Journal of Universal Computer Science, vol. 19, no. 7 (2013), 950-964
submitted: 31/8/12, accepted: 31/1/13, appeared: 1/4/13 © J.UCS

Interactive Design System for Schools using Cloud Computing

Habib M. Fardoun, Bassam Zafar, Abdulrahman H. Altalhi
(Faculty of Computing and Information Technology,
King Abdulaziz University, Jeddah, Saudi Arabia
{hfardoun, bzafar, ahaltalhi}@kau.edu.sa)

Antonio Paules
(University of Castilla-La Mancha, Castilla-La Mancha, Spain
apcipres@gmail.com)

Abstract: The design of an educational system involves a good understanding of the whole school environment in order to find the correct approach to develop a comprehensive educational system that will meet real educational needs in their operation. This article describes a design model for an educational system based on the teaching methods applied in the Spanish classrooms, which takes into account new advances in technology, while preserving the current teaching methods in the classroom to ensure a quality teaching and learning process. This development has been achieved by combining technological components such as Cloud Computing, Web Services and Distributed User Interfaces. The proposed system is based on a systematic approach where different phases are implemented, containing workflows and stages.

Keywords: CSchool, Interactive Design, Cloud Computing, Cloud Computing Architecture, Web Services, Software Engineering Architecture, Educational Systems, Distributed User Interfaces, Quality User Interfaces, ISO 9126-1
Categories: D.2.2, D.2.10, D.4.6, D.4.7, H.3.4, H.4.3, H.5.2, K.3.1, K.4.3, L.2.0, L.2.1, L.2.2, L.2.3, L.3.0, L.3.6

1 Introduction

The design and implementation of learning platforms is essential for the development and future prospects of information technology, communications and knowledge management within the teaching and learning process in order to preserve their quality. School users, students, tutors and families all require an agile educative system in this respect, which is, in turn, capable of storing the large volumes of information managed during this process.

This is the catalyst that reveals an urgent need to develop an effective, high-quality, easy-to-use teaching and learning that meets users' needs, by combining new technologies. Moreover, our system ensures quality use for user satisfaction. Thus, we have decided to follow ISO 9126-1 [ISO, 2001], for its development and approval, since, from our point of view, even though the system complexity increases, its use does not necessary become more difficult and complicated if taking into account all the supported factors when designing and creating the system, such as: methods of work in the classroom, teacher-student and tutor-student relationships, etc.

It is noteworthy that the research exposed in this paper is the continuation of a long-term research project that we started more than five years ago, with the creation of two educational applications for schools: TabletNET [Paules, 2009] and eLearniXML [Fardoun, 2011]. We have combined these with other technologies that we have worked with in educational environments: namely, Cloud Computing, Web Services and Distributed User Interfaces; these were used to finally create a new complete educational system: CSchool.

This paper consists of eight sections: having started with the introduction, in section two we will present previous work related to this research and a brief literature review of the technologies used in the development of the system, presenting the project idea in response to the need for such a system in schools. The third section will examine the system design and development, starting with its architecture and functionalities, the design of its user interfaces and its support of distributed user interfaces. In the fourth section, we will present the quality process followed while developing and deploying our system. The fifth section will present a brief description of Cloud Computing within Educational systems, and in the section six we will discuss how we have applied this in the construction of CSchool. Finally, we finish with our conclusions (section seven) and ideas for future work (section eight).

2 Project Definition and Necessity

From an educational point of view, schools need to be supported by tools to act in accordance with the curriculum materials, in order to develop better collaborative work by allowing teachers to better organize the teaching materials to be presented, using Information and Communication Technologies (ICTs) [Garrigós, 2011; Padilla, 2011].

Taking into account the status of the School 2.0 system [Ministerio de Ciencia e Innovación, 2011], it is possible to highlight the question: What are the real problems that teachers face daily in the classroom? This is the question that guided our research work presented in this paper. Therefore, solving these problems and that of how to make use ICTs in the classroom became our two main keys. The classroom environment, in which our system will be applied, is as follows: in addition to the equipment in the classroom, one tablet per student; a tablet for the teacher; a printer, projector and port replicator and, finally, a wireless connection for the TabletsPC. Analysing how schools are organized, we found different types of centre that combine several courses with groups of different levels, as well as single-track centres (just in one town) and others that are distributed across several populations, such as those of the CRA (rural schools) [Muñoz-Repiso, 2010]. Our first research step was to study the evolution of ICTs in the educational environments. After conducting several surveys and reviewing European and American statistics, we found that the use of ICTs has been constantly growing in the classroom, but with numerous existing problems, which have come to light alongside this growth. Some important weaknesses have been found, are the following:

- Low participation: between all the educational centres less than 8% of projects make use of the Internet.

- Only 26% of the communication with the parents of the students is made via the Internet.
- Only 30.6% of collaborative work utilises online resources.
- Only 32% of the evaluation of students is made using ICTs.
- Communication with the families has very weak percentage.

Now that these problems have being revealed, through many studies and research in Spanish centres [e.g. Tomai, 2010], we can see that there is a real need for such application that facilitates solving these problems, in order to support the educational environment. But, because of the huge magnitude of the project and the multiple requirements that we need to meet, we found that we needed to take into consideration several characteristics, such as the flexibility of the system and interoperability when new functionalities are incorporated and making the system accessible from other systems. These needs, led us to make use of Web Services, in order to achieve our goal, and to develop a new user interface for the system, to ensure that it is easy to use (learnability) and can be easily understood by the new users (e.g. parents). These features identified have been developed following various aspects of a quality model, which is contained in International Standard ISO 9126-1 [ISO, 2001].

As a result, the focus of designing and developing the system was, on first creating a bridge of communication with the students' parents and then facilitating teachers' work in the classroom as well as enhancing collaborative work between students.

3 CSchool Interactive Design System

In this section we present and identify the elements that support the CSchool system, starting with its architecture, functionalities and user interfaces.

3.1 Architecture and Functionalities

While designing the system architecture, we took into consideration the fact that the educational content and the students' activities are hosted in Web content managers. These Web services provided us with the basis for the interaction between the users and the system content. In any case, the system must have an architecture that allows access to it from any device, as well as a central secure system, with authentication, that allows the distribution and management of the content, providing flexibility by Web services access.

The system's functions are employed through the use of Web services. These Web services are the points of input and output of the system, facilitating communication between the different parts of the architecture. What we greatly valued was how communication between the various different Web services can benefit from the Web services communication in a banking environment [Broberg, 2009], so we considered how we could adapt this to our needs in an educational and informational system.

The main parts of the system can be defined as a cloud of services, some of which are:

- Storage logic: providing the logistics for storage of the system files.
- Schools: providing the business logic.
- Services: providing access to the system and generating accurate views of information from the other clouds.
- Administration: providing the necessary data for the country's education department to establish the student's curricula, enriched by the data received from the educative environment or "Schools" cloud.
- Evaluation: providing the tool for evaluation.

The structure of this architecture, [see Fig. 1], allows for new services to be deployed (or added), to facilitate its evolution and growth; also, it leads to a better communication between the different services, such as the information traffic, thanks to the support that Web services technology provides it At the end of this study, we noted the need to accommodate the business logic in separate layers. Thus, in the future, we may choose to add new layers to the system or to allow other systems to access the existing one.

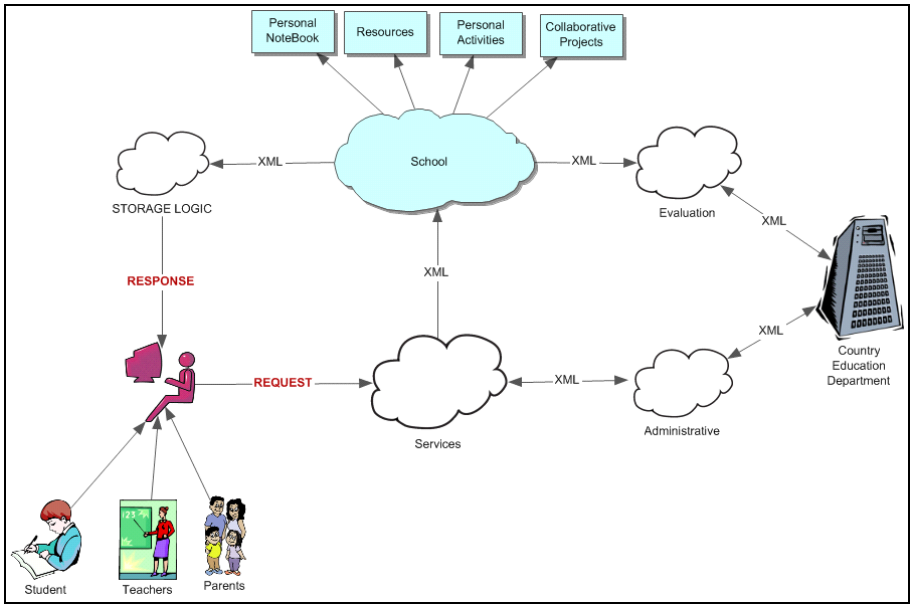

Figure 1: Shows the location of the cloud of services between the different parts of the system, to obtain a dynamic system.

3.2 User Interface

Each educational system must be spread across several applications to support the users' diversity; in our case, the principle users are: Teacher, Student and Parent.

Starting with the teacher application, we found that teachers must have access to an easy and dynamic, drag and drop, application with an amicable graphic interface,

which contains all the necessary elements, and which could be extended for the inclusion of future features, depending on their needs [Fardoun, 2011]. Also, it must provide teachers with a final view for the development of the activity, from the teacher's point of view as well as the student's, in order to facilitate the development and implementation of this activity [Wany, 2003]. In this section, we present some catches of the desktop application, [see Fig. 2], showing the main interface that members of the educational community will see, while using CSchool. Special attention is given to the interfaces used by teachers and students.

Figure 2: This figure shows the two steps of the teacher interface, where the teacher can first select a number of elements to compose an activity for teaching and, secondly, where he can view the appearance of the student interface with the elements he has selected for him/her.

While creating the learning activities in real life, teachers have all the options offered by their respective environments. To some extent, one of the weaknesses of computer systems is that it is not easy to emulate the combination of resources available. To this end, in CSchool we offer teachers the option of creating activities, for which they can choose all the objects involved from a text editor, to a discussion tool, presentations, a calendar, and son on. More detailed information about the system design and work can be found in [Fardoun, 2011; Paules, 2009].

During the graphical user interface development, we took into account the new academic resources available, and the need to incorporate new users (students and teachers) coming into the school. For this, the interface is designed so that users can obtain as simple and homogeneous a working environment as possible. Another point that was taken into account was the need for collaborative work between teachers and students. The graphical icons representing an action are identical to both teacher and student interfaces. At the same time, a distinction between the user interfaces for different user levels was achieved, so that the teacher can classify activities depending on their educational level. For this, we have created different interfaces for downloading, browsing and sending documents: [see Fig. 3].

Another feature of the system is that it allows students and teachers to send their files using a private email account and to access these files online, in the same way, as shown in [Fig. 3], thanks to our use of the Cloud and Web Services. This makes it possible for the system to work as a multi-server application, which can access

several servers in different physical locations. It is at this point where it is determined which type of configuration each centre has.

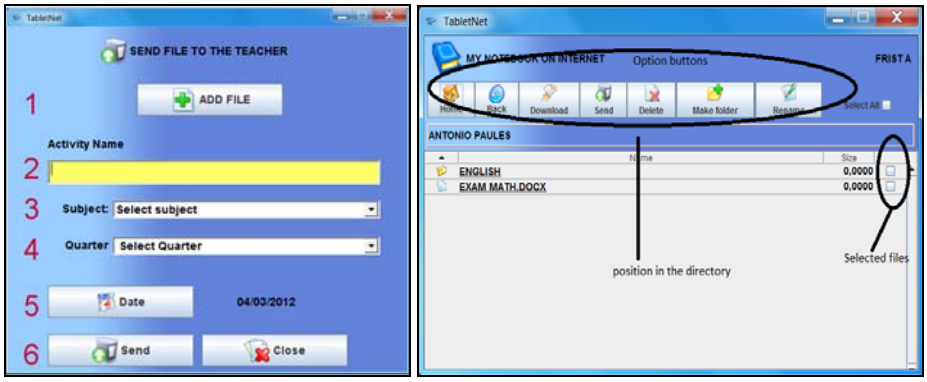

Figure 3: These interfaces are for uploading and downloading files: the first catch shows uploading documents sequentially on the same screen. In the second, the teacher can view the options available for navigation between the documents previously uploaded to the system.

3.3 CSchool: Distributed User Interface Support

The project scope and its location makes it necessary to include a distributed user interface support, in order to obtain a quality system that takes into account the use of new technologies, mobile devices and the use of multiple devices at the same time, on the cloud, so that users can interact with the application in a dynamic way. In this scenario, the cloud provides the communication channel users make use of to share information in real time. To serve this purpose, we have created specific services in the CSchool system, to establish communication between users, decisions as to which interfaces and the how the information will be distributed between several user interfaces are managed in the cloud before being displayed in the user's device. Applying Model Driven Architecture (MDA) [Wany, 2003] we have developed these interfaces and all the resources to support Distributed User Interfaces (DUI), and shared reading (viewing) and writing (editing) properties [Fardoun, 2012].

We have defined the functionality of the interface, as shown in [Fig. 4], so that the user can select whether he wants to share his screen, with an option to read-write properties, for including other users. The parties identified in the interface design are as follows: (A) indicates the menu options, which list actions such as to share items, and to add read and/or write properties. By managing part (A), the user interface will preview these actions in the other interface elements [(B), (C), (D) and (E)]. These DUI features within CSCloud, will facilitate collaborative work for students and teachers as well as facilitating the involving of parents in the educational process, by permitting them to interact with the scheduled sessions organised by the teacher. It also allows the creation of curriculum materials and activities preconfigured for inclusion in teaching units.

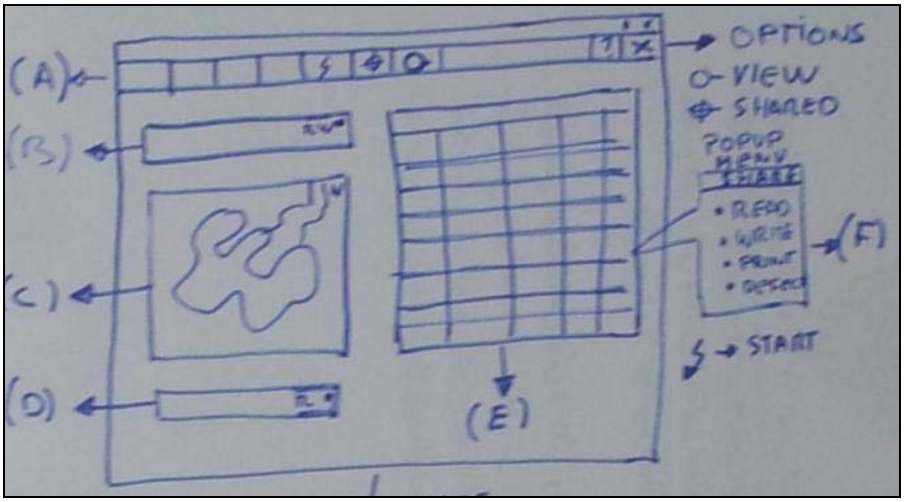

Figure 4: Architecture of CSchool: the interface of the application is divided into several sections. These sections are divided towards distribution between several devices to provide the means of communication and collaboration.

4 CSchool Quality Support

The software quality often marks the success or failure of systems in general, and computer systems in particular; we have been very conscious of this issue, during the development and deployment of CSchool. This section outlines the points that have been followed for the system development and implementation according to the ISO 9126 standard [ISO 9126, 2012].

- Functionality: a set of attributes is related to a set of functions and their properties. These functions are those that satisfy stated or implied needs;
- Usability: a set of attributes related to the effort required to use and individual assessment of such use by a set of users;
- Maintainability: a set of attributes associated with the facility to extend, modify or correct errors in a software system, and
- Portability: attributes related to the ability of a software system to be ported from one platform to another.

4.1 Functionality

The CSchool system meets the specified requirements, as the mapping functions of the system match those of the standard: it also meets the suitability sub-feature by using analysis tools and integrating the required design tools. On the other hand, interoperability is assured, by designing the communication with the server using a FTP protocol, RFC 0959 [Postel, 1985] and a web protocol. By performing user authentication, utilising a username and password so that the system can take into account the role of each user which allows them to access different parts, we can

prevent unauthorized access and thus comply with the safety standard. However, we have tried to facilitate, so far as we can, the system maintenance and also provide easy access to the users.

The exactitude sub-feature does not require any precise analysis, because there have been no calculations nor the need for any calculations performance to be done on the system's functionality. However, the documents downloading and delivery have been established as critical points, in order to verify the system's proper operation and performance. In terms of functionality compliance, the users (teachers and students) have participated in the discussion and the requirements acceptance process.

4.2 Easy Maintenance

The application development using Swing components, as well as and the design of personalized components allows us a quick and effective expandability and ensures the stability of the facilities while performing any changes. Using the appropriate tools will facilitate the analysis stage where discussion requirements tools are applied near to design ones. As evidence, once the system switches from a previous version, after testing by the developers it is the users (teachers and students) who indicate whether there are any errors in the updated system. The development performance, using prototypes, and the components development favour the maintainability and the expansion of the application over the time. At the same time, this will facilitate the development of future expansion.

4.3 Portability

The use of a Java Web Start Centre facilitates the deployment of the application and gives us the ability to restore previous versions (using the version control). Version control enables the deployment of the application update, including on the virtual machine. One of the main features of applications developed in Java is their portability, making it easy to use the system in different operating systems. Therefore, we can say that portability is ensured depending on the compliance programming language used. Portability was tested with positive results, in the following environments: Windows XP, Windows7, MacOsX, etc.

4.4 Usability

The user-centred design, as shown in [Fig. 5], emphasizes the designer's appreciation of the importance of understanding the application context of use. To achieve this, it was necessary to have further knowledge of the school environment and the behaviour within the schools for which we are developing this system. In addition, this participatory design process involved the participation of the end users (teachers and students) in the verification and development of the design.

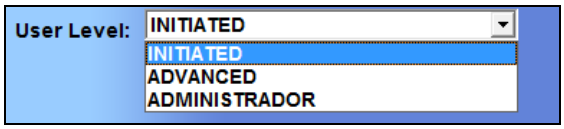

Figure 5: The application provides for different levels of users, with a number of different features for each, giving them more freedom.

Based on Jacob Nielsen's work [Nielsen, 2012], on the development of graphical user interfaces (GUIs), the following points were taken into account:

1. Visibility of system status: even if the application is active, there is not necessarily any communication with the server; it only communicates with the server when it performs the task of sending and receiving files. If an error occurs during connection to the server, the user is notified by a modal dialog box, leaving the screen where the error occurred blocked until the user accepts the message (to accept is the message assumed that it can be read).
2. User control and freedom: the user interacts directly with the interface. The same pattern has been followed in all system interfaces, giving the user total control, which is facilitated by using the buttons "leave" and "back".
3. Consistency and standards: the consistency of the interface contributes to its growth, as the user uses the knowledge gained in one application with other applications. The use of soft colours and the possibility to change its appearance using themes allows the programmer to adapt the application to institutional standards: in this way, consistency can be maintained with all applications in the institutions.
4. Error Prevention: the best treatment is to prevent errors; we have tried to achieve this by good design of the dialogue and actions that the user performs in the system, minimizing the risk of the appearance of errors. If a user commits an error in the system, he/she will be advised by a modal dialog box indicating the action to take for the process to continue successfully. For example: "Advice message: authentication error, check the key".
5. Correspondence between the system and the real world: the system is fully expressed in the language of its users; the use of familiar words from the scholarly environments helps users to understand actions in the application context: for example, use of the terms "my notebook", "classmates" and so on.
6. Recognise before remembering: the use of icons for menu items and using special colours helps the user to associate an icon with its functionality. The terminology that identifies the icon represents the action of the icon. The choice and placement of icons in both applications is not accidental because the teacher may need to tell students the options and for that reason, the icons in all applications are the same.
7. Flexibility and efficiency of use: the system is flexible, with different user levels for teachers and students, related to their level of knowledge.

8. Aesthetic and minimalist design: the design of the interface has no redundant elements or information irrelevant to the user and the aesthetics are maintained.
9. Help users to recognize and diagnose errors: all errors that may occur in the application are validated. The application verifies that no error can occur, for example, by checking required fields and the numbering of fields when the user completes a form.
10. Help and documentation: documentation and help manuals have been written for users and technical staff; teachers have produced a student handbook. A faculty expert in the use of ICT in education imparts information to the teachers.

By using ISO 9126 in the development of CSchool system requirements, we have brought clarity of definition to both purpose and operating capability. This also allowed us to make trade-offs between two or more characteristics during the design of the system.

5 Education by means of cloud computing

Cloud computing is a new phenomenon linked to Web 2.0. The origin of the term refers to the collection of different services stored on servers that users can access through the Internet. That is, the user has access to different software, applications and files stored in different, undefined, or virtual places (this is the reason why the term cloud is used), which are permanently available to users wherever they are [Gens, 2008]. Documents are not physically hosted on the user's computer; they can be retrieved from anywhere via an Internet connection. So, working in the cloud means that users don't need to depend on a particular program or even a specific operating system any more. The only requirement to start working in the cloud is a device with an Internet connection.

5.1 Cloud Computing Applied for Education

The great advantage of "The Cloud" is information sharing in a real way. Using a local area network, multiple users can work on the same file; but it can be more interactive, as in the following scenario. Several students share a document online; they can simultaneously view exercises, they can view and edit homework and also organise information shared in a presentation using data from different sources [Berrocoso, 2009]. Activities like this are supported by a collaborative and communicative environment, which is hosted on the cloud. Our educational cloud is composed of information and academic process data; those are linked together with other Web 2.0 applications to support students' interaction, managing their own academic process within the application. In our ISE Research group, we have designed a new platform, called CSchool. CSchool takes advantage of all these resources to create an adequate educational environment for students and teachers, to improve teaching and learning processes in secondary schools by exploiting Cloud services and distributed user interfaces.

6 CSchool Services

The cloud provides accommodation for the users' interaction, facilitated by CSchool services using distributed user interfaces; this service in the cloud supports user devices in an invisible way. Since schools are the context for implementation of our system, we initially studied the ways in which CSchool services can reflect actual school organization. Then, we identified the actors as user roles, in addition to creating scenarios according to these actors' fields and the distributed interfaces in which they are screened. The school schedule is specified following a procedure established by law for management and academic organizations [Delgado, 2011]. By accompanying this with qualification, evaluation, monitoring, communication and support by distributed user interfaces, we are actually able to define the most important CSchool services as follows [see Fig. 6]:

- Program Services: providing the necessary structure for the teacher to conduct an educational activity within the specific framework established by national education law.
- Services for teaching units: providing the structure for developing teaching units and organising these to be accessible via distributed user interfaces.
- Services for collaborative learning: collaborative learning tools or applications that encompass all the needs distributed across multiple user interfaces.
- Service for learning languages: distributing multimedia character reinforcement, audio and video for students and teachers across several user interfaces.
- Communication services for parents: schools and teachers can communicate with parents or tutors, in one-way communication; this communication is developed to support the distributed user interface.
- Distributed user interfaces: supports Web Services 2.0 on the cloud.

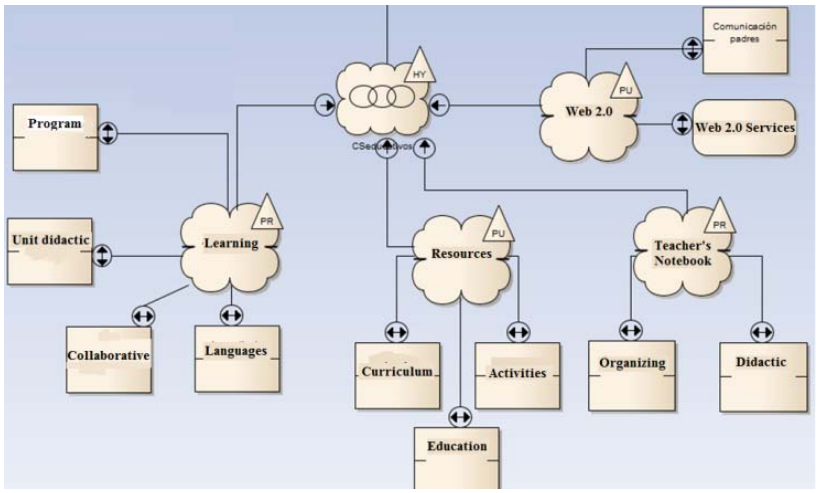

Figure 6: CSchool Services grouped by Clouds

All aforementioned services are grouped in different clouds, as well as being simultaneously grouped according to the scope of the services from a specific educational viewpoint. They can be hosted in public or private clouds but also in hybrid clouds, designed to support services for external entities. [Fig. 7] depicts a cloud structure [SOMF, 2011] following the methodology by which services are grouped by IAAS [Djemame, 2011], according to which each cloud offers a full service.

6.1 CSchool Services Design

For the CSchool conceptual model and system architecture we have created two distinct parts: first, we defined the different pricing types and architecture to cover the different possibilities of cloud services mentioned above; secondly, we considered the conceptual architecture of the distributed user interface, allowing the system users to establish collaborative relationship.

6.2 CSchool Charging Architecture

The charging architecture [Marti, 2006] defines the different types of pricing proposed for this system as follows: prepaid charging, bid rate charging; constant charging variable charging, and so on. An example is presented for the ways the CSchool charging scheme might operate, based on accessibility levels to files and data, [see Fig. 7].

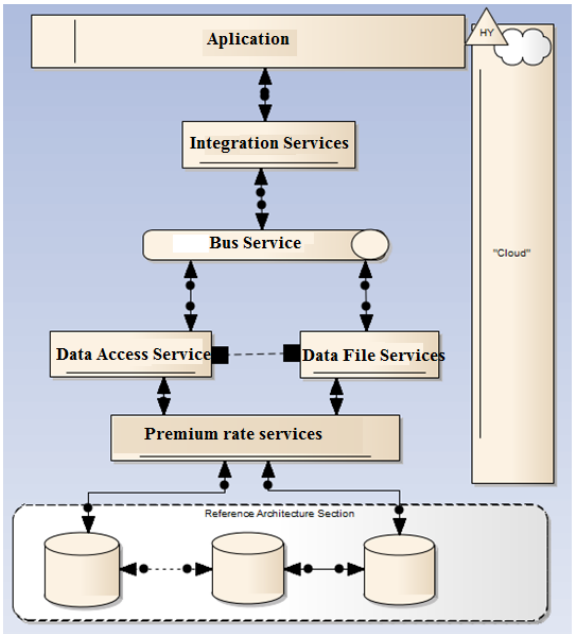

Figure 7: CSchool Charging Service

Elements under the CSchool charging are:

- System Users: constant charging.
- Files hosting: charging for lodging.
- Reprographics: constant charging.
- Postal sending: bidding charging.
- Multimedia services: video and audio stream.
- Activities and resources for creating lesson plans.

In order to set and store the CSchool charging system and to consult it, XML language is used in all steps carried out in the cloud. The XML charging system gives us the possibility to obtain a detailed bill grouped by the resource's type of charging, as well as providing all relevant details to the customer.

7 Conclusions

In this research work we have presented a quality new educational system. Its new features suggest it to be a coherent and a innovative system to be used by the scholastic users (teachers, students, parents, etc.). It combines many new technologies: such as Cloud computing and Web services to better distribute and access the information and knowledge, to be easy and dynamic to work with. It was also developed to support distributed user interfaces, to support the ease of use and interaction with the system. By following the ISO 9126-1 guidelines, we ensure and proved the quality of the system, its compliance to all suggested quality heuristics without losing the effectiveness of the traditional learning/teaching process. Thus our main achievement, is the creation of an effective and complete educational system, that schools can work with

Also, we conclude that the integration of cloud services within the educational systems ensures flexibility and dynamism in the classroom in preparing the learning activities and during the lessons. We discuss the idea of integrating of existing diverse systems, enhancing their security and usability by providing all the services with the same key. However, creating separate cloud systems for different services provides the system with total technical independence, achieving greater maintainability, even when updating and extending these services, which guarantee that the system operates even during maintenance.

8 Future Work

For the future, we are working on the installation of this system within different schools in Spain, to further test it and obtain complete and professional feedback that will can help us to improve the system in its future versions. On the other hand, we are working in parallel with these results, to create a secondary application that will allow teachers and tutors to control student's application in the classroom and at home, by utilising distributed user interfaces, to obtain greater control over the student's own learning and production.

The use of ICT in education continuously suggests new patterns to appear in the ever-evolving relationship between teachers and students in the classroom sessions. These patterns are currently creating great impact on the development of applications that support DUI (Distributed User Interfaces), which increase services that support cooperative work, such as the Cloud. We must also take into account the development of the application for cloud management in educational environments, in two particular aspects: management of services and improved integration of cloud systems in new applications, which may arise from the inclusion of new services, and the combination of existing cloud services. We are also studding to cover the current technological advances we found in mobile devices, tablets, IDPs, etc., as it is important to fully incorporate them into the current educational process and as a part of the educational content.

References

[Berrocoso, 2009] Berrocoso, J. V., Garrido, M. & Sosa, M. J. (2009). Políticas educativas para la integración de las TIC en Extremadura y sus efectos sobre la innovación didáctica y el proceso enseñanza-aprendizaje: la percepción del profesorado. Universidad de Extremadura. Facultad de Formación del Profesorado. Departamento de Ciencias de la Educación.

[Broberg, 2009] Broberg, J., Buyya, R. & Tari, Z. (2009). Creating a 'Cloud Storage' Mashup for High Performance, Low Cost Content Delivery. Service-Oriented Computing – ICSOC 2008 Workshops. Lecture Notes in Computer Science, 2009, Volume 5472/2009, (pp. 178-183), DOI: 10.1007/978-3-642-01247-1_17

[Delgado, 2011] Delgado, M. L. (2011). *Organización de centros educativos: modelos emergergentes*. La Muralla, ISBN: 9788471337979

[Djemame, 2011] Djemame, K., Padgett, J., Gourlay, I. & Armstrong, D. (2011). *Brokering of risk-aware service level agreements* in grids School of Computing, University of Leeds.

[Fardoun, 2011] Fardoun, H. M. (2011). *ElearniXML: towards a model-based approach for the development of e-learning systems*. PhD Thesis, University Castilla-La Mancha

[Fardoun, 2012] Fardoun, H. M., Paules, A. & Alghazzawi, D. M. (2012). "*CSchool: DUI for Educational System using Clouds*". Proceedings of the 2nd Workshop on Distributed User Interfaces: Collaboration and Usability. Austin, Texas, USA, May 05, 2012. (pp. 35-38). ISBN-10: 84-695-3318-5.

[Garrigós, 2011] Garrigós, H. P. & Jaume, M. J. R. (2011). *Tres años de experiencia cooperativa semi-presencial con el alumnado a través de las TIC* (Tecnologías de la Información y la Comunicación). Filología Española, Lingüística General y Teoría de la Literatura; Sociología I. Universidad de Alicante.

[Gens, 2008] Gens, F. (2008). *Defining "Cloud Services" and "Cloud Computing"*. Retrieved on August 22, 2010 from http://blogs.idc.com/ie/?p=190.

[ISO 9126, 2012] ISO 9126-1 (2012) *Software engineering – Product Quality*. Quality model.

[ISO, 2001] ISO 9126 (2001). *Software engineering – Product Quality*. Retrieved in June, 2012 from http://www.iso.org/iso/home.htm

[Marti, 2006] Marti, A. B. & Ibáñez, J. A. (2006). *Modelo de tarificación de servicios para redes móviles de siguiente generación*. Departamento de Ingeniería Telemática, Universidad Politécnica de Cataluña

[Ministerio de Ciencia e Innovación, 2011] Ministerio de Ciencia e Innovación. (2011). *Proyecto de investigación Las políticas de un «ordenador por niño» en España. Visiones y prácticas del profesorado ante el programa escuela 2.0. Un análisis comparado entre comunidades autónomas*

[Muñoz-Repiso, 2010] Muñoz-Repiso, A. G.V. & Tejedor, F. J. T. (2010). *Evaluación de procesos de innovación escolar basados en el uso de las TIC desarrollados en la Comunidad de Castilla y León*. Universidad de Salamanca. España.

[Nielsen, 2012] Nielsen, J. (2012). *Evidence-Based User Experience Research, Training, and Consulting*. Retrieved in September, 2012 from http://www.useit.com

[Padilla, 2011] Padilla, G. E. S., Leal, F. R. & Portes, C. F. (2011). *La elaboración de objetos de aprendizaje para la incorporación de las Tecnologías de la Información y Comunicación en las prácticas educativas*. Dirección General de Innovación Tecnológica, Universidad Autónoma de Tamaulipas

[Paules, 2009] Paules, A., Fardoun, H. M. & Isarre, J. R. (2009). *Gestión De Aula En Centros Educativos*. Retrieved from http://tabletnet.linkate.es/

[Postel, 1985] Postel, J. & Reynolds, J. (1985 Oct.). RFC 0959 File Transfer Protocol (FTP). Retrieved in June, 2012 from http://www.rfc-es.org/rfc/rfc0959-es.txt

[SOMF, 2011] SOMF 2.1. (2011) *Service-Oriented Discovery and Analysis Model Specifications*. Methodologies Corporation. Retrieved from http://www.sparxsystems.com/somf

[Tomai, 2010] Tomai, M., Rosa, V., Mebane, M. E., D'Acunti, A., Benedetti, M. & Francescato, D. (2010). *Virtual communities in schools as tools to promote social capital with high schools students*. Comput. Educ. 54, 1 (January 2010), (pp. 265-274). DOI=10.1016/j.compedu.2009.08.009.

[Wany, 2003] H. (2003) *MDA-based Development of E-Learning System*. Retrieved from http://csdl.computer.org/