ESEIG Mobile:
an m-Learning approach in a Superior School

Ricardo Queirós[1] and Mário Pinto[2]

[1] DI-ESEIG/IPP & KMILT, Porto, Portugal
ricardo.queiros@eu.ipp.pt
[2] DI-ESEIG/IPP & KMILT, Porto, Portugal
mariopinto@eu.ipp.pt

Abstract. In recent years, mobile learning has emerged as an educational approach to decrease the limitation of learning location and adapt the teaching-learning process to all type of students. However, the large number and variety of Web-enabled devices poses challenges for Web content creators who want to automatic get the delivery context and adapt the content to mobile devices. In this paper we study several approaches to adapt the learning content to mobile phones. We present an architecture for deliver uniform m-Learning content to students in a higher School. The system development is organized in two phases: firstly enabling the educational content to mobile devices and then adapting it to all the heterogeneous mobile platforms. With this approach, Web authors will not need to create specialized pages for each kind of device, since the content is automatically transformed to adapt to any mobile device capabilities from WAP to XHTML MP-compliant devices.

Keywords: SOA; interoperability; services; e-learning

INTRODUCTION

In ESEIG (Escola Superior de Estudos Industriais e de Gestão) - of the Polytechnic Institute of Porto - we use a Learning Management System (LMS) to provide access to the learning resources and activities. In a recent survey (see section 3) we verify that a large number of students use mobile devices. They are already experienced with mobile technology and are eager to use their devices in e-Learning scenarios. Another argument for the use of mobile devices came from the students' profile since most of them are already employed while studying in part-time. This situation decreases the chance to attend virtual events synchronously. Moreover, we also noticed that the students present different mobile devices with different characteristics that difficult the user experience regarding the access to mobile content. Based on these facts, we argue the need to automatically deliver uniform educational content on particular devices, normally referred as content adaptation.

In this paper we explore the use of open source technologies to provide a better design experience regarding mobile learning (m-Learning) content adaptation and promoting the "write once run anywhere" concept.

To understand the needs of our students we based on a survey conducted by a group of teachers at ESEIG. The aim of this study was to characterize the mobile devices usage, namely, the diversity of mobile technologies and services used by students and teachers, and analyze the future expectations concerning the usage of m-Learning platforms.

Based on this survey we obtained the basis for the ESEIG-Mobile system architecture. The ultimate goal of ESEIG-Mobile is to standardize the delivery of e-learning content to the mobile devices of our students. This system uses a three-tier model on a client–server architecture in which the user interface, functional process logic and data access are developed and maintained as independent modules. For each module, our concern was to use emergent and open-source solutions to leverage the potential of this new e-Learning paradigm where the characteristics of the mobile device of the student represents an important role and, at the same time, a huge issue. The large number and variety of Web-enabled devices poses challenges for Web content creators who want to automatic get the delivery context and adapt the content to mobile devices. This requires a thorough analysis of the available technologies and knowing good practices to help addressing this issue.

The remainder of this paper is organized as follows: Section 2 defines context delivery and enumerates several initiatives working on this subject. In the following section we present a survey made in our School regarding mobile devices. Then, we introduce the architecture of ESEIG Mobile and the design of its internal components. The next section we validate the ESEIG-Mobile prototype system based on the students' access statistics. Finally, we conclude with a summary of the main contributions of this work and a perspective of future research.

MOBILE CONTENT ADAPTATION

Mobile learning (m-learning) applications extend the electronic learning (e-learning) experience into the mobile context (Chang & Sheu 2002; Chen, Kao et al., 2002; Liu, Wang et al., 2002). M-learning uses mobile devices to enhance the teaching-learning process. However, it should not be seen as just another e-Learning channel for delivering the same content. In fact quality M-learning can only be delivered with an awareness of the special limitations and benefits of mobile devices (Parsons & Hyu, 2007). Due to those constraints the learning content must be adapted to suit the mobile device characteristics. Adaptation means a process of selection, generation or modification of content (text, images, audio and video) to suit to the user's computing environment and usage context (Parupalli, 2009). The concept of Content Adaptation is commonly related to mobile devices. Due to the variety of types and technologies supported they require special handling through a series of content transformations, in the deliver process, made by the content provider (server) (Zhang, 2007). Instead of authors having to create specialised pages for each kind of device, content adaptation automatically transforms an author's content to match the device characteristics. Some examples of such features are related with their limited computational power, small screen size, constrained keyboard functionality and media content type supported. The W3C Device Independence Working Group described many of the issues (Lewis,

2003) that authors must face in an environment in which there is an increasingly diverse set of devices used to access Web sites.

One approach is to use the common capabilities of the mobile devices and ignore the rest. Finding the Lowest Common Denominator (LCD) of the capabilities of target devices, will allow to you design a site that will work reasonably well in all devices. In order to allow content providers to share a consistent view of a default mobile experience the Best Practice WG has defined the Default Delivery Context (DDC) as a universal LCD (Rabin, 2008). This purpose is commonly adopt, however it limits the devices with better capabilities than LCD and decreases the use of a wider and heterogeneous mobile audience.

There are different adaptation points in the delivery of content to the device: server-side, in-network and client-side. The former needs to negotiate which version of a document should be delivered to a user in order to define the delivery context. One of the most widely used delivery context information is through the HTTP accept headers. These headers can be used to obtain the capabilities of a requesting device, such as, MIME types, character sets, preferred reply encoding and natural languages. In addition to the accept headers, the User-Agent header includes not standard information about the device and the browser being used. This lack of standardisation increases the difficult to interpret and extend this data (Gimson, 2006).

To overcome these difficulties emerged in recent years the device profiling concept - a repository of device capabilities, where a user agent (client) can supply the profile to the content provider (server), which can then adapt the content to suit the client device capabilities. The definition of the structure of the profile data is being covered by several standards, such as CC/PP (Kiss, 2010), User Agent PROFile (UAProf) (WAP, 2001) and Wireless Universal Resource FiLe (WURFL) (Passani, 2007).

The W3C CC/PP specification defines how client devices express their capabilities and preferences (the user agent profile) to the server that originates content (the origin server). The origin server uses the user agent profile to produce and deliver content appropriated to the client device. Using this specification, Web mobile content creators and user agents can easily define precise profiles for their products (e.g. Web servers use these profiles to adapt, through fine-tuned content selection or transformation, the content they serve to the needs of the Web device).

The UAProf (User agent profile) is a standard created by the Open Mobile Alliance (formerly the WAP Forum) to represent a concrete CC/PP vocabulary for mobile phones and defines an effective transmission of the CC/PP descriptions over wireless networks. Mobile phones that are conformant with the UAProf specification provide CC/PP descriptions for their capabilities to servers that use this information to optimize the content, where the information is communicated using XML containing several attributes (e.g. screen size, colour and audio capability, operating system and browser info, encoding).

The WURFL (Wireless Universal Resource File) is an XML configuration file, which contains information about device capabilities and features for a variety of mobile devices. Developers around the world contribute with device information and the WURFL repository is often updated reflecting new wireless devices coming on the market. In short, WURFL is a repository of wireless device capabilities describing the capabilities of common wireless devices worldwide and providing an API to programmatically query the capability repository.

Recently, to overcome the UAProf issues, the W3C MWI (Mobile Web Initiative) have outlined specifications for a Device Description Repository. These specifications include a formal vocabulary of core device properties and an Application Programming Interface (API). The consortium also published a working draft for a new independent language specification named W3C's DIAL (Device Independent Authoring Language). This specification is a language profile based on XHTML 2 and XForms, and uses the DISelect vocabulary to overcome the authoring for multiple delivery contexts. One known implementation is the XDIME language.

Others specifications arises recently to address the mobile content adaptation issues. It is the case of WNG (Passani, 2010) and WURFL. The Wireless Abstraction Library New Generation (WNG) is a Java tag-library that supports the use of universal mark-up for wireless devices. WNG allows the developer to write a web application once and have optimized content delivered to a variety of devices. It works on combination with the WURFL repository already detailed in this section.

Targeting e-Learning, several extensions appears recently to expose the LMS (e.g. Moodle) in mobile devices. One such case is the Mobile Moodle (MOMO).

All these standards and specifications help to formalize the design and implementation of mobile frameworks (Parsons, 2007; Paes & Moreira, 2008; Myers, 2004).

MOBILE EXPERIENCE SURVEY

An exploratory study concerning mobile devices usage was made at our Institution. The aim of this study was characterizing the mobile devices usage, namely the diversity of mobile technologies and services used by students and professors, and analyzing future expectations concerning the usage of m-Learning platforms (Queirós & Pinto, 2010; Seung-Won, 2005).

Research methodology

The survey was made using a questionnaire, sent to the Institution community, which includes almost a thousand and two hundred students, and eighty teachers. The questionnaire was sent by e-mail to all teachers, and the students were invited to answer the questionnaire through the Moodle e-Learning platform. The questionnaire was accomplished with a brief description of the study and their objectives, and it was structured in three main sections:

- Inquired profile: student or teacher;
- Services and technological characteristics: it comprises the identification of the main mobile services used and technological issues concerned with mobile devices;
- Educational mobile contents: it comprises the expectations about the usage of m-Learning platforms, the main services that they would like to use and the m-Learning constraints.

Results and discussion

We received one hundred and fifty valid questionnaires answers. From these ones, thirty two were from teachers and one hundred and eighteen were from students. Only two students answered that they haven't mobile devices. Regarding those who have mobile devices, we analyze that the majority of them owns a mobile device with Internet connection as shown in Figure 1.

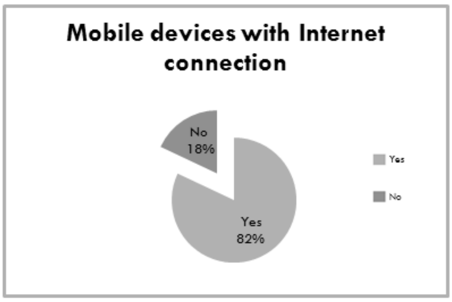

Fig. 1. Internet connectivity.

In fact, according to the survey results, eighty two percent of inquired persons have mobile devices with internet connectivity; from these ones, eighty six percent use internet connectivity based on GPRS (General Packed Radio Service) or WAP (Wireless Application protocol) technology, and only twelve percent of mobile devices support WiFi (Wireless LAN) technology.

One question addressed in the survey was about the main mobile services generally used by inquired persons. Figure 2 summarizes the achieved results.

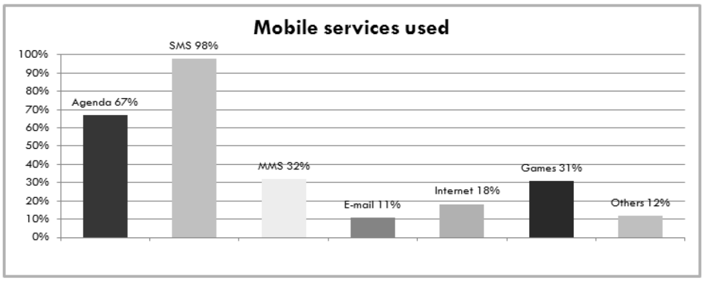

Fig. 2. Mobile services used.

Another issue addressed in the study was the potential role and expectations about educational mobile contents and services. Figure 3 summarizes the most relevant educational mobile services, according the survey answers.

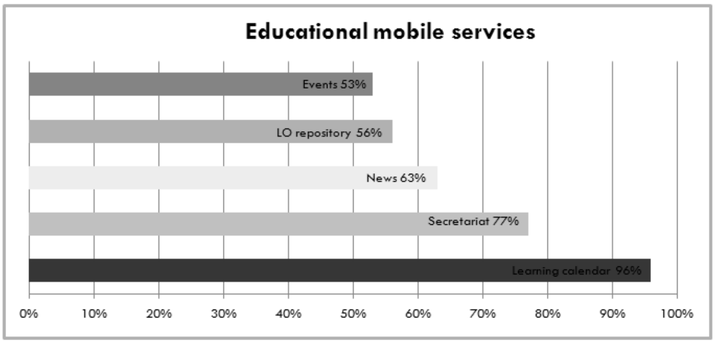

Fig. 3. Educational mobile services desired.

On the other hand, Figure 4 presents the main m-Learning constraints identified through the survey. The cost of the Internet provider, the screen dimensions and resolution are some of the students' complaints regarding the use of mobile devices.

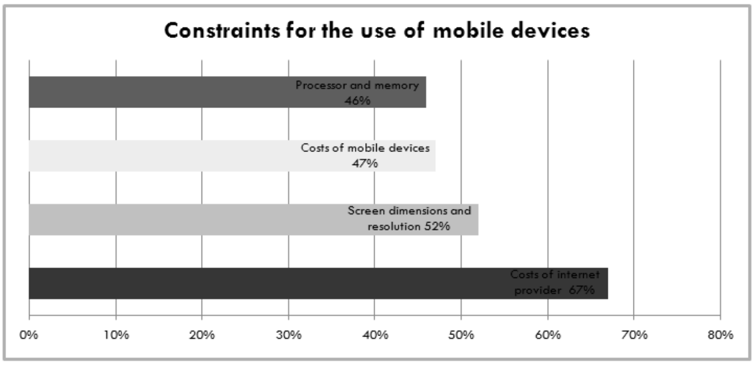

Fig. 4. Main constrains for the use of mobile devices.

The survey also includes two questions to analyse the expectations about the value added that m-Learning can bring to the students learning process. These questions are based on a likert scale of five degrees (Jameson, 2004), from nothing important (level one) to very important (level five).

One of them if about the potential role of m-Learning in the learning student's process: eighty six percent on inquired persons answered from important to very important, like shows Table 1.

Table 1. Role of m-Learning in the learning students process.

Likert scale	Answers (%)
Nothing important	3%
Some significance	13%
Important	39%
Significant	34%
Very important	13%

Another question is about the potential role of m-Learning in the distribution/access to learning contents: eighty five percent answered that m-Learning could perform an important or very important role in this field as shown in Table 2.

Table 2. Role of m-Learning in the distribution/access to learning contents.

Likert scale	**Answers (%)**
Nothing important	4%
Some significance	11%
Important	38%
Significant	42%
Very important	5%

According the survey results it is possible to present some considerations:

- Almost all students and teachers use mobile devices with internet connectivity, however these devices present different characteristics and support different technologies;
- There are a set of educational mobile contents and services, identified by inquired persons, that they would like to use in a m-Learning platform;
- A large percentage of students and teachers recognize the potential contribute of m-Learning in supporting educational contents and services, bringing added value to the learning students' process.

ARCHITECTURE

Based on the previous survey, we decided to design an open system, called ESEIG-Mobile. The ultimate goal of ESEIG-Mobile is to standardize the delivery of learning content produced at our School (ESEIG) to the diversity of mobile devices used by our students.

In the following subsections we present the overall architecture of the ESEIG learning systems and the concrete architecture of the ESEIG-Mobile system that will be integrated in the former. An evaluation of the ongoing development is also presented in the last subsection in order to validate our development strategy.

Overall Architecture

The ESEIG infrastructure comprises two servers as depicted in Figure 5. Each server is composed by several components organized as a three-tier model in which the users interface, functional process logic and data access is maintained as independent modules.

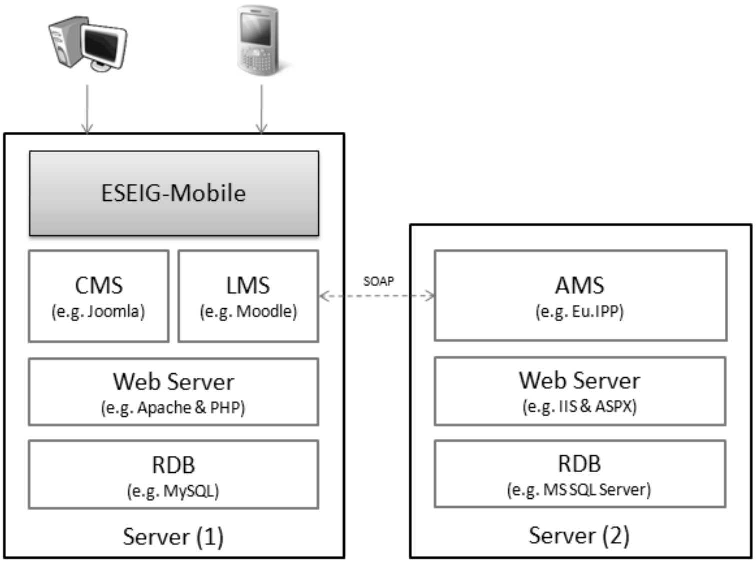

Fig. 5. Overall architecture.

The first server (1) stores the **Learning Management System** (LMS) and the **Content Management System** (CMS). The former is used for the administration, documentation, tracking, reporting of training programs, classroom and online events, and training content. The later is used for publishing and managing content on the World Wide Web and intranets.

The second server (2) stores the **Academic Management System** (AMS). An AMS aggregates all the information regarding administrative, financial, technical or scientific processes usual in educational institutions. Examples of these processes are the enrolment of students in courses, the management of grades or the payment of fees.

Both servers interact through Web services using SOAP. A typical scenario is the LMS importing data on students, courses and student enrolment in courses from the AMS to avoid the burden of entering this data manually.

The first server includes the ESEIG-Mobile system and for this reason we will give more importance in our study. In this server the installation of the components relied on XAMPP - a free and open source cross-platform web server solution package that includes the Apache HTTP Server, the MySQL database and a set of interpreters for scripts written in the PHP and Perl programming languages. In this server we installed a LMS and a CMS.

The selection of the LMS was based on the open source LMS systems available (e.g. Moodle, Sakai, .LRN or Dokeos) and in the significant share on the LMS market (Davis & Carmean & Wagner, 2009). Based on these criteria we choose Moodle as our LMS. Moodle is a free and open-source LMS written in PHP and created by Martin Dougiamas. Its name is an acronym for Modular Object-Oriented Dynamic Learning Environment. In early January of 2010, Moodle had a user-base of 46,624 registered sites with 32,464,992 users in 3,161,291 courses in 209 countries and in more than 75 languages (Cole & Foster, 2007). The most common functions of Moodle are the course information and documentation, documents repository, announcements, synchronous and a synchronous communication (email, chat room, discussion forum) and assignments.

The selection of the CMS was based on the available open source CMS (e.g. Joomla!, Drupal, Wordpress). From this list we choose Joomla! as our CMS. The Joomla! CMS is also written in PHP and stores data in a MySQL database. The most important features are a rich back-office to manage content in real time, page caching, RSS feeds, blogs, polls, search engine and support for language internationalization.

ESEIG-Mobile architecture

The architecture of the ESEIG-Mobile system is described by the component diagram shown in Figure 6.

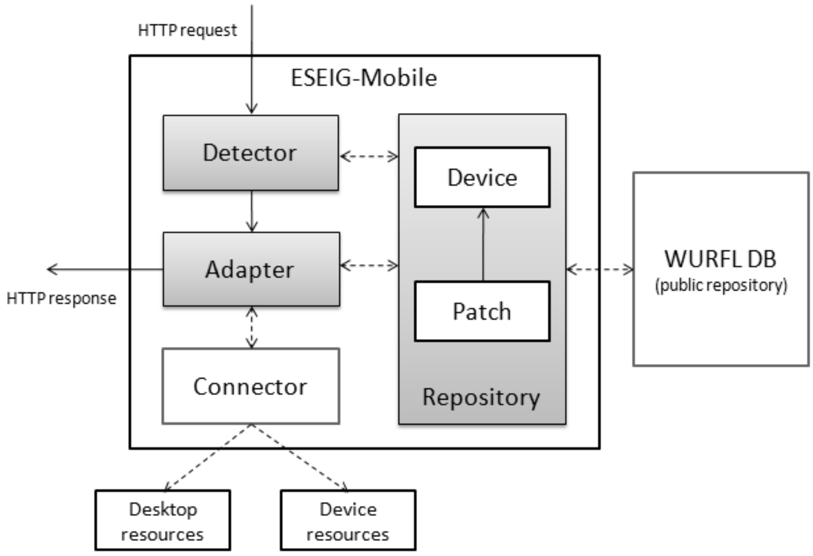

Fig. 6. Component diagram of the ESEIG-Mobile system.

The diagram includes the following components:

- The **Repository** component includes a repository with device capabilities and a patch to handle new updates. This component communicates with a public repository fed by a worldwide community;
- The **Detector** component receives HTTP requests and detects its origin by querying a special database formatted as an XML configuration file – the device repository;
- The **Adapter** component adapts the content based on the capabilities of the device. The adaptation process uses the Connector subcomponent to deal with the selection of resources based on the request of the Adapter component.

In the following sub-subsections these components are explained in more detail

The Repository component

The Repository component contains a file with a large list of device features based on WURFL. The WURFL is an open source database (based on a file called *wurfl.xml*) of wireless device capabilities. The WURFL repository can synchronize with a public repository of the WURFL DB where the developer community can make new additions to the WURFL DB. The **Patch repository** is a small XML file called *wurfl_patch.xml* that can enrich WURFL data dynamically. This file stores modified/enhanced groups and capability lists for new or existing WURFL devices. When the WURFL is parsed, the patch file is also imported to build a modified version of the device database.

The WURFL structure is formalize in a Document Type Definition (DTD) file. The DTD was the language inherited from Standard Generalized Markup Language (SGML) to define types of documents in XML. Its many limitations (e.g. insufficient data type support, lack of namespace awareness) (Harold, 2004) lead to an official W3C recommendation for a schema language called XML Schema Definition language (XSD) (Fallside, 2004) in 2001. XML Schemas are richer and more powerful than DTDs (Song & Zhang, 2004) and are written in XML. This new language overcame DTD limitations and provided several advanced features, such as the ability to build new types derived from basic ones (Biron, 2004), manage relationships between elements (similar to relational databases) and combine elements from several schemata (Queirós & Leal, 2009). Based on these facts, we convert the WURFL DTD file[1] in a XML Schema file.

The following figure shows an overall view of the WURFL XML Schema.

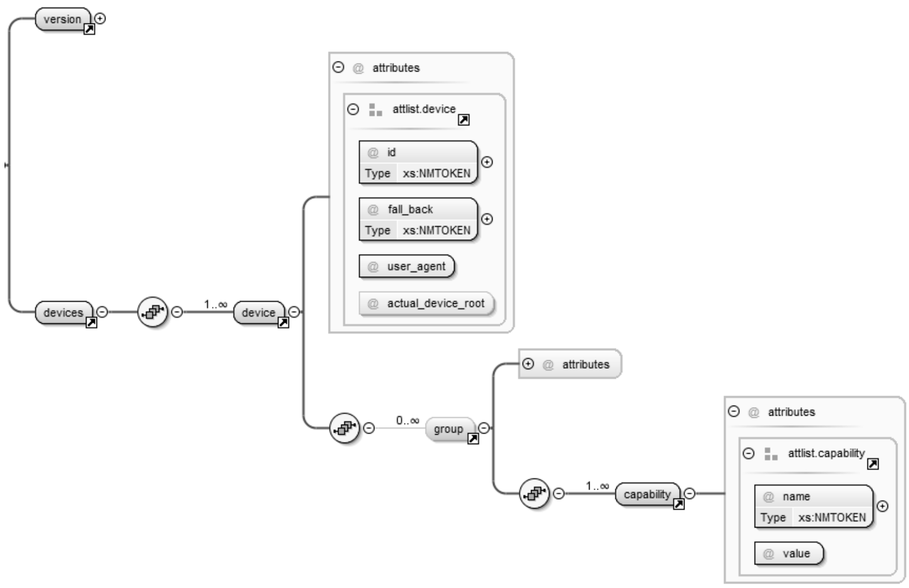

Fig. 7. The WURFL schema.

[1] Available at http://wurfl.sourceforge.net/wurfl.dtd

The schema has two top-level elements: the `version` and the `devices` elements. The `version` element is composed by a set of sub-elements:

- `ver` – the version of the WURFL database;
- `last_update` – the date of the last update of the database;
- `maintainers` – a set of maintainer elements related with the person(s) responsible by maintaining the database;
- `authors` – a set of author elements related with the person(s) responsible by creating the database;
- `authors` – description of the database.

The `devices` element contains one or more `device` sub-elements that model a certain device. This element contains the `user_agent` attribute, the device `id` attribute (created by the WURFL maintainer), the `fall_back` attribute (gives a way to infer more information about the device) and the `actual_device_root` attribute signals that the current device element may be chosen as the representative for all devices by the same brand and model name.

In addition to this data, a device element may carry information about device features commonly referred to as capabilities. A device capability is an XML fragment which contains information about a specific feature of a given device. The device capabilities are organized in groups. Groups are used to improve the readability of the WURFL XML database by humans.

For instance, Nokia phones support tables because *fall_back* is defined as generic (WURFL default) as described in the following piece of code.

```
<device user_agent="Nokia" fall_back="generic"
id="nokia_generic">
 <group id="ui">
  <capability
      name="break_list_of_links_with_br_element_recommended"
      value="false" />
 </group>
</device>
```

The WURFL is based on the concept of family of devices. All devices are descendent of a generic device, but they may also descend of more specialized families. This mechanism, called *'fall_back'*, lets programmers derive the capabilities of a given phone by looking at the capabilities of its family, unless a certain feature is specifically different for that phone (Passani, 2007).

The Detector component

The **Detector** component receives HTTP requests and detects its origin by querying the WURFL device repository. A Web browser, when requesting a web page, sends a set of HTTP headers which is included the user-agent header. In PHP it's easy to obtain the value of this header by using the global variable `$_SERVER['HTTP_USER_AGENT']`. Using WURFL we can verify if the request was made either by a desktop computer browser or a mobile device browser. The following code presents a simple way to obtain the origin of an HTTP request:

```
<?php
   require_once('/myDirectory/wurfl_config.php');
   require_once(WURFL_CLASS_FILE);
   $wurflObj = new wurfl_class();
   $wurflObj-
>GetDeviceCapabilitiesFromAgent($_SERVER["HTTP_USER_AGENT"]);

   if ($wurflObj->getDeviceCapability('is_wireless_device') )
   { header("Location: http://mobile.eseig.ipp.pt ");}
   else { header("Location: http:// eseig.ipp.pt"); }
?>
```

The code above uses the `getDeviceCapability` method. This method expects a capability name as a parameter and returns the respective capability value in return. For testing purposes there is a Firefox extension called "User Agent Switcher" that allows the manual edition of the user-agent.

The Adapter component

The Adapter component is the responsible for the content adaptation of the requests from the client mobile devices. This adaptation will be ensured, in a near future, by the use of WNG (Passani, 2010). WNG is a JSP tag library that abstracts the mark-up differences in all known wireless devices and allows the page creation similar to HTML, while delivering WML, C-HTML and XHTML Mobile Profile to the client device. Device capabilities are queried dynamically using the WURFL API. The connector component handles the connection with web resources and deals with the information querying and merging from the specific resources.

In this moment ESEIG-Mobile is in early development as we are only detecting if the HTTP request is made from a mobile device. We use the WURFL API to query the repository based on the *User Agent* header of the request and present a resource suitable to the respective device capabilities. The following snippet of code demonstrates how the detection is performed and how we can query a particular device capability (e.g. maximum number of color supported):

```
…
require_once('./wurfl_config.php');
require_once(WURFL_CLASS_FILE);
$userAgent = $_SERVER['HTTP_USER_AGENT'];
$wObj = new wurfl_class();
$wObj->GetDeviceCapabilitiesFromAgent($userAgent);
$max_colors = $wObj->getDeviceCapability('colors');
…
```

The example above returns the maximum number of colors supported by the device. Based on this value the connector will serve the suitable web resource according with the requester device capabilities as shown in Figure 7.

Fig. 8. An ESEIG-Mobile resource.

USAGE STATISTICS

In order to characterize the levels of access and usage of the ESEIG-Mobile web interface, a data set was collected through the Google Analytics service. This service is connected to the ESEIG-Mobile system, gathering a comprehensive set of data and statistics related with hit counters, rejected requests, new visitors, traffic and mobile operating systems used to access the ESEIG-Mobile interface.

The data was collected since November 2010 and in this paper we analyze the last four months, i.e., to February 2011.

In the following subsections we present the overall data and statistics collected through the Google Analytics service, in order to better understand the adherence to the mobile web interface, its evolution over time and the main weaknesses. These results are important to better understand the difficulties or constraints on the usage of ESEIG-Mobile system by ESEIG students and also to monitor the interest to use the mobile services offered through this system.

Data analysis

According to Figure 9, the access number (hit number) has risen hardly in the last few months. In fact, in November 2010 we had only eighteen system accesses; on the other hand, in February 2011 the hits number was three hundred and seven. Despite this strong grow, the number of new visitors has fixed around 60% of total accesses. These data clearly indicates that there are a large number of users who are still making the first contact with de ESEIG-Mobile services.

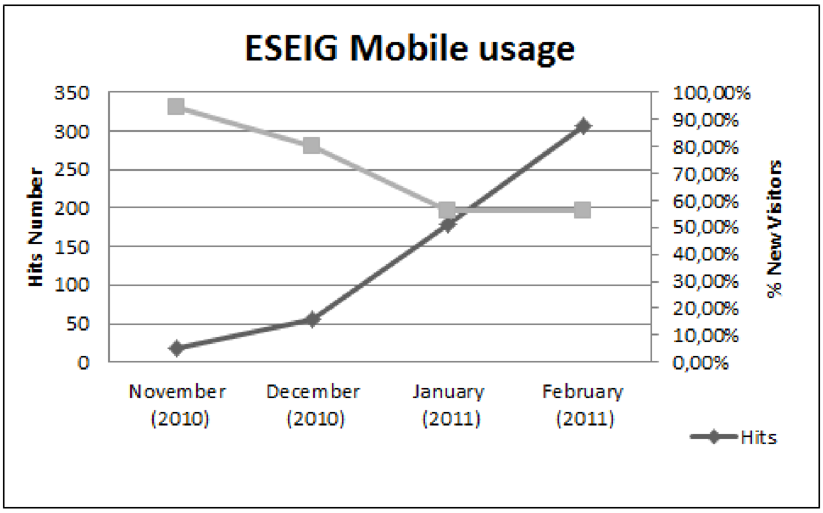

Fig. 9. ESEIG-Mobile usage: hits and new visitors.

Other interesting data that were collected refer to mobile operating systems and platforms used to access the ESEIG-Mobile web interface, depicted in Figure 10. Based on these data it is possible to relate each of the mobile access platforms used by students with the service rejection rate and also the number of hits.

This in a relevant information, since it allows us to understand which are the mobile platforms most commonly used, and which ones have a higher rate of access rejection. That information is an important feedback regarding the efficiency and effectiveness of the Repository, Detector and Adaptor components included in the ESEIG-Mobile system.

In fact, Symbian and Android are the main platforms used to access the ESEIG-Mobile interface, followed by iPhone, iPad and iPod. A surprising fact is the lower number of devices with the Windows Mobile operating system.

Moreover, the rejection rate in accessing the ESEIG-Mobile services is very low when using mobile devices with Windows Mobile or Samsung platforms. With iPhone and Android platforms the rejection rate is near to 50%; with other systems, the tendency is for higher rates of rejection. In order to obtain lower rates of rejection it is crucial to continue:

- to improve the graphical user web interface (GUI) of the ESEIG-Mobile and its usability;
- to improve the Detector and Adaptor components of ESEIG-Mobile system.

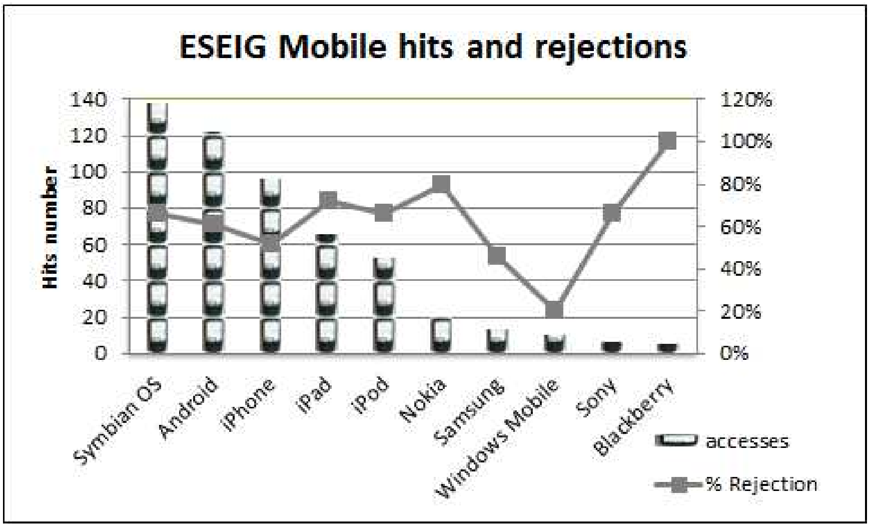

Fig. 10. Mobile devices operating systems.

Another important issue is to compare and monitoring the traffics evolution of both ESEIG-Mobile web interface and the traditional ESEIG web interface, accessible from any desktop or laptop. Table 3 illustrates de hit numbers from both platforms, over the last four months.

Table 3. ESEIG traffic data

Months	ESEIG-Mobile hits	ESEIG Desktop hits
November 2010	18	21.351
December 2010	55	16.388
January 2011	179	29.085
February 2011	307	31.417

In fact, there are differences in the number of hits between the two platforms. However, both platforms increased the number of requests over time. The ESEIG-Mobile web interface has recorded higher growth rates, according data in Table 3. This fact is due to the new trends on the learning paradigm where the mobile learning occupy a main role and to the evolution and dissemination of the mobile devices.

Data discussion

The results presented in the above subsection illustrate the current status on the usage of ESEIG-Mobile web interface. The analysis of this data could help us to better understand the strengths and weaknesses in the access and usage of the ESEIG-Mobile web interface.

Regarding the access rate, one can consider that although the access rate is relatively low, it has increased significantly. The amounts collected can result from the fact that the platform is very recent, and therefore still unknown by most students. Moreover, the high rates of new visitors may indicate that the ESEIG-Mobile web interface starts to be increasingly popular.

Another important issue that arises from the data analysis is the diversity and heterogeneity of the client devices. Symbian, Android, iPhone and iPad are the leading mobile devices, but there are a large number of other devices that ESEIG-Mobile system should respond. This fact proves the need to find an approach to deliver uniform content to address the heterogeneity of mobile devices existent nowadays.

It is also important to understand the reasons for the high rejection rates observed in some mobile devices. This will be, certainly, a critical success factor for ESEIG-Mobile interface. We anticipate two possible causes for these rejection rates:

1. Incomplete representation of the device characteristics - the Repository component included in the ESEIG-Mobile system contains a file with a large list of device features based on WURFL. This list could need to be complemented with a more comprehensive set of device features.
2. Performance issues – each request is detected and validated using the WURFL database. A cache-aware approach could increase the ESEIG-Mobile responsiveness. The Tera-WURFL project is a good candidate to fulfill this requirement since it uses a MySQL database backend to store the WURFL data and caches the results of device detections. The project also supports implementations of WALL such as WALL4PHP to create websites in an abstract language which is delivered to the visiting user in several formats (e.g. CHTML, XHTML, XHTML-MP) based on the mobile browser's support.

CONCLUSION

In this paper, we present several approaches for defining delivery context and also a survey targeted to ESEIG students and teachers that base our work. The survey shows the real perspectives and expectations of the students and teachers' community on this emergent field of educational mobile contents. The mobile devices advent could enable a more useful proximity between students and teachers, facilitating and promoting the learning process.

In order to address the main issues regarding the heterogeneity of mobile devices found in our community we also present the design and the prototype implementation of an open system for the delivery of suitable and uniform e-Learning content to the mobile devices of our students.

To validate our approach we present the usage statistics of the ESEIG-Mobile project based on the Google Analytics data. The analysis of this data is very important since it helps us to confirm and understand the heterogeneity of the students' mobile devices and their usage habits and preferences. It also helps to identify and find the best approaches to improve the ESEIG-Mobile system.

In this moment ESEIG-Mobile is in early development as we are only detecting if the HTTP request is made from a mobile device and query some device capabilities from the WURFL device repository. We expect some challenges in the prototype implementation process regarding, for instance, the transformation of the Web resources in the WNG format. For this task we are considering using Extensible Stylesheet Language for Transformation (XSLT) to formally describe the transformations. Other ongoing work is related with increasing the device repository performance migrating from the WURFL XML database to a relational database (e.g. MySQL) using, for instance, the Tera-WURFL project.

References

Biron, P. V., & Malhotra, A. (2004). XML Schema Part2: Datatypes Second Edition. W3C Document.

Chang, C., & Sheu J. (2004). Design and implementation of ad hoc classroom and eSchoolbag systems for ubiquitous learning. In IEEE Int. Workshop Wireless and Mobile Technologies in Education.

Chen, Y. & T. Kao, et al. (2002). A mobile scaffolding-aid-based bird-watching learning systems. In IEEE Int. Workshop Wireless and Mobile Technologies in Education.

Cole, J. & Foster, H. (2007). Using Moodle - Teaching with the Popular Open Source Cours Management System. O'Reilly – Community Press. ISBN 9780596529185.

Davis, B., Carmean, C. & Wagner, E.D. (2009). The Evolution of the LMS: From Management to Learning - Deep Analysis of Trends Shaping the Future of eLearning, Sage Road Solutions, LLC.

Fallside, D. C. (2004). XML Schema Part 0: Primer Second Edition. World Wide Web Consortium, Recommendation.

Gimson, R., Lewis, R. & Sathish, S. (2006). Delivery Context Overview for Device Independence - W3C Working Group Note, from http://www.w3.org/TR/di-dco.

Harold, E.R. (2204). The XML Bible, 3rd edition, Hungry Minds.

Jameson, S. (2004). Likert Scales, how to (ab)use them. In Blackwell Publishing, Nº 38, 1212-1218.

Kiss, C. (2010). Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 2.0 - W3C Working Group Note, from http://www.w3.org/TR/CCPP-struct-vocab2/

Lewis, R. (2003). Authoring Challenges for Device Independence, from http://www.w3.org/TR/acdi/

Liu, T., H. Wang & et al. (2002). Applying wireless technologies to build a highly interactive learning environment. In IEEE Int. Workshop Wireless and Mobile Technologies in Education.

Myers, A., Nichols, J.& Miller, R. (2004). Taking Handeld Devices to the Next Level. In IEEE Computer Society, from http://www.cs.cmu.edu/~pebbles/papers/pebblesControlIEEE.pdf

Paes, C., Moreira, F. (2008). Aprendizagem com dispositivos móveis: aspectos técnicos e pedagógicos a serem considerados num sistema de educação. In V Conferência Internacional de Tecnologias de Informação e Comunicação na Educação.

Parsons D. & Ryu, H. (2007). Software Architectures for Mobile Learning. In Mobile Learning Technologies and Applications, eds. David Parsons and Hokyoung Ryu, Massey University, 2007. ISBN 978-0-473-11947-8.

Parsons, D., Ryu, H & Cranshaw, M. (2007). A Design requirements framework for mobile learning environments. In Journal of Computers, Vol. 2, Nº 4.

Parupalli, R. (2009). Dynamic Content Adaptation to Mobile Devices. In 3rd National Seminar on e-Learning and e-Learning Technologies, India.
Passani, L. Wireless Universal Resource FiLe (WURFL), from http://wurfl.sourceforge.net/
Passani, L. (2010). Introducing WALL: a Library to Multiserve Applications on the Wireless, from http://wurfl.sourceforge.net/java/tutorial.php
Queirós, R. & Leal, J.P. (2009). schem@Doc: a web-based XML Schema visualizer. In Inforum - Simpósio de Informática - 7th edition of XML: Aplicações e Tecnologias Associadas, Lisboa, Portugal.
Queirós, R & Pinto, M. (2010). ESEIG Mobile: An m-Learning Approach in a Superior School. In Enterprise Information Systems International Conference, vol.110, part 5, pp.355-363, Viana do Castelo, Portugal.
Rabin, J. & McCathieNevile, C. (2008). Mobile Web Best Practices 1.0" – basic guidelines, from http://www.w3.org/TR/mobile-bp/#ddc
Seung-Won, N. (2005). Design and Implementation of Resource Sharing System for Creation of Multiple Instructions In Mobile Internet Environment. In Advances in Intelligent and Soft Computing, vol. 29, Soft Computing as Transdisciplinary Science and Technology, pp. 797-807.
Song, G. & Zhang, K. (2004). Visual XML Schemas Based on Reserved Graph Grammars. In International Conference on Information Technology: Coding and Computing (ITCC'04), Volume 1.
Wireless Application Protocol Forum, Ltd. User Agent Profile (UAProf), from http://www.openmobilealliance.org/tech/affiliates/wap/wap-248-uaprof-20011020-a.pdf
Zhang, W., Kunz, T. & Hansen,K.M. (2007). Product line enabled intelligent mobile middleware. In Proceedings of the 12th IEEE Interational Conference on Engineering of Complex Computer Systems, Auckland, New Zealand, pages 148-157.