The Journal of Systems and Software 84 (2011) 1883–1891

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

State of the art of frameworks and middleware for facilitating mobile and ubiquitous learning development

Sergio Martin*, Gabriel Diaz, Inmaculada Plaza, Elena Ruiz, Manuel Castro, Juan Peire

Electrical and Computer Engineering Dep., Spanish University for Distance Education (UNED), Juan del Rosal 12, 28040 Madrid, Spain

ARTICLE INFO

Article history:
Received 17 January 2011
Received in revised form 13 May 2011
Accepted 16 June 2011
Available online 23 June 2011

Keywords:
Framework
Middleware
Mobile and ubiquitous learning

ABSTRACT

The emergence of mobile and ubiquitous technologies as important tools to complement formal learning has been accompanied by a growing interest in their educational benefits and applications. Mobile devices can be used to promote learning anywhere and anytime, to foster social learning and knowledge sharing, or to visualize augmented reality applications for learning purposes. However, the development of these applications is difficult for many researchers because it requires understanding many different protocols; dealing with distributed schemas, processes, platforms, and services; learning new programming languages; and interacting with different hardware sensors and drivers. For that reason, the use of frameworks and middleware that encapsulate part of this complexity appears to be fundamental to the further development of mobile learning projects. This study analyzes the state of the art of frameworks and middleware devoted to simplifying the development of mobile and ubiquitous learning applications. The results can be useful to many researchers involved in the development of projects using these technologies by providing an overview of the features implemented in each of these frameworks.

1. Introduction

Technology is becoming increasingly important in education as part of the day-to-day learning experience. During the last part of the 20th century, technologies related to communications and the Internet, including e-learning platforms, led to revolutions in several fields.

More recently, over the past decade, the Web has evolved from Web 1.0, where most users played the passive role of mere readers, to Web 2.0, where users have become active players, publishing their own content and interacting in social networks (Jarvis, 2009). However, in addition to the Social Web, many other new technologies are developing with the potential to have a deep impact on education (Johnson et al., 2010), including games, augmented reality, new human–computer interfaces, and ubiquitous and mobile technologies.

Most of these technologies use mobile devices to grow and reach their audiences, taking advantage of the ability of mobile devices to integrate with ubiquitous technologies (Naismith et al., 2004). High-speed Internet connectivity is currently available anytime and anywhere on these devices because of 3G and UMTS networks. Mobile applications can incorporate a user's location to provide personalized services using location methods, such as GPS, WiFi, or cell tower triangulation or identification technologies such as RFID (Radio Frequency Identification) or NFC (Near Field Communication). This context-awareness has the potential to revolutionize new mobile social applications by taking into account not only friends' profiles but also their locations when starting conversations or establishing new friendships (Hansen and Bouvin, 2009; Baldauf et al., 2007). These devices are also re-inventing games (Boticki et al., 2009), as they can use their Internet connections to connect to multi-player games, location technologies to create outdoor games, and new human–computer interfaces (HCI), such as movement sensors and tactile screens (Liestøl, 2009), to immerse players in more realistic games (Lavin-Mera et al., 2009). In general, all of these technologies are taking advantage of the spatial and temporal flexibility offered by mobile devices.

This evolution is leading to new learning environments where applications must be aware of users' needs to personalize the information and services they provide depending on the user's context (i.e., geographic location, user profile and preferences) and those of other nearby users (Keegan et al., 2009; Yau and Joy, 2008). These applications should offer communication and collaboration features that can be used to build social-based knowledge, should be integrated with existing learning platforms, such as Moodle, for gathering the students' e-portfolios, and should use existing services and knowledge (e.g., communication and collaboration methods, assessment, learning objects) (Cheung et al., 2006; Conde et al., 2009). Finally, these learning applications should interact with each other in a digital ecosystem of mobile applications and

* Corresponding author. Tel.: +34 913987623; fax: +34 913986028.
E-mail addresses: smartin@ieec.uned.es (S. Martin), gdiaz@ieec.uned.es (G. Diaz), iplaza@unizar.es (I. Plaza), elena@issi.uned.es (E. Ruiz), mcastro@ieec.uned.es (M. Castro), jpeire@ieec.uned.es (J. Peire).

0164-1212/$ – see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2011.06.042

services through accepted standards, fostering interoperability and easy extension.

However, the development of these applications is difficult for many researchers because it requires them to understand many different protocols, to deal with distributed schemas, processes, platforms, and services, to learn new programming languages, and to interact with different hardware sensors and drivers (Schmidt and Forbess, 1999; Ryan et al., 1999; Biegel and Cahill, 2004). For that reason, frameworks and middleware that encapsulate part of this complexity are fundamental to ensuring the successful development of complex mobile learning projects, helping developers focus on learning design and creativity instead of dealing with sensors and services.

This paper studies the state of the art in frameworks and middleware that facilitate the development of mobile and ubiquitous learning applications. Just a few frameworks are devoted to mobile learning because most of the systems are designed for general purposes, not just for education. These systems might also be used for mobile learning purposes, although they will fail to meet some specific learning-related requirements, such as support for learning standards and e-portfolios and integration with e-learning platforms.

The paper is structured in five sections. The first introduces the motivation of the study. The second section describes the methodology used to gather and compare the different frameworks and middleware found in the literature. The third section provides an overview of the main systems found, laying the groundwork for the fourth section, which includes a comparison and analysis. Finally, some conclusions are provided in the fifth section.

The results can be useful for researchers involved in the development of mobile and ubiquitous learning projects by providing an overview of the features implemented in each platform, so that researchers can decide which framework is most suitable for their project. The study can also be useful for developers who create new frameworks or middleware for mobile and ubiquitous learning. The analysis offers a clear picture of the features implemented by the existing systems, which developers can then improve or complement.

2. Methodology

The first stage of this study defines the features that must be analyzed when studying in frameworks and middleware for mobile learning.

The second stage consists of a literature analysis of the existing frameworks and middleware for the creation of mobile and ubiquitous learning applications, giving a brief summary of their contributions and main functionalities.

The third stage provides an overview of the analyzed systems, focusing on their operating systems, programming languages, and main purposes. This stage offers a preliminary overview of these systems.

Finally the fourth stage compares the different features of each analyzed system. The comparison uses a binary scale to show the implementation of each feature. Finally, we draw some conclusions to help researchers find the most suitable system for their mobile or ubiquitous learning projects.

3. Results

3.1. Features definition

Biegel identified the main features that a programming model to develop context-aware applications in mobile environments must address (Biegel, 2004):

- *Context acquisition.* The method of context-data acquisition is very important when designing context-aware systems because it predefines the architectural style of the system, at least to some extent (Baldauf et al., 2007). Gu classifies contexts into two main categories based on the means by which context is obtained (Gu et al., 2005).
  - o Direct context is directly acquired or obtained from a context provider, which can be an internal source such as an indoor location provider or an external source such as a weather information server.
  - o Indirect context is derived by interpreting direct context through context reasoning.
- *Uncertainty of context data.* Measurements made of the real world by sensors based on physical transducers will always contain a degree of uncertainty and incompleteness, which together result in an inherent unreliability of context data based on such measurements (Biegel, 2004). Thus, using several sensor sources is recommended to minimize this uncertainty.
- *Representation of context data.* To enable the application to process, reason about, and react to context data, the application developer must take a systematic approach to the representation of context data. The selected representation format should allow the application to efficiently process and reason about context data (Biegel, 2004). Finally, the last relevant feature to define regarding architectural design is how to represent context information.
- *Privacy.* Traditional concerns regarding privacy are amplified in context-aware applications that are predicated on access to a wide range of sensitive data and involve ad hoc collaborations between entities (Biegel, 2004).
- *Scalability.* Scalability refers to the ability to incrementally increase the abilities of a system while maintaining or improving performance. It is essential that developers of context-aware applications in mobile environments have appropriate abstractions and system support available to ensure that their applications are scalable (Biegel, 2004).
- *Synchrony.* Most distributed applications with sensors distributed in the environment are based on synchronous operation in which an operation has to wait for a response before execution can continue. This method of operation is inadequate for context-aware applications that need to be notified asynchronously when new context data is available (Biegel, 2004).
- *Extensibility and reusability.* Extensibility can be defined as the ability to add new functionality to an application, while reusability can be defined as the ability of a functionality to be used again, unmodified, in a different system. Facilitating the extensibility and reusability of application components enables the incremental evolution of applications, reducing development effort and the need to develop new functions from scratch (Biegel, 2004).

Georgieva provides two types of classification (Georgieva et al., 2005), i.e., information and communication technologies, as in Naismith et al. (2004), and educational technologies. The educational-oriented classification is based on the following main indicators:

- Support of synchronous and/or asynchronous communication.
- Support of e-learning standards. Currently most m-learning systems do not support e-learning specifications such as SCORM.
- Access to learning materials and/or administrative services. This means that the student receives course content such as tests or administrative information such as schedules and exam marks.

In the process of designing and developing a mobile learning project, in addition to identifying the relevant sources of context data and services for a particular application, the developer also often has to write low-level code to interact with sensor hardware at the device protocol level (Schmidt and Forbess, 1999; Ryan et al., 1999). This development is time-consuming, error-prone, and is only accessible to fairly experienced programmers (Biegel, 2004).

The proliferation of mobile devices has also increased demand for mobile application development. For that reason, frameworks that provide an API are becoming a fundamental piece of the mobile computing puzzle, enabling the reuse of common components, decreasing the complexity of programming tasks, and simplifying the design and development of mobile applications.

However, a software framework has no future in the open-source era if its code is proprietary. Openness means that more people have access to a framework through an open community of users. Some of these users search for already developed solutions while others want to contribute, improving the framework and sharing improvements within the community (Jarvis, 2009). However, most frameworks do not reach critical mass, remaining isolated research projects without a real impact on society.

3.2. Frameworks and middleware devoted to mobile computing

Regarding to frameworks and middleware, there are many general purpose, i.e., non learning-oriented, frameworks that can be used to create mobile applications. For example, Dey described one of the first such systems, “The Context Toolkit”, in Dey et al. (2002). Gu proposed the Service-Oriented Context-Aware Middleware (SOCAM project; Gu et al., 2005). A similar architecture was introduced in the CASS project (Context-awareness sub-structure) developed by Fahy and Clarke (2004), and another framework based on a layered architecture was constructed in the Hydrogen project (Hofer et al., 2002).

Hansen took a different approach in developing the HyCon Framework, which is mainly devoted to hypermedia services, although it also supports J2ME applications (Hansen and Bouvin, 2009). Its architectural design is similar to that of other hypermedia architectures such as the Dexter architecture (Halasz and Schwartz, 1994), the Open Hypermedia System Working Group’s Open Hypermedia Model (Reich et al., 1999) and Construct (Wiil et al., 2001).

Gupta proposed the MobiSoC middleware system (Gupta et al., 2009), which is devoted to supporting mobile community-centric social computing applications. The SAMOA middleware described in Bottazzi et al. (2007) is similar to MobiSoC, although SAMOA is user-centric instead of community-centric.

Shriram developed an environment (Community Network) in which service providers (hardware and network operators) can publish services and deliver content to users (Shriram and Sugumaran, 2007). This environment offers some security features, encrypting the information sent. The environment offers a service discovery feature where all services are registered and a querying mechanism to locate services and resources. This system is designed to support location-based technologies to personalize services.

Davidyuk developed CAPNET (Davidyuk et al., 2004), a context-aware middleware aimed at managing user context (sensing, data processing) and supporting multimedia and communication services, providing developers of media applications with functionalities to capture images, audio and video. Stavroulaki developed the DAFNE architecture, an open framework that provides personalized services in an intelligent environment (Stavroulaki et al., 2006). Finally, another interesting framework is the Open Mobile Access Abstract Framework (OMAF), developed in the MOBIlearn European project (Bormida, 2002) and based on layers of infrastructure and application profiles.

3.3. Frameworks and middleware devoted to mobile learning

There are also some specific frameworks devoted to educational environments. These learning-oriented frameworks provide functionalities intended to enhance learning through mobile devices.

The first such system is the COMTEXT framework (Zanela et al., 2009), which is devoted to the design of ubiquitous learning applications that consider four main elements: the learner’s known profile and needs, the context (physical, temporal and social) that surrounds mobile learners, the educational paradigm or model considered, and the possibilities and limitations of mobile and wireless technologies. This framework was designed as a tool for testing concepts related to competence development and ubiquitous learning. It is a web-based environment that provides access to the following tools:

- Communication/collaboration: the environment integrates blog, forum, and email tools as well as access to Skype (used for synchronous interaction via chats or direct conversation). For collaboration, the environment includes mindmap software. These tools are integrated into the system with direct access.
- Multimedia: access to YouTube to download videos.
- File repository: on-line repository of documents (including maps, simulations, presentations, demonstrations, a manual about processes or procedures, and all types of documents related to the context of work).
- Assessment: ad hoc assessment system.
- Location: the system supports location determination by GPS, cell towers, and Wi-Fi.
- Context-management: association of coordinates with names (called contexts).
- Reminders: this tool reminds the user of some activities but is not connected to any platform (company, university, LMS calendar). The system reminds the user of events but does not provide access to those events.

Chen developed a simulation of a framework for context management focusing on the method used to share context (push or pull based sharing) (Chen et al., 2007). The proposed architecture divides nodes into mobile context managers. These nodes are organized into an overlay segment-tree virtual network and context providers/service requestors which send/receive contextual information via the segment-tree virtual network. The mobile context managers not update all topology information in the overlay network, but they update periodically the information of neighboring nodes in the overlay network. Authors propose a push-based approach to handle real-time information and combine pull-based approach for supporting context-aware environments in MANETs. The proposed framework can efficiently support context-aware environments in MANETs. Furthermore, authors propose two approaches: push-based and pull-based to handle emergency and services supporting, respectively. The system does not retrieve any location information, which is outside its scope. These authors assume that the system will be implemented on a framework that will implement this feature. This framework does not provide any educational, collaborative or communication features.

The AGORA (Arrufat et al., 2008) environment provides collaboration (chat and photo sharing) features through middleware and facilitates the development of new applications by a set of plug-ins that helps to integrate new services into the system. AGORA architecture can be defined in three main components. In first place, the plug-in framework allows fast development of applications as plug-ins. These plug-ins can be packaged, exchanged between peers and even be installed on the run. This is achieved thanks to the plug-in manager, which also verifies integrity and controls plug-in lifecycle. The plug-in framework eases application development, whereas middleware offers a rich set of collaborative functionalities. The application level multicast is in charge of reducing group communication overhead. By deploying these three components together, the authors offer a complete solution for rapid application development in MANETs. AGORA's collaboration middleware provides synchronous and reliable communication: communication channels, naming and publish/subscribe services. This collaboration middleware provides different communication mechanisms as well as group management primitives. Membership information, named communication channels and different communication paradigms are available for plug-ins. The authors have also focused on improving communication performance (multicast). This system does not contain location-based features.

Belimpasakis developed a middleware system called "Sharing Middleware" (Belimpasakis et al., 2008), which allows third party developers to create applications with sharing capabilities while remaining agnostic regarding the lower level sharing technologies, protocols and data transfer interfaces. The goal of this system is to allow developers to create applications with powerful sharing capabilities but limited programming complexity. This system provides a Service Directory that allows users to discover and use other developers' services. The system supports content with metadata (DublinCore and Digital Item Declaration Language Lite), simplifying content searching. One of the main features of the system is data sharing. This feature is supported by the use of several types of feed formats, depending on the environment. UPnP AV is used for the home environment, as it supports many electronic devices. ATOM is used for the Internet domain (e.g., blogs, photo sharing). For the enterprise domain, the WebDav protocol is used because this protocol is supported by most distributed storage systems. Using these feed protocols, users can upload and download content. This system offers a programming API to simplify programming tasks. It provides the sharing middleware with uniform access to a number of sharing protocols, each wrapped in a dynamically registered plug-in. The plug-ins, in turn, utilize the data transport services provided by different access bearers/interfaces. However, the API is not very intuitive, so it still requires significant effort on the part of programmers.

Liu developed a Mobile Multimedia Management System that supports course management, material (including multimedia) storage and search and self-adaptive content adjustment (Liu et al., 2007). The key parts of the proposed system are the mobile learning management system and the multimedia learning engine. The main functions of the mobile learning management system are course management, material storing and searching and self-adaptive contents adjusting. This system includes a mobile instructor system, which provides teaching functions used in teacher such as bulletin board. The database is used to store teaching content with multimedia format. The system can either send teacher's voice/video information directly to student by streaming media method, or allow student select class material from available list stored in mobile database to learning. The multimedia learning engine handles the multimedia data transfer between teachers and students using the wireless network. It also provides interactive tools (i.e., FAQ, chat) and a notes system. This multimedia learning engine also contains a media player that provides services including multimedia browse, play control, etc. The use of an Mobile Multimedia Management System is an important contribution of this work, although it re-uses existing environments rather than developing a new educational environment. The most remarkable feature of this framework is its real-time multimedia content delivery.

Gang and Zongkai (2005) developed a framework based on two layers: the multimedia adaptation layer, devoted to multimedia management (e.g., image size changing, frame dropping of encoded videos or cross-media conversion), and the learning object adaptation layer, supporting a LOM extension. This framework is intended to provide users with personalized learning objects, depending on their profiles and locations, adapting multimedia content to the users' devices. It is logically divided into two layers: a multimedia adaptation layer and a learning object adaptation layer. The major components of the framework include the Context Description Manager, Learning Resource Adaptation Engine and Learning Resource Database. The Context Description Manager exchange and manage two types of context information between the learner client and learning content server: learner context description and MPEG-21 DIA context description. The former is the metadata information about the learner including learner profile, learner model and learning settings. The Learning Resource Adaptation Engine consists of two adaptation sub-engines for the adaptation in every layer: Learning Object Adaptation Engine and Multimedia Adaptation Engine. The structures of the subengines are very similar. Every sub-engine comprises two logical modules, Resource Adaptation Engine and Description Adaptation Engine. Resource Adaptation refers to the adaptation of learning object or multimedia content itself, while Description Adaptation refers to the adaptation of metadata information regarding learning objects or multimedia contents. Regarding the Learning Resource Database, there are four databases in our framework to store learning resource and related information. Learning Object Database and Multimedia Object Database are used to store the real contents in the two adaptation layers, while Extended LOM Metadata Database and Digital Item Database are used to store corresponding metadata information of the contents in every layer. Context Description Manager receives context descriptions containing environment constraints and learner preferences from learner client and sends them to Learning Resource Adaptation Engine. Then the adaptation engine determines the optimal adaptation operations according to the given context description and metadata information regarding learning resources. After appropriate adaptations are applied, the contents of the adapted learning resources are delivered to the learner and the adapted metadata information is stored back into related databases.

The objective of the framework described in Tan et al. (2009) is to integrate a context-aware (location-based) system with a learning environment. The proposed system is made of three agents that work and interact in the learning environment: a Learner Agent, a Location-aware Agent, and a Resource Agent. The Learner Agent is responsible for reporting students' location information to the Location-aware Agent, as well as cooperating with the Resource Agent to manage the authentic examples. The architecture consists of an installation agent and three adaptable mobile application computing solutions to accommodate different mobile platforms. To implement the multi-platform adaptation architecture requires MIDP2 and CLDC1.1 ready mobile device. With the multi-platform adaptation architecture, a mobile user need to go to a URL with the mobile web browser and to download the installation agent, a J2ME program that can run on any MIDP2 and CLDC1.1 ready mobile device. The proposed framework uses GPS-based location to add location data to pictures taken from the learners' camera phones. The other contribution of this framework is its integration of the location-based system with a mobile web-based collaborative creation platform where students can share content with their surrounding peers. The system automatically creates groups of students depending on their geographical locations and profiles.

Basaeed proposed an open architecture that facilitates contextualization using current widely used web standards (Basaeed et al., 2007). The architecture consists of three main modules: context sensing (e.g., personal information and personal preferences, devices, and connectivity), context reasoning, and context delivery. Context can be sensed either directly (through direct user interaction) or indirectly (e.g., using HTTP headers, historical information, learner information, device and connectivity type, etc.). The context reasoning module set variables that control the contextualization process based on the context state. The process is achieved by firing a set of rules after resolving conflicts. After successful categorization, learners with the same experience level are delivered the same depth of content. The system offers learning objects (LOM-based) to users depending on their context (e.g., device capabilities). This proposed framework uses a three-tier, web-based architecture. To handle the learning context, the system uses three contextual categories: learner, device, and connectivity. Each category is subdivided to a set of attributes. These attributes are set directly or indirectly to form the current context. Based on the current context, attribute-based and stereotype-based rules are applied to set delivery variables to either a "predefined" or a "collaborative" value. The learning web is constructed to match the various constraints and the user's time limits. Afterward, the context-aware delivery module forms the GUI and delivers the context-aware content to the user. The process of adaptation and delivery is repeated throughout the learning session. The system does not include sensor-based data in the user context, so the level of context-based personalization is limited.

Martin describes an open source framework called M2Learn (Martin et al., 2010), aimed at improving and simplifying the development of future mobile and ubiquitous learning applications. This framework offers an easy and intuitive API that significantly simplifies the use of many services and sensors, encapsulating their complexity. In general terms, the M2Learn framework is designed to "lower the floor", i.e., to make it easier to build new mobile learning applications, and to "raise the ceiling", i.e., to increase the ability of designers to build more sophisticated applications. This dual effect is achieved by transparent sensor data acquisition; by supporting different location technologies, including GPS, cell towers, Wi-Fi, remote identification through RFID, motion recognition with accelerometers, and digital compasses; and by advanced context management. This framework also makes it easier to create communication, collaboration and assessment modules making use of existing services in e-learning platforms such as Moodle. The framework also simplifies the use of learning objects, such as those defined with LOM and IMS-QTI, and provides many other services fostering communication and interoperability between different systems. This framework also aims to centralize the student's e-portfolio into a Moodle server by submitting all learning results and activities to the Moodle submission service. Thus, regardless of the application source where the learning experience was conducted (e.g., games, mobile applications, and web-based tools), the educator can review and grade all of the students' activities within the Moodle platform.

4. Discussion

This section compares the above frameworks and middleware systems, focusing on the different features each one implements. The outcome of this analysis is a complete perspective of the state of the art of frameworks and middleware that facilitate the creation of mobile and ubiquitous learning applications.

4.1. Classification of quantitative data

Based on the previous analysis, Table 1 offers an overview of each framework and middleware system, providing information about the operating system and underlying programming technologies for each system. The table also summarizes the main purpose of the frameworks, fostering a clearer understanding of each one.

Table 2 offers a more detailed perspective on the features implemented by each system. This framework comparison uses a binary scale to avoid subjectivity in determining the level of implementation of each feature. Therefore, if the framework implements the feature, then the cell contains an "×"; otherwise, it is left blank.

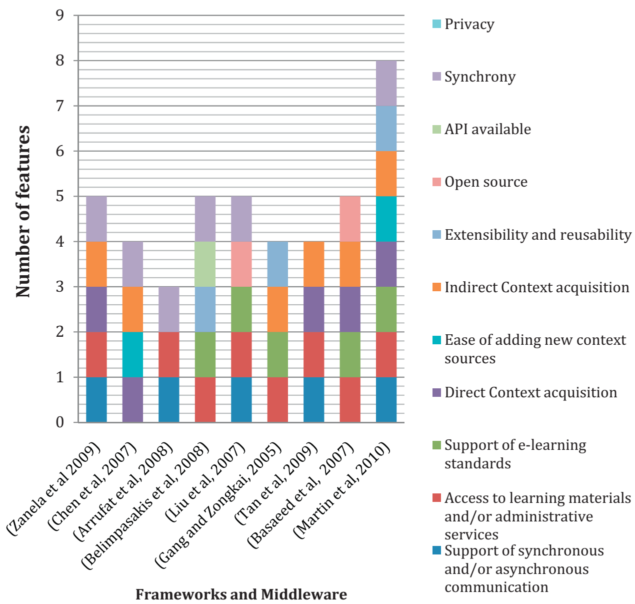

Fig. 1. Features implemented in each framework or middleware.

4.2. Interpretation

Table 2 provides a global view of the various systems and the features they contain, but a deeper analysis is required to determine the differences between architectures for each feature or group of features. For this analysis, related features have been grouped together into different categories for analysis.

One conclusion is that the framework introduced by Martin et al. (2010) is the only framework that supports all of these features. Fig. 1 presents a visual representation of the features implemented in the various frameworks.

4.2.1. Support of synchronous and/or asynchronous communication

Four of the frameworks examined implement communication and collaboration in some way. The framework described by Zanela et al. (2009) provides various interaction tools, such as discussion forums and Skype, but these are in fact just links to external applications that are not integrated into the framework.

Arrufat et al. (2008) offer both synchronous and asynchronous messaging between peers, whereas the approach presented by Liu et al. (2007) only contains synchronous messaging through a chat service. Finally, Tan et al. (2009) describe a collaboration mechanism that allows students to annotate and publish their work so other students can see it.

The framework proposed by Martin et al. (2010) integrates services from e-learning platforms (Moodle at this stage, although it could be easily extended to others). This integration allows for the use of both synchronous tools, such as Moodle chats, and asynchronous features, such as forums and blogs. The use of these technologies allows students to publish their work, ideas, and experiences in the virtual course environment from a mobile device. This not only fosters collaboration and communication, but also compiles a student's e-portfolio in the e-learning platform regardless of the application where the experience took place.

Table 1
Comparison of operating systems, programming languages and main purposes of frameworks and middleware systems for mobile and ubiquitous learning.

Framework and middleware	Operating system	Programming technologies	Main purpose
Zanela et al. (2009)	Windows Mobile	Mobile Client: HTML Server: PHP, SOAP and XML	General purpose
Chen et al. (2007)	None, it is a simulation	None, it is a simulation	Efficient context sharing
Arrufat et al. (2008)	Windows Mobile	Mobile Client: Visual Basic, C++ & C# Server: Java Message Service and Java Naming Directory Interface	Content sharing and social interaction
Belimpasakis et al. (2008)	Symbian OS	Mobile Client: J2ME	Content sharing
Liu et al. (2007)	Windows Mobile	Not specified	Multimedia management
Gang and Zongkai (2005)	Multi-platform	Mobile Client: HTML Server: J2EE, Web Services and XML	Adaptation and delivery of learning contents
Tan et al. (2009)	Symbian OS and Windows Mobile	Mobile Client: J2ME MIDP 2.0	Content sharing and social interaction
Basaeed et al. (2007)	Multi-platform	Mobile Client: HTML Server: PHP, XML and Web Services	General purpose
Martin et al. (2010)	Windows Mobile	Mobile Client: Visual Basic, C++ & C# Server: REST Web Services	General purpose

4.2.2. Direct context acquisition

Regarding the acquisition of data from sensors (e.g., GPS, WiFi, RFID, motion sensors), four of the systems provide location information (i.e., Zanela et al., 2009; Chen et al., 2007; Tan et al., 2009; Martin et al., 2009). However, Zanela's framework does not support other sensors, such as remote identification or motion sensors. GPS and cell tower location data are widely available in many mobile applications. Thus, many frameworks also implement this feature.

Because other technologies, such as motion sensors, are relatively new to the market, they are not supported in most of the systems analyzed here. However, the use of these sensitive technologies can improve, on one hand, the personalization of the services provided, and on the other hand, the engagement of the users in the system through more attractive user interfaces.

Tan et al. (2009) present a research framework that supports the use of various location methods, including RFID identification and QR codes, although the framework is a system that does not yet implement these functionalities. Similarly, the proposal of Chen et al. (2007) also accounted for the possibility of integrating different sensing types, but rather than implementing these technologies, the authors propose that the system built over the framework will carry out the data sensing. However, Zanela's framework (Zanela et al., 2009) only incorporates location-based technologies based on both GPS and Wi-Fi. Finally, the M2Learn framework (Martin et al., 2010) supports various types of location information (GPS, cell towers, and WiFi), RFID identification though RFID, and other sensors such as motion sensors. The availability of different sensing methods makes the mobile learning applications highly versatile systems that can be adapted to almost any environment. Applications can range from the development of location-based applications to the transformation of daily objects into smart objects through RFID or the creation of immersive applications through the use of natural interaction with motion sensors.

4.2.3. Indirect context acquisition

Six of the studied systems provide some kind of indirect context acquisition. Basaeed et al. (2007) describe a certain level of context management, proposing the use of inference rules to manage the context information provided by the student's profile to achieve more advanced context and service personalization. Gang and Zongkai (2005) compiled context including learner profiles, learner models and learning settings. Tan et al. (2009) proposed a system of context management centered on the creation of social groups that depend on students' locations. Zanela's framework (Zanela et al., 2009) used physical location data to obtain symbolic names, allowing the real-time tracking of movements.

Although several systems carry out some kind of context management, only the systems developed by Chen et al. (2007) and Martin et al. (2010) allow new context sources to be added and foster context sharing to support service personalization. In particular, Martin's framework (Martin et al., 2010) allows the addition of new context sources, such as new sensing modules, so that the system can be constantly improved. The compiled context is refined through a set of services that transform location data into postal addresses and symbolic names. This feature allows location data to be related to any service or content. This framework also includes the ContextHub, which allows the users' context to be shared with both users and external applications (supporting mashups).

4.2.4. Extensibility and ease of adding new context sources

An important feature in a mobile and ubiquitous environment is the Service Discovery function, which allows users to find personalized services according to their current environment. The frameworks described by Belimpasakis et al. (2008) and Martin et al. (2010) are the only systems to implement this feature. In particular, the Belimpasakis framework is designed to provide content depending on the user's environment. The M2Learn Service Discovery described by Martin et al. (2010) is implemented through the ContextualServiceDirectory module that allows services, content, and learning objects to be associated with specific locations and times. These features allow for the plug-and-play configuration of environments, one of the main advantages of the M2Learn system. An application based on M2Learn automatically changes its configuration based on modifications in the ContextualServiceDirectory, the RSS, or the contextual information from the ContextHub.

4.2.5. Access to learning materials and/or administrative services

Most frameworks provide some content support because mobile learning developers view content as one of the main pillars of mobile learning, but few enables the combination of content with sensor data to provide a value-added service.

Zanela et al. (2009) integrated videos from YouTube with a file storage system to allow users to share content. Arrufat et al. (2008) focused on photo sharing, whereas Belimpasakis et al. (2008) mainly concentrated on enabling content sharing.

In contrast, Liu et al. (2007) focused on multimedia by implementing both a multimedia player and an engine. Gang and Zongkai (2005) focused on providing content through learning objects, adapting the content delivery to each user profile.

Tan et al. (2009), in contrast, sought to foster the creation of collaborative content through text and pictures. A similar perspective is offered by Basaeed et al. (2007), who proposed a framework that allows text and images to be visualized based on the user's profile.

The framework proposed in Zanela et al. (2009) includes a reminder service that sends e-mail notifications to users. The interoperability of institutional calendars (e.g., academic, enterprise)

Table 2
Comparison of features of frameworks and middleware systems for mobile and ubiquitous learning.

Features	Frameworks								
	Zanela et al. (2009)	Chen et al. (2007)	Arrufat et al. (2008)	Belimpasakis et al. (2008)	Liu et al. (2007)	Gang and Zongkai (2005)	Tan et al. (2009)	Basaeed et al. (2007)	Martin et al. (2010)
Support of synchronous and/or asynchronous communication	×		×		×		×		×
Access to learning materials and/or administrative services	×		×	×	×	×	×	×	×
Support of e-learning standards				×	×	×		×	×
Direct context acquisition	×	×					×		×
Ease of adding new context sources		×							×
Indirect context acquisition	×	×				×	×	×	×
Extensibility and reusability				×		×			×
Open source					×			×	×
API available				×					×
Synchrony	×	×	×	×	×				×
Privacy									
Total implemented features	5	4	3	5	5	4	4	5	8

with scheduling or alert systems could considerably improve personalized self-scheduling, helping to make the user aware of the main events occurring in the area. To enable this type of functionality, the framework proposed by Martin et al. (2010) uses the vCalendar format to connect to external calendars, such as the Moodle calendar, so it is almost trivial to make learning applications aware of course events. This framework also supports the use of any type of media content, with a focus on the use of LOM-based learning objects.

Service personalization can be achieved through the use of mashups that merge the user's context with external services to provide value-added applications. The term mashup usually also includes the use of feed aggregators that personalize the information provided to the user, including different feed formats (i.e., Atom, APnP AV, and WebDAV), and this is only supported in the systems developed by Belimpasakis et al. (2008) and Martin et al. (2010). However, the mashup framework created by Belimpasakis cannot include geographical location as it does not include location information, considerably limiting its potential mashup applications. The M2Learn framework (Martin et al., 2010) supports the easy development of mashup systems through RSS implementation together with the ContextHub module, which distributes the users' context along with the environment in an XML-based format to other mobile applications.

4.2.6. *Support of e-learning standards and reusability*

Support of e-learning standards is fundamental to ensure interoperability and reusability of services and resources between platforms. Only four of the nine systems examined here provide learning object support, although this is a fundamental pillar of learning technologies. Thus, the framework proposed by Belimpasakis et al. (2008) supports Dublin Core based resources, whereas the systems described by Gang and Zongkai (2005), Basaeed et al. (2007), and Martin et al. (2010) support LOM-based objects.

Only the systems proposed by Liu et al. (2007) and Martin et al. (2010) support interoperability with e-learning platforms. Liu's development is implemented using an ad hoc mobile learning platform rather than an existing e-learning platform such as Moodle. Martin's framework provides high-level interoperability by allowing for automatic reuse in the mobile learning application of the learning services used in the e-learning platform, and it also allows the outcome of the mobile application (e.g., log, result, comments, experiences) to be integrated within the e-learning platform. This feature helps to compile the student's entire e-portfolio in the virtual course, regardless of whether the information comes from an activity carried out in the virtual course, from a mobile application, or from a game.

Finally, only three of the studied frameworks provide assessment support. In particular, the framework described by Liu et al. (2007) gives the students access to assessment materials in a database. The framework described by Zanela et al. (2009) contains an evaluation module that provides feedback on the level of competence that the learner has achieved. Martin's framework (Martin et al., 2010) supports assessment through the support of the IMS-QTI standard for questionnaires and test performance.

4.2.7. *Open source and API available*

Only three of the systems (i.e., Liu et al., 2007; Basaeed et al., 2007; Martin et al., 2010) are open source, but neither of the first two provides an API to simplify the development of applications. Therefore, developers could change the code of these open systems, but it would not be easy for them to do so, as they would have to understand all of the existing code to be able to adapt it to their own requirements. The M2Learn framework proposed by Martin et al.

(2010) is the only open source framework that provides an API to facilitate the development of new mobile learning applications.

The other framework that provides an API to simplify the development of applications is the system proposed by Belimpasakis et al. (2008). However, this framework is not open source, so external developers are not allowed to modify the source code according to their requirements.

4.2.8. Synchrony

All the studied systems, except the ones proposed in Gang and Zongkai (2005), Tan et al. (2009) and Basaeed et al. (2007), includes some kind of synchrony in their implementation. However, synchrony is a useful feature for some services, such as chats or media sharing, but no in others, such as remote function calls. The use of synchrony remote function calls, as in the case of the system proposed by Martin et al. (2010), can generate performance problems because of connection delays in the mobile networks. This problem can be avoided using call-backs in the remote calls to functions or methods.

4.2.9. Privacy

Finally, regarding privacy, none of the studied frameworks propose any measure to ensure user's privacy. Probably the main reason is because all of them are research prototypes, and not commercial systems.

4.3. Adoption recommendations

The framework proposed by Zanela et al. (2009) is especially suitable for developing projects that require collaborative features in content delivery and evaluation activities. The main disadvantage of this system is that extensibility and reusability are very limited, because the system just provides access to external tools, such as YouTube or Skype. Thus, it may be a good tool for those Windows-based projects that do not require personalization and are focused on collaborative learning using course content and assessment. In contrast, the M2Learn framework (Martin et al., 2010) allows high extensibility by providing and API, being open source, following learning standards, supporting feeds, and using configurable indirect context sensing.

Regarding picture sharing, the AGORA environment proposed in Arrufat et al. (2008) may be a good candidate for Windows Mobile-based projects that require intensive communication channels, because it is designed to optimize multicast communication performance. The system proposed by Tan et al. (2009) is also focused on pictures sharing among students. This framework is the only one focused on content sharing that incorporates direct context sharing. Its main added-value is its algorithm for creation of groups of students depending on their geographical locations and profiles. Thus, this framework should be chosen for projects with location-aware requirements. In contrast, the "Sharing Middleware" proposed in Belimpasakis et al. (2008) should be used for those projects that interoperability is the most important feature. This framework support a more powerful content sharing using different protocols (i.e., DublinCore, Digital Item Declaration Language Lite, UPnP AVV, ATOM, and WebDav), although no communication tools and sensing are provided. An advantage against the systems described in Arrufat et al. (2008) and Belimpasakis et al. (2008) is the easiness of extending the system and creating high level applications on top of it thanks to the API provided.

The system described in Liu et al. (2007) might be used as Mobile Multimedia Management for Windows Mobile. Its first disadvantage is that it does not make use of the real added-value features of the mobile devices, such as sensing. The second disadvantage is that it uses an own platform, making difficult the interoperability with existing platforms, such as Moodle, Sakay or Blackboard. In fact, some of these well-known Learning Management Systems already support web-based access, i.e., multiplatform, through mobile devices, e.g., Moodle. Thus, most of the researchers might adopt an existing platform with multi-device support instead of using an ad hoc solution.

The system proposed by Gang and Zongkai (2005) neither takes advantage of the location-aware features of mobile devices. It offers a multi-platform adaptation and delivery service for multimedia contents. The use of standards such as MPEG-21 DIA and LOM maximize interoperability with other platforms. The system might be useful for developers that seek to provide personalized multimedia contents to their users. The M2Learn framework of Martin et al. (2010) may be more useful for those project aimed at providing personalized services and contents depending on users' location. However, M2Learn does not support multimedia content sharing.

Chen's simulation framework (Chen et al., 2007) may be used as foundations for further developments in context-aware frameworks that require high efficiency of data delivery in mobile ad hoc networks. Chen et al. propose a real-time environment able to support urgent push-based event dissemination of data to neighboring nodes.

5. Conclusion

The development of frameworks and middleware to simplify the development of mobile and ubiquitous learning applications will accelerate the life cycle of mobile learning projects by helping developers create applications more easily, focusing on learning design and educational outcomes rather than on programming.

The present study of the state of the art of the current systems available for developers provides a comparison of the main functionalities and features implemented in each. This comparison will help mobile learning researchers to decide what technology to use in future developments. A successful choice can reduce project costs related to human resources, as less time will be used to develop and test the project.

In conclusion, our analysis indicates a lack of frameworks and middleware systems devoted to facilitating the creation of mobile and ubiquitous learning applications. Currently most projects start from scratch and implement all of their own functionalities.

In addition, most of the frameworks found do not implement many features or are designed for particular purposes. However, although some frameworks are designed for general purposes and implement a wide range of features and functionalities, they do not offer support for all current mobile platforms and operating systems, i.e., iPhone OS, Android, RIM and Symbian. Although there are some web-based frameworks, their functionalities are very limited because they cannot access sensors.

The final conclusion of this study is that a need exists to more advanced frameworks that encapsulate the complexity involved in dealing with different sensors (not just location) and in learning objects, services, standards and platforms. These frameworks should be created to obtain the best from new mobile devices, such as new Smartphones and Tablets (e.g., iPhone, iPad and Android-based devices), and from new e-learning platforms and services, allowing developers and educators to focus their efforts on creativity, learning design, and learning outcomes instead of on programming complexity.

In fact, as the current mobile market is divided among several important platforms, i.e., iPhone OS, Android, RIM and Symbian, the best approach for these new frameworks is probably to implement these features and functionalities based on HTML5 and Javascript, so that new mobile learning applications built over them will run in any platform for years, regardless of the device and platform. Thus, developers will maximize the potential audience for their projects.

Finally, privacy is a fundamental issue that is not addressed by the current frameworks. Further development must be done in the privacy and security field to create commercial systems that ensure users' rights.

Acknowledgments

The authors would like to acknowledge support from the Spanish Science and Innovation Ministry, Project TIN2008-06083-C03/TSI "s-Labs – Integración de Servicios Abiertos para Laboratorios Remotos y Virtuales Distribuidos," and CYTED-508AC0341, Project "SOLITE-SOFTWARE LIBRE EN TELEFORMACIÓN". Finally the authors want to acknowledge the support provided by e-Madrid Project, S2009/TIC-1650, "Investigación y Desarrollo de tecnologías para el e-learning en la Comunidad de Madrid".

References

Arrufat, M., París, G., García-López, P., 2008. AGORA: an integrated approach for collaboration in MANETs. In: Proceedings of the First International Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications, Innsbruck, Austria.

Baldauf, M., Dustdar, S., Rosenberg, F., 2007. A survey on context-aware systems. International Journal of Ad Hoc and Ubiquitous Computing 2 (4), 263–277.

Basaeed, E.I., Berri, J., Zemerly, M.J., Benlamri, R., 2007. Web-based context-aware m-learning architecture. International Journal of Interactive Mobile Technologies (iJIM) 1 (1), 2007.

Belimpasakis, P., Luoma, J.P., Börzsei, M., 2008. Content sharing middleware for mobile devices. In: Proceedings of the First International Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications, Innsbruck, Austria.

Biegel, G., 2004. A Programming Model for Mobile, Context-Aware Applications. Thesis submitted to the University of Dublin, Trinity College. October 2004.

Biegel, G., Cahill, V., 2004. A framework for developing mobile, context-aware applications. In: Proceedings of Second IEEE Conference on Pervasive Computing and Communications (Percom) 2004, Orlando, FL, March14–17.

Bormida, G., 2002. MOBIlearn Description of Work, Last accessed: May 2002, URL: http://www.mobilearn.org.

Boticki, I., Looi, C.-K., Wong, L.-H., 2009. Doing collaboration and learning fractions with mobile devices. In: The 14th Global Chinese Conference on Computers in Education (GCCCE 2010), NTU, Singapore.

Bottazzi, D., Montanari, R., Toninelli, A., 2007. Context-aware middleware for anytime, anywhere social networks. Proceeding on the IEEE International Conference on Intelligent Systems 22 (5), 23–32.

Chen, T.-S., Yu, G.-J., Chen, H.-J., 2007. A framework of mobile context management for supporting context-aware environments in mobile ad hoc networks. In: Proceedings of the 2007 International Conference on Wireless Communications and Mobile Computing, Honolulu, Hawaii, USA.

Cheung, B., Stewart, B., McGreal, R., 2006. Going mobile with Moodle: first steps. In: IADIS International Conference on Mobile Learning.

Conde, M., Casany, M.J., Alier, M., García, F.J., 2009. Back and forth: from the LMS to the mobile device. A SOA approach. In: IADIS International Conference Mobile Learning 2009, Barcelona, Spain.

Davidyuk, O., Riekki, J., Rautio, V.-M., Sun, J., 2004. Context-aware middleware for mobile multimedia applications. In: Proceedings of the 3rd International Conference on Mobile and Ubiquitous Multimedia, Maryland, ISBN:1-58113-981-0.

Dey, A.K., Salber, D., Abowd, G.D., 2002. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. In: Moran, T.P., Dourish, P. (Eds.), Context-Aware Computing: A Special Triple Issue of Human–Computer Interaction. Lawrence-Erlbaum, March.

Fahy, P., Clarke, S., 2004. CASS – middleware for mobile context-aware applications. In: Proceeding on the Second International Conference on Mobile Systems, Applications and Services, Boston, MA, USA, June 6.

Gang, Z., Zongkai, Y., 2005. Learning resource adaptation and delivery framework for mobile learning. In: 35th ASEE/IEEE Frontiers in Education Conference, Indianapolis, USA, October 9–22.

Georgieva, E., Smrikarov, A., Georgiev, T., 2005. A General classification of mobile learning systems. In: International Conference on Computer Systems and Technologies (CompSysTech' 2005).

Gu, T., Punga, H.K., Zhang, D.K., 2005. A service-oriented middleware for building context-aware services. Journal of Network and Computer Applications 28, 1–18.

Gupta, A., Kalra, A., Boston, D., Borcea, C., 2009. MobiSoC: a middleware for mobile social computing applications. Mobile Network Application 14 (November), 35–52.

Halasz, F.G., Schwartz, M., 1994. The Dexter hypertext reference model. Communications of the ACM 37 (2), 30–39.

Hansen, F., Bouvin, N., 2009. Mobile learning in context—context-aware hypermedia in the wild. International Journal of Interactive Mobile Technologies (iJIM) 3 (1), 2009.

Hofer, T., Pichler, M., Leonhartsberger, G., 2002. Hydrogen. Context —a framework to support context-awareness on mobile devices. Technical Report.

Jarvis, J., 2009. What would google do? Collins Business.

Johnson, L., Levine, A., Smith, R., 2010. The 2010 Horizon Report. The New Media Consortium, Austin, Texas.

Keegan, D., Stoyanov, S., Kicken, W., Nadeem, D., 2009. Instructional design guidelines. In: Mobile Performance Support System for Vocation Education and Training (mPSS) Project.

Lavin-Mera, P., Torrente, J., Moreno-Ger, P., Vallejo Pinto, J.A., Fernández-Manjón, B., 2009. Mobile Game Development for Multiple Devices in Education. International Journal of Emerging Technologies in Learning iJET 4 (s2), 19–26.

Liestøl, G., 2009. Situated simulations: a prototyped augmented reality genre for learning on the iPhone. International Journal of Interactive Mobile Technologies (iJIM) 3.

Liu, Y., Li, R., Xu, C., Yu, F., 2007. Design and application of multimedia mobile learning framework. In: Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, Hong Kong, 19–22 August.

Martin, S., Gil, R., Sancristobal, E., Diaz, G., Castro, M., Peire, J., Milev, M., Mileva, N., 2009. Middleware for the development of context-aware applications inside m-Learning: Connecting e-learning to the mobile world. Fourth International Multi-Conference on Computing in the Global Information Technology, pp. 217–220 (6). 23-29 August 2009, Cannes (France).

Martin, S., Gil, R., Sancristobal, E., Diaz, G., Castro, M., Peire, J., Milev, M., Mileva, N., 2010. Enabling innovations in mobile-learning: a context-aware and service-based middleware. International Journal on Advances in Software. IARIA 3 (1–2).

Naismith, L., Lonsdale, P., Vavoula, G., Sharples, M., 2004. Report 11: Literature review in mobile technologies and learning. FUTURELAB SERIES. Accessed on-line on June 25, 2011. URL: http://archive.futurelab.org.uk/resources/documents/lit_reviews/Mobile_Review.pdf.

Reich, S., Wiil, U.K., Nürnberg, P.J., Davis, H.C., Grønbæk, K., Anderson, K.M., Millard, D.E., Haake, J., 1999. Addressing interoperability in open hypermedia: the design of the open hypermedia protocol. The New Review of Hypermedia and Multimedia 5, 207–248.

Ryan, N.S., Pascoe, J., Morse, D.R.,1999. FieldNote: a handheld information system for the field. In: Proc. TeleGeo'99, 1st International Workshop on TeleGeoProcessing. Claude Bernard University of Lyon, pp. 156–163.

Schmidt, A., Forbess, J., 1999. What GPS doesn't tell you: determining one's context with low-level sensors. In: Proceedings of the Sixth IEEE International Conference on Electronics, Circuits and Systems, Paphos, Cyprus, September 5–8.

Shriram, R., Sugumaran, V., 2007. Adaptive middleware architecture for information sharing on mobile phones. In: Proceedings of the 2007 ACM Symposium on Applied Computing, Seoul, Korea.

Stavroulaki, V., Demestichas, K., Adamopoulou, E., Demestichas, P., 2006. Distributed web-based management framework for ambient reconfigurable services in the intelligent environment. Mobile Networks and Applications (11), 889–900.

Tan, Q., Kinshuk, Kuo, Y.-H., Jeng, Y.-L., Wu, P.-H., Huang, Y.-M., Liu, T.-C., Chang, M., 2009. Location-based adaptive mobile learning research framework and topics. In: Proceeding on the 2009 International Conference on Computational Science and Engineering, Vancouver, Canada, August.

Wiil, U.K., Hicks, D.L., Nürnberg, P.J., 2001. Multiple open services: a new approach to service provision in open hypermedia systems. In: Davis et al. (Eds.), Proceedings of the 2001 ACM Hypertext Conference, pp. 83–92.

Yau, J.Y.-K., Joy, M., April 2008. A context-aware personalized m-learning application based on m-learning preferences. In: 2010 6th IEEE International Conference on Wireless, Mobile, and Ubiquitous Technologies in Education, pp. 11–18.

Zanela, A., Kich, M., Schlemmer, E., 2009. A framework for the design of ubiquitous learning applications. In: Proceedings of the 42nd Hawaii International Conference on System Sciences.

Sergio Martin is assistant professor at the Electrical and Computer Engineering Department (DIEEC) of the Spanish University for Distance Education (UNED). His research interests include mobile, ubiquitous and pervasive computing, especially focusing on its applications in education, as well as improving the teaching of electronics through the use of new technologies. Martin has a Ph.D. from UNED and is Computer Engineer.

Manuel Castro is full professor at the Electrical and Computer Engineering Department (DIEEC) of the Spanish University for Distance Education (UNED). His research interests include learning-enhanced technologies, specially focusing on engineering education. Castro has a Ph.D. in industrial engineering from UNED.