

Contributions to Using IT in Education: An Educational Video Player

Ciprian CUCU¹, Marian-Pompiliu CRISTESCU², Corina-Ioana CRISTESCU³

¹“1 Decembrie 1918” University, Alba Iulia, Romania

²“Lucian Blaga” University, Sibiu, Romania

³Academy of Economic Studies, Bucharest, Romania

cucu.ciprian@gmail.com, mp_cristescu@yahoo.com, ci_cristescu@yahoo.com

Following the trends of IT use in education we propose four main principles of educational application development: anytime-anywhere access (including access for mobile devices); user-generated content; assigning users an active role inside the platform; creating a correlation between natural activities and educational activities – adding fun into the equation. With these principles in mind we developed a small Educational Player – a movie player that connects to a database to show information relevant to the events in the movie, more specific to the exact key-frame (in the movie time-line) at the moment of the request. Though the application is still in a conceptual form, its possibilities make us confident in its future success either as an informal way of learning, or by using it in a formal educational context either as an e-learning tool, either as interactivity tool in a traditional classroom setting.

Keywords: *E-learning, Educational Technologies, Multimedia Applications*

1 Introduction: general concepts and video teaching

IT in Education: Development Trends

The development of Information Technologies and the increase in access to such technologies, led an increase in their current use in the learning process. Considering the application and implementation of IT in education, such technologies can be divided into two categories: technologies used in the classroom and technologies used in the distance education activities.

In the case of the first category, the tendency is to move from office software (PowerPoint presentations) and the methodology of transposing classical courses in an electronic format towards complex, interactive applications that would offer functionalities such as [1]: *demonstration* (the professor controls the content shown), *monitoring* (the professor monitors the students' activities) and *communication* (student-to-student or student-to-professor inside the application framework).

When considering distance education, tools and techniques used have developed in the last decade from mailing courses and video-

conferences to complete integrated online platforms. The main trends can be summarized as follows:

- The development of m-Learning (Mobile Learning) that implies designing access to educational platforms for mobile devices. M-Learning can offer important advantages, such as increasing of student motivation or increasing monitoring capabilities, as show by Georgiev et al in [2]. This trend became relevant due to the extending of mobile devices (laptop, notebook, mobile phone, PDA) usage [3].
- E-learning 2.0: considered as an application of Web 2.0 principles in the field of educational technologies. One of the effect of moving towards e-learning 2.0 is the fact that students become an active component of the educational process: they no longer receive resources, study them and get evaluated in the traditional sense, but can involve themselves in publishing contents, can collaborate and support each other, while evaluation tends to be done by constant monitoring of their involvement and contributions. Moreover, by using

technologies such as blogs, wikis, podcasts and RSS feeds, the students are no longer limited to discussing predefined subjects with their classmates, but can interact on a wide area of subjects with interested parties all over the world. [4]

- Virtual educational platforms: This trend developed by the furthering of game-learning theories. Kurt Squire [5] noticed the attraction of individuals towards complex games that imply a profound cognitive activity, and are not a sum of simple decisions. Another education model proposed by Barab et al [6] proposes combining games with learning, insisting on the concept of adding to the learning process of a component that is in general missing: *the fun*. Moving forward from these theories, virtual worlds have appeared and are being used in education, such as *Second Life* or *Quest-Atlantis*, where students can interact with objects around them, with other students and can create content for others to use.

Summarizing the trends mentioned above, it is safe to say the most important principles of educational technologies evolution, follow four main principles:

- anytime-anywhere access (including access for mobile devices);
- user-generated content;
- assigning users an active role inside the platform;
- creating a correlation between natural activities and educational activities – adding *fun* into the equation.

Using video technologies in the educational process

Starting in the late '80s, early '90s, the theory that learners should be involved in doing domain-related activities, not just receive the results of someone else's activities as summarized in texts or heard in lectures grew constantly, supported by various studies and research [7]. Their conclusions led to supporting "experiential learning", considering that learning involves real-world participation [8] and "inquiry-based

learning" which supports learning as an inquiry process and not one of memorization [9].

On another side, some of these works aimed at finding reasons for the lack of motivation (some) children / students have shown for the educational process and concluded that "kids don't like school is not that the work is too hard, but that it is utterly boring" [10].

Living in the digital age, the individual is subjected to various media, multimedia and marketing stimuli thus making it easy to get distracted and / or bored with typical learning approaches. The answer, for most researchers in the field [11] can be found in the entertainment industry – an industry that already proved it can engage, challenge and "capture" the individual.

Most of the work done in this area, related to two possibilities: either using computer games, either using videos as educational tools. This paper, proposes the development of the latter and bases its hypotheses on Baggett's finding [12] that information obtained visually is more memorable. In addition, the result for retention supports Jonassen, Peck, and Wilson's [13] argument that stories presented via video can also help learners easily remembers the content in comparison with expository materials. On the basis of the qualitative data analysis, the learners perceived that the video-based instruction was more memorable than the text-based instruction. This could be because representations derived from both auditory and visual symbol systems can be better for building mental models of the situation than representations based on only linguistic information [12]. Attention is one of the major influences on motivation that learners should have for successful learning [14] and in these studies, there was a significant difference in learners' attention between video-based instruction and traditional text-based instruction. This result implies that video-based instruction can effectively be used to motivate learners by attracting their attention.

As a conclusion to these [15], video-based instruction can be an effective method to

enhance learners' retention in context-based learning.

2 Concept and usability

Considering the principles presented above, we began developing a software application that aims to implement and develop several information technologies to bring together a natural / fun activity (watching a movie) with the possibility of delivering information that is related to the video content being played.

The main research challenge we propose is showing that a simple plug-in that brings up resources related to a specific video file and a specific key-frame in the video can have an educational effect consisting in – increased attention, retention and collaboration. To prove this, empirical studies have to be conducted on a fully-functional prototype build on the proposed concept and design. Although a first step in development has been accomplished (sending parameters from the player to a web server), to answer usability issues, the main contributions of this paper are the definition of the conceptual model, the design of the application (as a mock-up) and the presentation of the database architecture and of the technologies used.

By using the proposed application, the user views the movie file in *streaming* – a methodology that increases in popularity with the development of high-speed Internet connections and video websites such as Mega Video, ShowTime and others. While viewing the desired film, the user can require further information related to the contents of that particular film, this being one of the main features of the application: the association between the information and a specific key-frame in the video file. The time frame in which information can be added is of one second, thus for a one hour film, 3600 key-frame positions would be available. Also, the information added can be split up into categories, giving the users the specific information they require.

Developing the player concept imposed in the beginning of the process an usability issue, consisting in a choice between the possibilities to display the user-required

information:

- the first option would be to use the caption-type system already implemented in some streaming video-players usually used to display advertisements, subtitles or meta-information; this option has the advantage of being fully implemented and easy to configure, yet a much bigger disadvantage: the information is limited to text and small graphics and the content has to fit in the size of the player windows (considering some users might use the player in windowed form and not full screen);
- the second option would be to simply use the actual web page that hosts the player window to interact with the player, request the video being played and the current key-frame and simply make database requests to get the correct content. This option offers the advantage of using simple web development techniques and offers a lot of space to display several types of information, while the player is still available. The disadvantage however, is still in the usability area: showing the player while the user is viewing the requested information is confusing; also if the player is in full-screen, none of the functionalities will be available. Another downside of this method is that it implies the use of two different systems (the player and the web page) that communicate, but are distinct;
- the third option, that tries to take the benefits of both options mentioned above, is to make use of the **modal window** concept, defined in user interface design as a child window that requires the user to interact with it before they can return to parent application, thus preventing the flow on the application main window [16]. In this case, the request for information can be done from the player window itself (using a button similar to the caption or the volume buttons) and the response can be provided in a full-screen, fluid-

width (resolution dependent) overlay modal window (similar to the “Light Box” model [17] [18]). This option offers the following advantages:

The call towards an informational object (information associated to a specific video file and a specific key-frame) is done from the player window itself, available in window form or also in full-screen mode; the button could also be hidden and shown on mouse-over only to further enhance the user experience by creating an unobtrusive application.

The modal window provides enough space to host a large amount of information of various types and sources; since its underlying implementation is actually an HTML file, it would be able to display text, images or other movie clips acquired from one or several different sources through the use of specific programming tools (such as PHP and JavaScript).

The modal window also will provide focus to the requested information, no other content being available until the window is closed; also, to respond to usability issues, the modal window will not be similar to a pop-up window and not open a secondary browser window or page, but display the information in an **overlay** – above the current page, where the player window resides. The

overlay window can be easily closed on a specific key press (e.g. the “esc” key), or by using a simulated close button in the actual overlay (a picture designed to close the window).

The “EDU-player” is thus proposed to be a typical streaming player (based on the existing JW Player – an extensible open source application) loaded in any browser window; the “EDU” functionality will be added through a plug-in that will connect the player itself to the resources database (database interaction and proposed database structure is detailed in the next section).

Figure 1 shows how the player is usually positioned in the browser window (item 3 on the mock-up), among general web-page content such as page heading (item 1) sidebar content (item 2) and others (item 5).

The controls of the player (item no. 4) will be only available on the “mouse over” event, to better comply with the purpose of building an unobtrusive, user-friendly application. The “EDU” plug-in will be available through the plus button on the control panel of the player. Its main function is to load the overlay modal window (figure two) with two parameters: movie file name and key-frame position. This can be easily implemented using the JW Player API (described in section 4 of this paper).

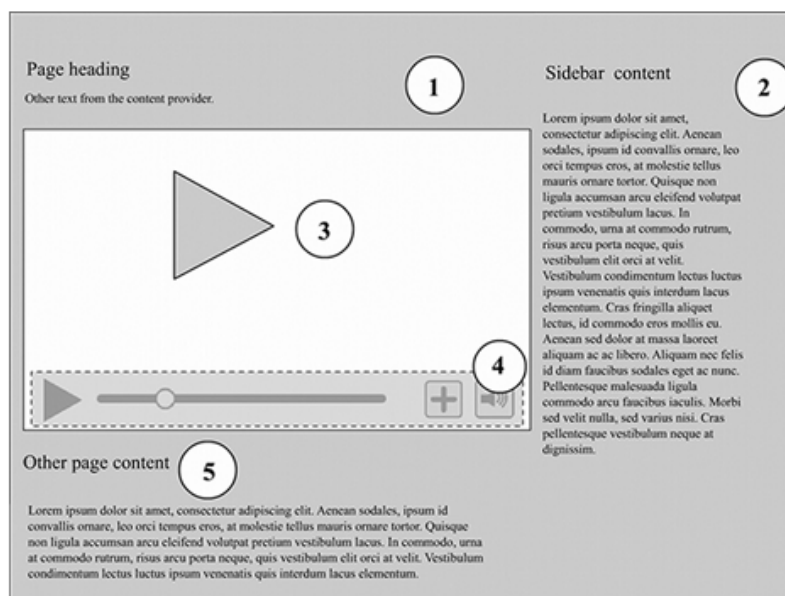


Fig. 1. Web page with EDU-player window

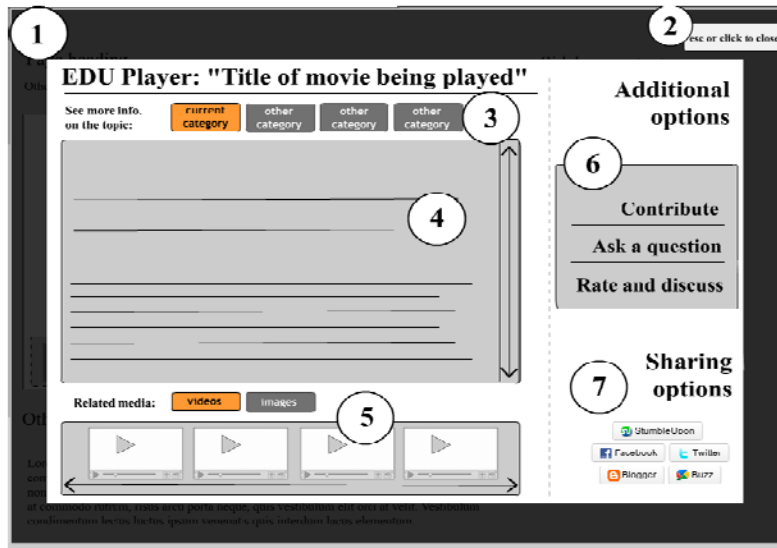


Fig. 2. Overlay window with key-frame related resources

The overlay modal window, shown in figure 2 consists of:

- Opaque (semi-transparent) overlay covering the main window (the browser / player window); the opacity effect is important in suggesting the link between the modal window as an application dependent on the main window.
- Close button with information “esc to close” - to provide the user with two frequently used options to close the modal window and revert to the main window.
- Section control area: the educational resources displayed can be organized into different categories and shown on request.
- Main display area: will show the resources associated with the file and the current key-frame; the HTML format will allow text information as well as images or audio data.
- Related media area: will display images and videos from various sources related to the concepts contained in the current key-frame. These images and videos can be loaded from the “EDU” server itself, or can be any URL embedded in the modal window.
- Additional options: the application has to answer also to the web 2.0 requirements – mainly user-editable contents. This will allow a better interaction between users that can contribute to the educational resources provided (figure 3), can ask questions (figure 4) or can discuss and rate a specific video (figure 5).
- Sharing options: the web is a social network, thus sharing the information or video through the main existing social networking websites is a desirable feature of the application.

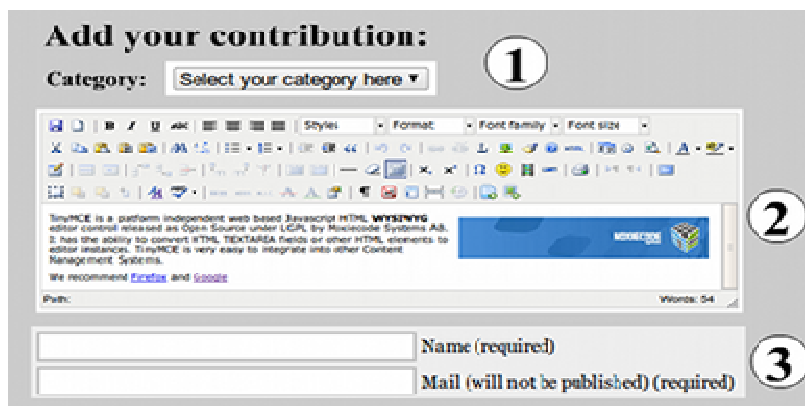


Fig. 3. Adding a user contribution

Choosing the contribute option hides the information display area (sections 3 and 4 from figure no. 2) and shows the contribution interface consisting of: 1. category selection (the category to which the content will be added); 2. rich text area based on the “TinyMCE” [19] or similar WYSIWYG editor that allows formatted text, images,

links and other types of content; 3. User information – though a user registration system would be a interesting feature to consider in the future, we believe that a blog-like user information system that links the contribution to a name and an e-mail is better in terms of usability.

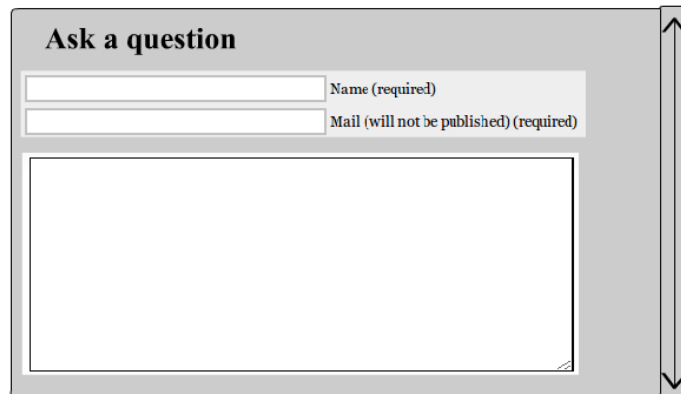


Fig. 4. Ask a question - screen

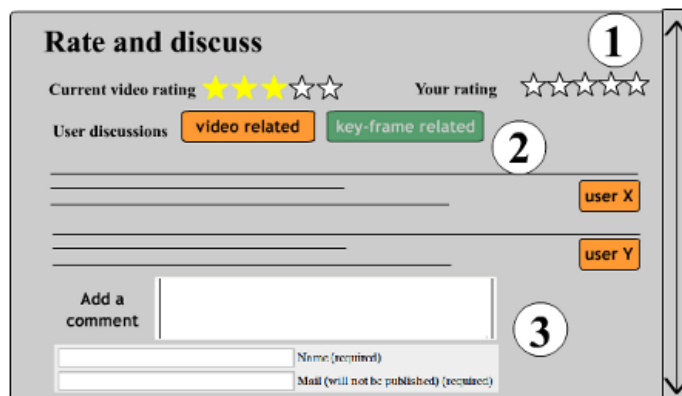


Fig. 5. Rate and discuss - screen

The “Ask a question” screen (figure 4) is a simple implementation of a contact form. Its purpose is to provide a simple communication interface between the user and the system's administrator. The answer will be sent by e-mail.

Figure 5 shows the “Rate and discuss” screen. Its importance is also an usability issue, putting the basis of an indexing system of the videos and specific key-frames in the video that underline the main concepts.

Thus further search possibilities will include searching highest rated videos or key-frames in videos, as well as most commented videos

/ key-frames.

The screen shows the average rating as well as the users' current rating in a usual visual format (section 1 of the screen). Section 2 allows the user to choose what type of comments should be displayed or added (related to the video file or the specific key-frame) and section 3 allows the user to add his own comment.

The proposed concept is aimed at two main different usage types:

- in an informal context: individuals using this video player can benefit from the information as a way to find out more

about an interesting subject without having to search the Internet (just by one click at a certain time).

- in a formal learning context: the EDU-Player can be added as a module to an existing e-learning platform (such as Moodle), where it can be used to give students information in a more interactive fashion, as well as allow for interesting assignments such as contributing to the information in the database related to the proposed film. The students' activity could easily be monitored and their contributions could be evaluated both in quantity and quality.

3 Architecture and data structure

In figure 6 we propose the general architecture of the application that shows how its components interact to produce the desired result. Its main components are the media player with the “EDU” plug-in, the overlay modal window, a web server

containing an AJAX module and a PHP script for database interaction and a MySQL database server.

When the user activates the “EDU” plug-in – by clicking the plus icon on the player controller – two actions are performed (step 1): the overlay modal window is opened and the parameters (movie file name and key-frame) are sent to the AJAX module. The AJAX module forwards the data to the PHP script (2) that requests relevant information from the database (3) and sends the content back to the AJAX module (4). In the last step, the content is filled in the overlay window by the AJAX module.

A secondary scenario is a user added contribution / question / rating; the content, entered in the overlay window will also be sent to the AJAX module (1b) and processed by the PHP script to be inserted (or update existing content) in the database. The inner architecture of the media player used is described in section 4.

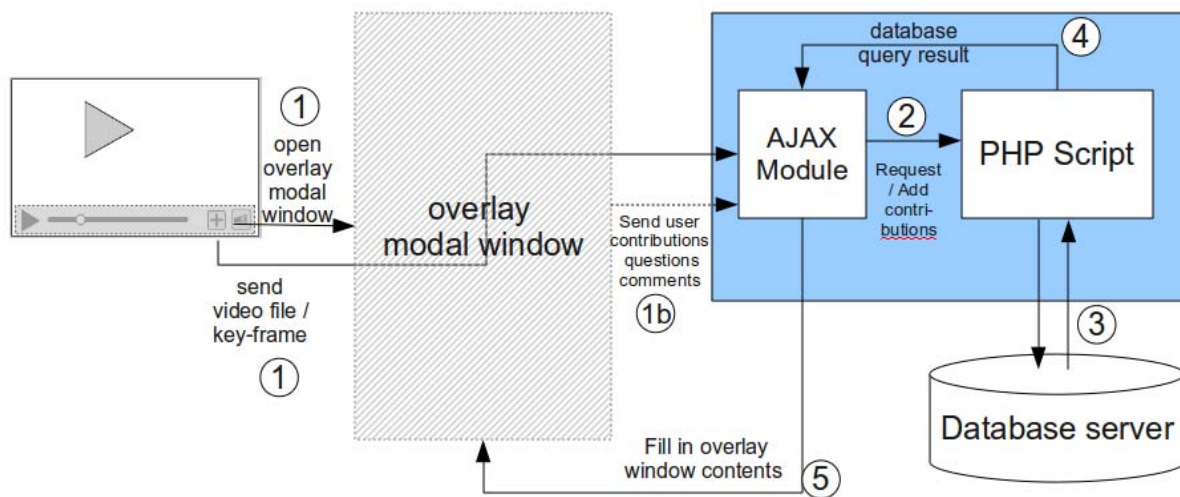


Fig. 6. Application general architecture

Regarding the database structure, we propose a simple relationship-based database schema, illustrated in figure 7. The main table – *contributions* – records the actual information related to a specific video file and a specific key frame; it also records the date / time of each contribution, allows

definition and use of tags to improve data organization. The relationships between tables are standard one-to-many, each video being able to have multiple contributions, each contribution being associated to one or more categories.

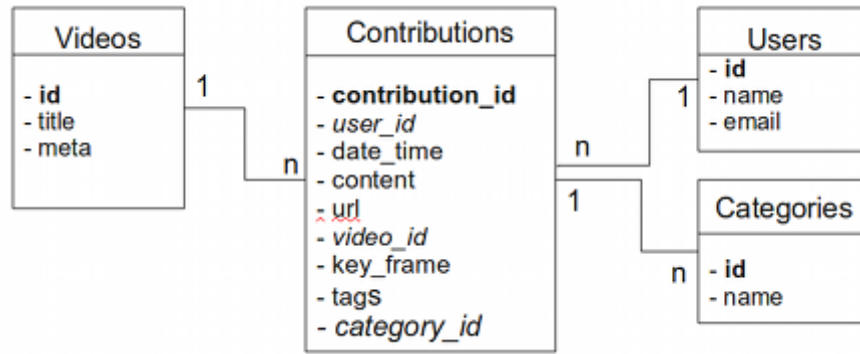


Fig. 7. Database schema

4 Technologies needed and implementation constraints

To implement this project, several technologies are available. In our first approach we used the JW FLV Media Player 4.5 for movie playback, MySQL for database management, PHP for implementing a pseudo-streaming server and to send queries to the MySQL server and JavaScript / AJAX to control the embedded player and request database information without page refresh.

JW FLV Media Player 4.5 is the latest version of the player developed by Long Tail Video [20] and allows the playback of the

most common Internet audio-video formats: FLV, MP4 and MP3. JW player integrates with a streaming system – that allows seeking even in parts that have not yet been downloaded.

Functionalities of the JW player include playing YouTube videos, the ability to play from a predefined play list (in an XML format), the ability to control the player with JavaScript and to extend the player using plug-ins (developed in Action Script). The conceptual model of the player is shown in Figure 1 and described as follows [21]:

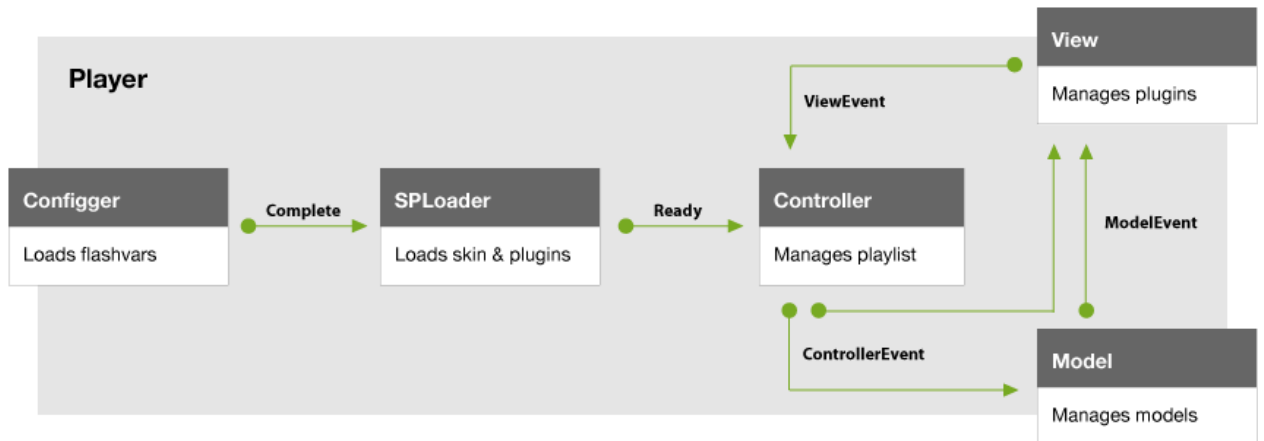


Fig. 8. Conceptual model of the JW Player [21]

The player is developed based on the well-known MVC architecture (Model-View-Controller) – in this case the Model component manages the playback of the videos, the View component manages user interface (control bar, display, play list) as well as external plug-ins and JavaScript API and the **Controller** checks and forwards all directive from the **View** to the **Model** and

manages the play list. Besides the MVC triad, the JW player uses another two modules – Configger and SP Loader. The Configger loads standard flash settings that refer to the video file being played (e.g. author, title, length, streaming type), to the interface and player layout (e.g. width, height) or to the player's behavior (e.g. full screen, volume, and auto-start). The SP

Loader configures plug-ins that is to be loaded into the player and allow the loading of a customized skin (if one is present).

The main important functionality in the context of implementing the Edu-Player plug-in is the JavaScript interface that allows player communication and control. Thus two sets of function are defined – functions that get events from the player (listeners) and functions that send events to the player. Both categories can be set for any of the three main modules: model, view and controller.

In order to get information from the player, the player object has to be initialized and then listeners have to be set up using the *addControllerListener*, *addModelListener* or *addViewListener* methods. Each module has a set of predefined events that can trigger a user-defined function. One possible implementation uses the TIME event, belonging to the controller module, event that triggers the desired function at each 1/10 of a second:

- player initialization:

```
player=
window.document[thePlayer.id];
```
- setting listener:

```
player.addModelListener("TIME",
"positionListener");
```
- user-defined function

```
function positionListener(obj)
{
currentPosition = obj.position;
}
```

The TIME event triggers at every 1/10th of a second and thus updates the value of the *currentposition* variable. This variable forwards its value – the current key-frame – to the AJAX script that will retrieve the correspondent information from the database. In order to implement the project using the JW player, the video files have to be in one of the recognized formats. This implies the conversion of regular video files to the recommended FLV format, which can be accomplished using **ffmpeg**: e.g. `ffmpeg -y -i tudors.avi -ar 44100 -f flv tudors.flv`.

Http PHP Pseudo-streaming: Xmoov PHP

Using streaming as a method of movie file

playback brings a series of advantages, such as content protection or the possibility to seek in the file, even to sections that have not yet loaded. Dedicated streaming solutions (FMS, Wowza, Red5) offer these functionalities but involve a more complex implementation process, which makes *pseudo-streaming* solutions more interesting. Xmoov PHP – a PHP script running on an Apache web server is such a solution, which offers [22]:

- Random access: allows users random immediate access to parts of the file that have not been yet downloaded.
- Bandwidth throttling: allows configurations of the bandwidth by fixed or dynamic dimensioning.
- Cache memory control: allows or forbids storing of video files in the cache of client computers.
- Rapid configuration: minimal settings for standard usage; works with most Apache & PHP configurations.

The JW Player – xmoov connection is also very easy to set up, by setting the flash variable *streamer* to the server location of the xmoov script (e.g. `http://localhost/eduplayer/xmoov/xmoov.php`). The only secondary requirement of the xmoov script is the presence of meta-data in the video file, which can be accomplished using special scripts. For the proposed example we used YAMDI – Yet Another MetaData Injector, which adds to the flv file the following meta-data: creator, metadatacreator, hasKeyframes, hasVideo, hasAudio, hasMetaData, canSeekToEnd, duration, datasize, videosize, videocodecid, audiosize, audiocodecid, audiosamplerate, audiosamplesize, stereo, filesize, lasttimestamp, lastkeyframetimestamp, lastkeyframelocation, keyframes (filepositions, times), width, height, framerate, videodatarate, audiodatarate. The command used to inject metadata with YAMDI is as follows: `./yamdi -i tudors.flv -o tudors_meta.flv`

Ajax-PHP-MySQL

To display the information corresponding to a certain key-frame without refreshing the page where the player is loaded, we used a typical AJAX script, that sends a

HttpRequest to a PHP file and loads the answer (the information from the database) into the InnerHtml property of a predefined DIV. The reversed process, by which the users can add information to the database is implemented through a simple php-mysql application: the *currentposition* value is sent from javascript through the url and the php script gets the value (the player position) by using the GET super-global php variable.

5 Advantages, limitations and future development

Among the advantages this application could offer, we would like to underline:

- combining a natural / fun activity with the delivery of educational resources to help increase motivation and interest towards information, and the involvement of individuals in their own learning process;
- the possibility of using the application both in an informal and in a formal educational context by adding it as a module to an existing e-learning application;
- the possibility users have to contribute to the development of the knowledge base, following the principles of collaborative learning and e-learning 2.0;
- all the software components used are open source, adding benefits such as flexibility, modularity, accessibility and portability.

The application's main limitations are:

- limitations in the educational areas in which it can be used, technical subject being harder to integrate;
- the risk of losing the educational effect to the dramatic effect and the attraction of the actual video – similar to the risk associated with using computer games as educational tools [5]

To enhance the positive effects and diminish its limitations, the project can be developed further, by:

- integration of existing data sources in its database (e.g. Wikipedia)
- developing different user roles to better manage the large quantity of information

and to allow user monitoring (especially in the case of using the application in a formal context).

6 Conclusions

In the field of educational technologies, there are many approaches on how to implement various platforms that deliver educational content, allow for student interaction (collaborative learning) and evaluate the students' learned abilities and knowledge. Most of these approaches are limited to converting regular course material to courses suited for electronic delivery and evaluate based on application of standard traditional tests using the computer – a process called by several e-learning practitioners *the dump'n'pump* method. [23]

One of the new approaches in relation to educational technologies, tries to add fun into the process of learning by using applications such as computer games as educational platforms or, as we propose, using movies as a way to deliver educational content.

Our approach proposes the hypothesis that educators can benefit from the students' inherent curiosity by allowing them to choose the interesting aspects presented in a video and either request more information or contribute to the information already stored regarding those particular aspects. We believe such an application could be used in an informal setting (students using it at home) but also that it could easily be integrated into an existing e-learning platform as part of the educational process and would help motivate the students to play a more active part in their education.

References

- [1] H. F. Vahlensieck, *Teaching in a Computer Lab*, Switzerland 2005, Available at: http://www.genevalogic.com/fileadmin/rektion/whitepapers/Teaching_in_a_Computer_Lab_US.pdf, Accessed 11 september 2009.
- [2] T. Georgiev, E. Georgieva and A. Smrikarov, "M-Learning - a New Stage of E-Learning," *Proceedings of the*

- International Conference on Computer Systems and Technologies – CompSysTech'2004*, 2004.
- [3] C. A. Savill-Smith and P. Kent, *The use of palmtop computers for learning: a review of the literature*, quoted by Jill Attewell, Carol Savill-Smith in *Mobile learning and social inclusion: focusing on learners and learning*, in “Learning with mobile devices. Research and development”, Learning and Skills Development Agency, London, 2004.
- [4] S. Downes, *E-learning 2.0*, Available at <http://elearnmag.org/subpage.cfm?section=articles&article=29-1>, Accessed 10 september 2009.
- [5] K. Squire, “Changing the game: What happens when video games enter the classroom?”, *Innovate*, Vol. 1, No. 6, The Fischler School of Education and Human Services at Nova Southeastern University, Available at: <http://www.innovateonline.info/index.php?view=article&id=82>, Accessed 10 september 2009.
- [6] S. A. Barab, M. Thomas, T. Dodge, R. Carteaux and H. Tuzun, “Making learning fun: Quest Atlantis, a game without guns,” *Educational Technology Research and Development*, Vol. 53, No. 1, pp. 86-108, 2005.
- [7] J. D. Bransford, A. L. Brown and R. Cocking, *How People Learn*, National Academy Press, Washington, 1999.
- [8] D. A. Kolb, *Experiential Learning: experience as the source of learning and development*, New Jersey: Prentice-Hall, 1984.
- [9] S. A. Barab, K. E. Hay, M. G. Barnett and K. Squire, “Constructing virtual worlds: Tracing the historical development of learner practices / understandings,” *Cognition and Instruction*, Vol. 19, No. 1, 2001, pp. 47-94;
- [10] S. Papert, “Does Easy Do It? Children, Games, and Learning,” *Game Developer Magazine*, 1998.
- [11] K. Squire, “Cultural framing of computer/video games,” *Int. J. Comput. Gaming Res.*, Vol. 2, No. 1, 2002, Available at: <http://gamestudies.org/0102/squire>.
- [12] P. Baggett, “Role of temporal overlap of visual and auditory material in forming dual media associations,” *Journal of Educational Psychology*, Vol. 76, No. 3, 1984, pp. 408–417.
- [13] D. H. Jonassen, K. L. Peck and B. G. Wilson, *Learning with technology: A constructivist perspective*. Upper Saddle River, NJ: Prentice Hall, 1999.
- [14] J. M. Keller, *Use of the ARCS model of motivation in teacher training*. ERIC, ED 288520, 1983.
- [15] H. J. Choi and S. D. Johnson, “The Effect of Context-Based Video Instruction on Learning and Motivation in Online Courses,” *The American Journal of Distance Education*, Vol. 19, No. 4, pp. 215–227, Lawrence Erlbaum Associates, Inc., 2005.
- [16] J. Raskin, *The Humane Interface: New Directions for Designing Interactive Systems*. Addison-Wesley Publishing, 2000.
- [17] <http://www.huddletogether.com/projects/lightbox/>
- [18] <http://particletree.com/features/lightbox-gone-wild/>
- [19] <http://tinymce.moxiecode.com/index.php>
- [20] <http://www.longtailvideo.com/players/jw-flv-player/>
- [21] <http://developer.longtailvideo.com/trac/wiki/FlashOverview>
- [22] <http://xmoov.com/xmoov-php/>
- [23] <http://moodle.org/mod/forum/discuss.php?d=95946#p435882>



Ciprian Ioan CUCU has graduated the Faculty of Economics of the West University of Timișoara in 2004 as head of class, in the field of Business Informatics. He is currently a PhD student at the “Babeș-Bolyai” University of Cluj-Napoca, Faculty of Economics and Business Management, in the field of Cybernetics, Statistic and Economic Informatics. Between 2004 and

2006 he worked as an external collaborator at the West University of Timișoara and from 2006 he is working at the “1 Decembrie 1918” University of Alba Iulia as a teaching assistant. He is also a freelancer in the field of web design and programming and co-author of one book related to web technologies, as well as the author or co-author of several articles in fields such as educational technologies, knowledge management and e-management, published in the proceedings of important international conferences. He was also co-organizer of several international conferences and member in the editorial board of the journal “Annales Universitatis Apulensis Series Oeconomica”.



Marian-Pompiliu CRISTESCU has graduated the Faculty of Planification and Economic Cybernetics in 1985, he holds a PhD diploma in Economics from 2003, obtained at the Faculty of Cybernetics, Statistic and Economic Informatics from the Academy of Economic Studies in Bucharest. Between 1985 and 1991 he worked as an analyst – programmer at the I.A.C.M Computation Office in Olt – Slatina and at the Electronic Computation Territorial Center in Sibiu. In 1991 he joined the university teaching system, going through all didactic positions starting with teaching assistant, lecturer in 2001 and assistant professor since 2008. Presently, he is a full assistant professor in Economic Informatics at the Faculty of Economic Sciences - “Lucian Blaga” University of Sibiu. He is the author of 8 books and over 60 scientific articles in the field of software quality, programming environments, data bases and economic informatics systems. He is equally focused on software development, being the author or co-author of over 25 programming systems for economic management. He has received many diplomas to certify his achievements in the scientific research domain. He has participated as a project director or as a leading team member at 8 research contracts. He is an active member of the scientific and editing committee for the following magazines and journals: Economic Informatics, Journal of Applied Quantitative Methods, The Economic Magazine edited by the Academy of Economic Studies of Moldavia – Chișinău and the „Lucian Blaga” University of Sibiu. He has participated in the scientific committee of over 10 national and international conferences, for the Informatics section and has coordinated the editing of 2 volumes with the projects of some international scientific conferences.



Corina-Ioana CRISTESCU has graduated the Faculty of Cybernetics, Statistics and Economic Informatics in 2009, she holds a bachelor degree in Economics, in the Economic Informatics specialty, with a thesis on Distributed Applications Optimization. She is currently a student at the Scientific Economic Informatics Master program. Between 2007 and 2008 she worked as a programmer for the Halcyon Solutions' company center in Bucharest. From 2008 to the present day she works as a software developer for Trapeze Networks, a leader in wireless solutions. She is the co-author of 6 scientific articles in the field of software quality, programming environments, data bases and economic informatics systems. She is equally focused on software development, being the author of 2 programming systems for human resources and economic management. One of the software systems was presented to a scientific committee as part of a scientific paper in the domain of Distributed Applications Optimizations, and was distinguished with the first prize in the annual Student's Scientific Conference Session held at the Academy of Economic Studies in 2009. Her interest domains include software quality, programming environments, economic informatics systems, optimization algorithms, network programming.