



(a)

(b)

Figure 5. Screenshots from the myMytileneCity web site: (a) contents of user's account (includes two lodging preferences); (b) selection of mobile device's profile when 'checking out'.

In addition to developing an interface which provides easy interaction and access to information, emphasis has been put on developing an aesthetically pleasing interface.

The following figures present various parts of the prototype usage. Figure 5 shows screenshots of the web site used by visitors to select tourist content of interest (to be later included as content items of the mobile application). Figure 6 demonstrates an excerpt of generated XML code, describing the users' selected content (this code corresponds to the user preferences shown in Figure 5a. Finally, Figure 7 illustrates representative screens of a mobile device emulator (main menu, list of lodging preferences, detailed description of a selected hotel).

The final application size is approximately 520 KB, including the CLDC/MIDP footprint, the kXML parser libraries and the MIDlet jar file (Java classes, XML-based tourist content and the accompanying graphics). This size is considered to be of acceptable volume for today's J2ME enabled devices and contrasts the huge application volume that was generated in Section V.A. using Navipocket, which was mainly attributed to the use of graphics and video files.

```
<?xml version="1.0" encoding="UTF-8"?>
<city name="Mytilene">
   <content id="2" type="Accommodation">
     <sub id="20" name="Hotels">
     <cname>Hotel Erato</cname>
     <cdescr> Fine view of the …</cdescr>
     <caddress> Pavlos Bostani … </caddress>
     <cphone>22510-41160</cphone>
     <cimage>erato.jpg</cimage>
     </sub>
   </content >
   <content id="2" type="Accommodation">
     <sub id="20" name="Hotels">
     <cname>Blue Sea</cname>
     <cdescr> Situated on the … </cdescr>
     <caddress>Kountouriotou 22… </caddress>
     <cphone> 22510-23994-995 </cphone>
     <cimage> bluesea.jpg </cimage>
     </sub>
   </content >
</city>
```

Figure 6. An excerpt of XML code describing the users' accommodation preferences.

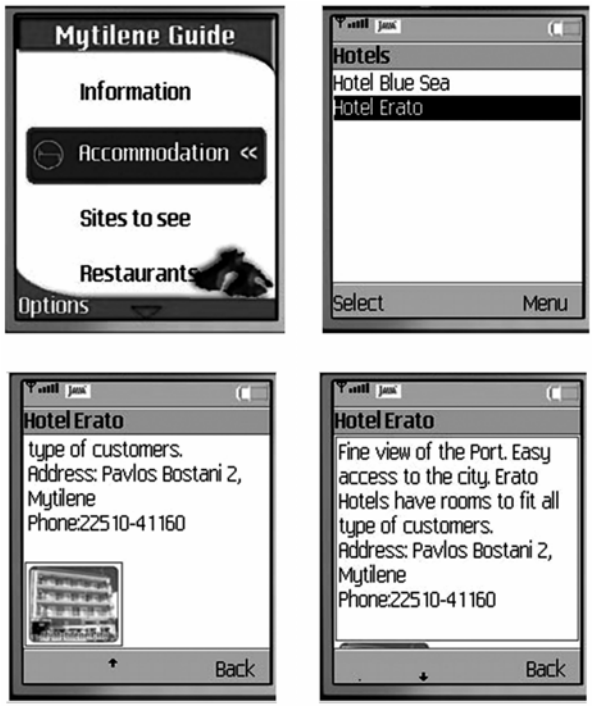


Figure 7. Screens of a mobile phone emulator executing the myMytileneCity guide application.

VI. Requirements for cultural application authoring tools and development platforms

It is evident, that the development tools reviewed in Section IV have different features and devices target groups. The set of requirements presented herein have been identified through the experiences gained during the implementation of the three case-studies, as well as from the compilation of the results of the usability tests performed upon them. These exercises indicated that the requirements identified in the paper are not unique to cultural applications, that is the results can be generalized to other types of applications for small devices. These requirements are listed below:

- rapidity of application's development and deployment (rapid development implies minimal implementation effort and cost)
- reduction of development effort and technical knowledge (e.g. programming skills) required by designers; familiarity of designers with the tool's workspace
- provision of tools for designers and developers that allow a new level of expressiveness, efficiency and interactivity for multimedia content creation and usable UI design, personalized according to the user profile (such

design could exceed customer expectations and optimize content delivery)
- support for a broad range of mobile devices (ideally, support for PDAs, smart phones and mobile phones)
- restriction on the resource overhead posed by the run-time environment (supporting libraries, APIs, etc.)
- seamless connectivity of applications to services with minimal programming effort
- platform independence of applications from underlying devices hardware and operating systems
- potential for developing entirely new content and services that overcome the restriction set by rigidly defined content templates
- capability for dynamic customization and over-the-air update of existing applications content and functionality
- increased deployment base of tools' runtime environments, i.e. management software and media players installed by the major device manufacturers
- minimization of cost for both the designer tools and the runtime environments
- support for location-based services, i.e. availability of resources and services depending on the end user's physical location
- support for 'push model', namely for pushing content to mobile terminals with minimal user intervention the moment an important event occurs
- support for disconnected operation, i.e. ability to run applications in standalone mode even when the mobile terminal is out of any network's coverage area
- need for large development community base, which may assist the exchange of development experiences (e.g. through developer forums)
- availability of add-on application libraries, which may accelerate the implementation of custom services.

Table 1 below summarizes the features of the available authoring and development tools (Flash Lite, Navipocket, J2ME and .Net platform for the Mobile Web) within respect to the above listed set of requirements.

The above synopsis shows that the choice of the appropriate development technology is not a straightforward task, since the four reviewed technologies vary significantly in terms of their merits and weaknesses. In particular, the selection of a candidate development technology should depend on user and application needs, such as:

- the technology literacy of developers and familiarity with relevant multimedia-based application environments
- the urgency of project completion
- the application requirements regarding network connectivity, dynamic updates, supported services
- the targeted devices
- the project's budget

In conclusion, in order to satisfy application and designer needs for developing operational and profitable cultural and tourist applications, future releases of authoring tools and development platforms should be directed in combining the strengths of the existing technologies.

VII. Conclusions

This paper reviews state-of-the-art technologies for developing mobile applications that enhance the visiting experience in cultural organizations and support tourists traveling experience. Based on this review it suggests authoring tools requirements for developing cultural applications on PDAs and mobile phones based on user, application and designer needs.

To support the requirements gathering process three case studies have been developed, which focused on the use of PDAs and mobile phones for providing cultural and tourist information and promoting cultural content and tourist facilities. These case studies were based on the use of the Navipocket authoring tool, Flash Lite and J2ME application development platform, respectively. The prototypes implementation contributed to the evaluation of the main advantages and shortcomings of such development technologies. The requirements stated in the paper are not unique to cultural applications, which means that the results can be generalised to other types of applications.

Specifically it has been concluded that the choice of the appropriate technology depends on factors like the developers technical background, the application requirements, the targeted devices and the project's timeline and budget.

Future development of authoring tools and development platforms should bring together the strengths of existing multimedia technologies in order to satisfy application and designer requirements for developing effective and beneficial cultural and tourist applications.

	Flash Lite	Navipocket	J2ME	.Net platform for the Mobile Web
Development and deployment speed	Relatively fast	Very fast	Slow	Relatively fast
Technical knowledge required	Users with no prior knowledge on Flash require fair training	Effortless	Advanced Java programming skills are required	.NET mobile controls programming skills required
Content development and UI design tools	Very advanced	Restrictive	Not integrated, requires design automation tools, e.g. J2ME Polish [17]	Very advanced, through the ASP.NET mobile controls
Targeted mobile devices	PDAs, smart phones, mobile phones	PDAs	PDAs, smart phones, mobile phones	Pocket PCs, PDAs, smart phones
Run-time environment resource overhead	~ 6 MB	~ 1 MB	Up to 100 KB for storage (CLDC/MIDP and kXML)	4,4 MB footprint for the .NET Compact Framework
Applications connectivity	Feasible, requires programming effort	Customized according to customer requirements	Feasible (via HTTP), needs programming effort	Feasible (via HTTP), needs programming effort
Platform independence	Mobile devices with Flash Lite or Flash Player SDK technology	Requires Windows Pocket PC; executed on PDA platforms manufactured by OPHRYS	Execution on any device supporting CLDC/MIDP	Targets devices with Microsoft OS
Accessible content format	Handles proprietary file formats in addition to either 'external' or integrated multimedia file formats	Handles proprietary file formats in addition to bitmaps and mpeg files	Any (text, XML, WML, cHTML, HTML, XHTML, etc.); requires specialized parsers for analyzing XML content	The appropriate content format (HTML, WML, cHTML) is generated depending on the target device
Potential for developing entirely new content and services	Development restricted by Flash Lite authoring environment	Development restricted by Navipocket Creator's functionality	Capacity to develop rich content and new powerful applications, inherited by Java language	Capacity to develop rich content and new powerful applications, inherited by .NET framework
Support for dynamic application update	Applications may synchronize with the backend infrastructure to dynamically update content	Not supported	Applications may synchronize with the backend infrastructure to dynamically update content	Applications may synchronize with the backend infrastructure to dynamically update content
Run-time environment's deployment base	Most major manufacturers	Not supported	Very large deployment base (virtually all modern mobile devices)	Not supported
Cost	~€10 for Flash Player, ~€700 for Flash Professional and	Free license given by OPHRYS; required to purchase the OPHRYS PDA	Free	Free license for the .NET Compact Framework, ~€700 for Visual Studio 2005 Professional
Support for location-based services	Not supported	Not supported	Yes (precise location identification though the optional 'Location API' [21])	Approximate location identification through mobile operators networks
Support for 'push model'	Not supported	Not supported	Yes (in MIDP 2.0)	Not supported
Support for disconnected operation	Yes	Yes	Yes	Yes
Developer community base	A starting community	Very limited	Large community of developers	Large community of developers

Table 1. Features of the reviewed authoring tools (Flash Lite, Navipocket) and development platforms (J2ME, .NET).

Acknowledgements

The authors would like to thank OPHRYS for the provision of Navipocket software for the implementation of the "Fables" prototype.

References

[1] Adobe: Mobile & Devices, last visited 30/10/2006

[2] S. Amirian, Hand-held Mobile Computing in Museums. A Background Paper, 2001.

[3] H. Anegg, H. Kunczier, E. Michmayr, G. Pospischil & M. Umlauft, LoL@, "Designing a Location Based UMTS Application. Elecktrotechnik und Informationstechnik", 191(2). pp. 48-51, 2002.

[4] ASP.NET Mobile Controls, 2006, http://www.asp.net/mobile/.

[5] J. Baus, C. Kray & A. Krueger, Visualization of Route Descriptions in as Resource-Adaptive Navigation Aid. *Cognitive Processing*, 2 (2-3). pp. 323-245, 2001.

[6] B. Bederson, A. Clamage, M.P. Czerwinski & G. Robertson, A Fisheye Calendar Interface for PDAs: Providing Overviews for Small Displays. *In Proceedings of the International Conference on Human Factors in Computing Systems (CHI'03)*, pp. 618-619, 2003.

[7] B. Brown & M. Chalmers, Tourism and Mobile Technology. *In Proceedings of the 8th European Conference on Computer Supported Cooperative Work (CSCW'03)*, pp. 335-354, 2003.

[8] Canadian Heritage Information Network (CHIN): Tip Sheets, Personal Digital Assistants (PDA), PDA Aesthetics and Interface Design, Creating and Managing Digital Content, 2004. Last visited in 10/05/2006: http://www.chin.gc.ca/English/Digital_Content/index.html

[9] P. Checkland & J. Scholes, *Soft Systems Methodology in Action.* Chichester-NY: John Wiley & Sons, 1990.

[10] K. Cheverst, N. Davies, K. Mitchell, A. Friday & C. Efstratiou, Developing a Context-Aware Electronic Tourist Guide: Some issues and Experiences. *In Proceedings of the 2000 Conference on Human Factors in Computing Systems (CHI'00)*. pp. 17-24, 2000.

[11] C. Ciavarella & F. Paterno, Design criteria for location-aware, indoor, PDA applications. *In Proceedings of Human-Computer Interaction with Mobile Devices and Services, 5th International Symposium Mobile (HCI 2003)*, Luca Chittaro (ed.). pp. 131-144, 2003.

[12] A.K. Dey, D. Salber & G.D. Abowd, A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. *Human-Computer Interaction*, 16, pp. 2-4, 2001.

[13] M. Dunlop & S. Brewster, The challenge of mobile devices for human computer interaction. *Personal and Ubiquitous Computing*, 6 (4). pp. 235 -236, 2002.

[14] D. Economou, S. Varitimiadis, M. Samaroudi, & D. Gavalas, The Effectiveness of Personal Digital Assistants (PDAs), Audio-Guides or Paper Leaflets to Enhance the Museum Visit Experience. *In Proceedings of the 3rd International Conference on Museology (ICM'2006)*, 2006.

[15] J. Evans & P. Sterry, Portable computers & interactive media: A new paradigm for interpreting museum collections. *In Cultural Heritage Informatics 1999: Selected papers from ICHIM 99*, D. Bearman and J. Trant (eds.). Kluwer Academic Publishers. pp. 93-101, 1999.

[16] K. Holtsblatt, Designing for mobile devices: Experiences, Challenges and Methods. *In Communications of the ACM*, 48(7). pp. 33-35, 2005.

[17] J2ME Polish. Last visit: 12/01/2008, http://www.j2mepolish.org/.

[18] Java 2 Platform Micro Edition (J2ME). Last visited in 12/01/2008: http://java.sun.com/j2me/.

[19] Java API for XML Processing (JAXP). Last visited in 12/01/2008: http://java.sun.com/webservices/jaxp/index.jsp.

[20] M. Jones & G. Marsden, *Mobile Interaction Design*, *Wiley*, 2005, ISBN: 978-0-470-09089-3.

[21] JSR 179: Location API for J2ME. Last visited in 12/01/2008: http://jcp.org/en/jsr/detail?id=179.

[22] M. Kenteris, D. Gavalas & D. Economou, An Innovative Mobile Electronic Tourist Guide

Application. *Personal and Ubiquitous Computing*, in press.

[23] C. Kray & J. Baus, A survey of mobile guides. *In Proceedings of HCI in Mobile Guides. In conjunction with Fifth International Symposium on Human computer Interaction with Mobile Devices and Services*. pp. 1-5, 2003.

[24] kXML parser. Last visited in 10/4/2006: http://kxmlrpc.objectweb.org/.

[25] D. Lehn & C. Heath, Displacing the object: Mobile technologies and interpretive resources. *In Proceedings of the ICHIM 03. Archives and Museum Informatics Europe, 2003*. Last visited in 12/01/2008: http://www.ichim.org/ichim03/PDF/088C.pdf.

[26] K. Micha & D. Economou, Using Personal Digital Assistants (PDAs) and to enhance the museum visit experience. *In Proceedings of the 10th Panhellenic Conference on Informatics*, LNCS 3746. pp. 188-198, 2005.

[27] Microsoft Windows Mobile (2006), http://www.microsoft.com/windowsmobile/.

[28] R. Oppermann & M. Specht, A Context-Sensitve Nomadic Exhibition Guide. *In Proceedings of the 2nd Symposium on Handheld and Ubiquitous Computing*, (HUC2K'2000). pp. 127-149, 2000.

[29] J. Preece, Y. Rogers & H. Sharp, *Interaction design: Beyond human – computer interaction*. Danvers, MA: Wiley, 2002.

[30] N. Proctor & C. Tellis, The state of the art in museum handhelds in 2003. *In Proceedings of the International Conference of Museums and the Web 2003*. D. Bearman and J. Trant, (eds.). Archives & Museum Informatics, 2003. Last visited in 12/01/2008: http://www.archimuse.com/mw2003/papers/proctor/proctor.html.

[31] F. Sparacino, K. Larson, R. MacNeil, G. Davenport & A. Pentland, Technologies and methods for interactive exhibit design: from wireless object & body tracking to wearable computers. *In Proceedings of International Conference on Hypertext and Interactive Museums, ICHIM'99*, Washington, DC. Sept. 22-26, 1999. Last visited in 12/01/2008: http://ic.media.mit.edu/Publications/Conferences/MuseumTracking/PDF/MuseumTracking.pdf.

[32] Ultra-Mobile PCs, The Official UMPC site, Last visited in 27/02/2007: http://umpc.com/default.aspx.

[33] V. Vlahakis, N. Ioannidis & J. Karigiannis, ARCHEOGUIDE: Challenges and Solutions of a Personalised Augmented Reality Guide for Archaeological sites. *In Computer Graphics in Art, History and Archaeology, Special Issue of the IEEE Computer Graphics and Applications Magazine*, 22(5). pp. 52-60, 2002.

[34] G. Wilson, Multimedia Tour Programme at Tate Modern. *In Proceedings of the International Conference of Museums and the Web 2004*. D. Bearman and J. Trant (eds.). Archives & Museum Informatics, 2004. Last visited in 12/01/2008: http://www.archimuse.com/mw2004/papers/wilson/wilson.html.

[35] A. Woodruff, P.M. Aoki, A. Hurst & T.M.H. Szymanski, Electronic guidebooks and visitor attention. *In Proceedings of the 6th International Cultural Heritage Informatics Meeting*. pp. 437-454, 2004. Last visited in 12/01/2008: http://www2.parc.com/csl/members/woodruff/publications/2001-Woodruff-ICHIM2001-VisitorAttention.pdf.

[36] M. Zancanaro, O. Stock & I. Alfaro, Using Cinematic Techniques in a Multimedia Museum Guide. *In Proceedings of the 9th International Conference on Intelligent User Interface*, pp. 316 - 318, 2004.