Authoring Systems in Computer-Based Education: Learning Efficacy and Opportunities

Oussama Hamal, Housseine Bachiri, Nour-eddine El Faddouli, and Samir Bennani

Abstract This paper essentially grants an in-depth summary and analysis of the existing research studies dealing with authoring systems in computer-based education. It primarily endeavors to underline the momentous role of computerized learning environments in education. Likewise, it provides a framework for teachers to think more deeply and creatively about how they design and structure activities for different learners and learning styles. While the benefits of engaging in the learning design process exist regardless of the delivery mode (electronic or face-to-face), they are particularly relevant to distance learning or oftentimes used in blended learning. Nevertheless the teacher or the designer of the learning environment should not lean into object construction and provision to the detriment of a variety of pedagogical models which are built around collaborative activity on the part of the learners. Identifying the tools needed to create these types of environments requires knowledge of how these tools work and how they are categorized.

Keywords Content design · Learnability · e-Learning development productivity · Instructional materials · Learning efficacy

1 Introduction

A bibliographic reading has allowed us to identify several definitions of authoring systems ranging from the most simplistic to the most complicated. In this section we will try to propose various definitions to arrive at the end of our own definition which will be compatible with the content to be presented in our MOOC.

O. Hamal (✉) · N. E. Faddouli · S. Bennani
RIME Team - LRIE Laboratory (EMI), Mohammed V University, Rabat, Morocco
e-mail: faddouli@emi.ac.ma

H. Bachiri · N. E. Faddouli · S. Bennani
Faculty of Sciences and Technologies (FSTT), Abdelmalek Essaâdi University, Tangier, Morocco

M. Ben Ahmed et al. (eds.), *Innovations in Smart Cities Applications Volume 5*,
Lecture Notes in Networks and Systems 393, https://doi.org/10.1007/978-3-030-94191-8_58

According to Masterman and Vogel (2013), an authoring system is any software that supports learning design and allows users to create a representation of their design. At the simplest level, this may involve using a word processor or other general purpose authoring tool like Microsoft PowerPoint. At the most supported level, this involves the use of systems which allow the generation of the most complete learning environment and which gives the possibility of previewing and exporting in various formats which can be executable files, SCORM formats.

(Sharable Courseware Object Reference Model), HTML5 or any other format that can be easily integrated into LMS platforms.

According to (Dağ et al. 2014) authoring systems are software used to develop learning products. The main objective of using this software is to develop educational content. They make it possible to produce interactive courses or learning objects that can be presented in the form of hypermedia or multimedia, by integrating and linking objects, such as texts, images, animation and videos. Users with a basic level of computer science are expected to use any educational authoring system, which has a graphical interface for designing the interface of e-learning materials and designing the content. In addition, the use of a programming language or a scripting language allows the development of more advanced learning objects. The developed learning materials or objects can be published on the web or run on a stand-alone computer or can be integrated as a content component in a learning management system.

According to Murray (2003), an authoring system is generally considered to be responsible for creating all of the elements of a learning environment, such as the area of learning to be presented to students, the user interface, and the functions of the learning environment. It must be noted that some authoring tools treat the resource creation process as a programming task; they provide an array of tools that allow programmers to easily build an intelligent tutoring system. Other authoring systems have a more ambitious goal; they allow people not to expert in the field of programming to create intelligent applications and tutors that ensure the creation of computerized learning environments, ranging from the simple exposure of course content to the demonstration of the correct and incorrect solutions of a set of tasks.

1.1 Our Proposed Definition

In education, an authoring system is software that allows you to create, edit, review, test, and configure resources and objects for educational use. The systems predominantly adopted by content authors are graphical user interface authoring systems. They allow users who do not have programming knowledge to generate computerized learning environments that can be distributed or reused later in several formats, or integrated in e-learning in LMS platforms. However, the use of programming languages makes it possible to generate more intuitive and intelligent learning environments. The production of the material generally uses an educational authoring system and other software which is not designed primarily for the educational field, in this case, it is called the auxiliary authoring system.

2 Related Work

2.1 Families of Authoring Systems

The definitions that we have cited and proposed show that there is not a single family of authoring systems. Indeed, there are those that require programming knowledge to be able to use them, and others that simplify the task and opt for graphical interfaces.

2.1.1 Author Systems with Programming Interface

These systems require knowledge of commands, syntax and coding conventions. They tend to use languages that are less versatile than programming languages but have a simpler and more natural syntax, similar to that of common languages. Programming interface authoring systems, such as HyperCard, have come with their own scripting languages (e.g., HyperTatk) to provide more flexibility and allow users to programmatically perform displays and computer interactions that do not require programming, and hence could not be developed using the authoring system itself. The word "scripting" was used to describe these languages because they were very robust, allowing developers to create programs other than those intended only for education. Most current scripting languages are built into authoring systems. The advantage of these systems is that they allow authors to control the behavior of each component and each object within the limits that the authoring system provides. In fact, he major obstacle is that the simplest of them require knowledge, heuristic thinking and algorithm design.

2.1.2 The Authoring Systems with Graphical Interface

Unlike programming and writing languages, these authoring systems have on-screen tools (menus, prompts,) that allow users to enter text, compose graphics, prescribe ramifications, etc. As the program generates the underlying code, the program structuring requirements are implicit in the system creation procedures. Contemporary authoring systems are more powerful and complex than menu-based systems used in the past, which were typically limited to composing text and graphics and processing user input. They provide utilities for developing audio, video and animation resources and accessing databases, or online resources.

More importantly, these systems allow the creation of educational content for users who are not basic programmers, through elements of the system's graphical interface. To create an application or an EIA, these systems are provided with accessible interfaces similar to those used in common word processing or editing software. Thus they have the ability to display and integrate multimedia resources (sound, image, video, etc.) and the creation of ramifications and hypertext links in just a few clicks. Everything is overseen by functions programmed in a language hidden from author users. However, these systems can use authoring languages close to natural language (such as Toolbook Instructor) which is a specialized computer metalanguage that allows courseware designers to avoid having to program in a computer language, such as Pascal (Yahiaoui 2012). The interface offered by these systems to users is often presented in menus, dialog boxes or other user interface mechanisms. They also provide means to guide the authoring user, which can range from simple messages that are used for information purposes, to assistant modules or even intelligent agents based on the analysis and prediction of authoring practices. Newer GUI authoring systems like (Adobe Captivate and Ispring) provide users with various predefined choices and suggestions, such as content presentation templates. Most of these systems are designed to create basic computer and smartphone courses or to generate them online. The more robust bets, on the other hand, support a variety of media and file types, such as text, graphics, video and audio, to which they add evaluation and test creation functions.

(Steven et al. 2003) argue that neither of these two approaches - based on the programming interface or graphical interface - is likely to provide a general solution to the author's problem. While both approaches are effective in themselves, a task analysis reveals that the author's task is multifaceted and unlikely to be solved in the future.

2.2 Categorization of Authoring Systems

The authoring systems with graphical interface are very varied. In this section, we will describe the main categories of available systems. This categorization is essential for any possible choice, as it allows us to choose the right authoring system according to our learning product needs. It is important to note that these categories are not mutually exclusive. Many systems have elements that qualify them for two or more categories. However, most of them can be categorized based on their primary use or design.

2.2.1 Autonomous Authoring Environments

These applications make it possible to create complete learning environments based on their robustness; they are not based on documents created externally (with the exception of media resources and databases). They typically incorporate

WYSIWYG functionality for layout and screen design, and use an object-oriented approach to structure course elements and activities. The term WYSIWYG is an English acronym meaning "What You See Is What You Get", which can literally be translated in French as "what you see is what you get". This word is widely used in the computer field to refer to graphical interfaces that directly display the final rendering of the document. This type of application is for example present in the vast majority of word processing software which directly displays the page as it appears. This was not always the case, for a time (on the first monitors) words in bold or italics were represented in color on the computer screen (WYSIWYG - Definition, n.d.).

2.3 Rapid Application Development (DRA) Tools

These are open tools for building robust interactive applications (usually for webcasting). They produce binary runtime files that are executed by a player or plug-in (Flex®, Flash®, etc.)

2.3.1 Cloud-Based e-learning Authoring Systems

These systems are cloud-based applications that are installed on a cloud server and use the web browser as the application interface, without requiring installation on your local computer. Some of these cloud authoring systems require the installation of a thin client or browser plug-in. They have several advantages, including:

- When working in groups on the same project, they allow several authors to see the same content at the same time, and therefore to collaborate on it simultaneously. Desktop authoring systems require us to send files to other authors sequentially and to track versions manually.
- The update and configuration of the system are managed centrally. Everyone is always using the latest version, the authors only have one copy of the software available on the cloud server. Desktop authoring systems can be a problem for a group working remotely on the same project, if the versions are not in sync and the functionality is not the same or, worse yet, if the versions are so different that they do not accept files transferred between them.
- They can provide updated and aggregated data on usage, project progress, etc. Some example of cloud authoring systems:
  - Claro®
  - GoMo Learning Suite®
  - Litmos Author®
  - Ilias SCORM Editor (open source)

Many vendors abandon desktop-based authoring applications because they cannot be used collaboratively; some keep desktop-based versions as an option. Desktop applications generally perform better than their web-based cousins, and have more functionality. Some desktop tools (for example, those that feature video editing tools) have no equivalent on the web due to high minimum performance requirements. Examples:

- Adapt [open source] Captivate®
- Course Builder (open source) iSpring Suite®
- ToolBook
- Mediator

3 System Authors of Simulation Development

These tools are specially designed to develop simulations and the animations that compose them. Some integrate scientific data sets that make it possible to model physical phenomena in order to simulate the real world as closely as possible (for example, simulations of chemical tests).

3.1 System Simulation Development Tools

These tools are optimized for systems training, essentially producing a recording of what is happening on a computer screen (often called "screencasts"). They make it easy to capture and caption interface features with narration or voiceover, additional graphics and interaction.

Examples:

- Assima Training Suite®
- Adobe Captivate®
- Firefly Simulation Developer®

3.2 Development Systems for 2D Simulations

These tools are used to create 2D simulations. They are much easier to learn and use and less expensive than 3D simulation development tools, some digital learning development systems as adapte have the possibility of developing such simulation without having recourse to these tools.

Examples:

- Vyond and GoAnimate (2D in cloud)
- Expladio 3D (on computer)

3.3 Serious Game Development Authoring Systems

While there are many RAD and simulation tools that we can use to build learning-through-play applications, the tools in this category are game-builders, or particular game standards, or have gamification as a central feature of learning.

4 Author Systems for mLearning

Most recent authoring systems can now deliver content to mobile devices, simply because mLearning is emerging as the dominant paradigm for the delivery of e-learning. Systems provide this capability by using a mobile device screen template and output files that work with the operating system of the mobile device.

However, tools specifically designed for mobile learning (mLearning) are emerging, for example, offering a capacity for authoring audio learning content (for example, spoken words, podcasts) as well as assessments and associated interactive surveys. Other tools are optimized to provide online learning content through the browsing capability of the phone. Responsive design is now a key feature of many of these tools, and this design will be covered in detail in the MOOC we are creating.

Educational apps for mobile devices differ in several ways, including the following:

- Operating systems and hardware specifications (especially screen size and resolution) of mobile devices are very different from device to device, resulting in the emergence of responsive design for more details, (responsive design)
- The connection speed to data networks is very variable, depending on the time of day, the location of the user, etc.
- Performance is generally much less powerful than desktop computers. It depends on factors like memory, disk space, chip design, etc.
- Mobile phones are very personalized (as opposed to desktops) making it difficult to define a basic design.
- There are different paradigms for interacting with mobile devices (eg using fingers, especially thumbs, rather than a mouse). This poses problems for “rollover” type interactions and large text entry windows.
- Many phones can dynamically change portrait or landscape orientation. Content may need to be adjusted accordingly, or be viewed only in locked mode (which users should be aware of).

– There is a need to test content developed on many different platforms - small businesses normally do not have the resources to acquire all of this content. One way to avoid this problem is to use http://www.deviceanywhere.com for testing (a cloud-based cell phone emulator that shows you what your learning looks like and how it works on any platform). It must be noted that emulators are not always 100% compatible with the real device.

5 The Selection Criteria for an Author System

While choosing the best creative tool requires careful attention to details, we can even choose the system that's right for us if we have a clear idea of what we want to achieve. The tools currently available offer a variety of functionality. For example, some authoring systems are designed to develop in-depth assessments, simulations, or content for tablets and smartphones. If we have special needs, we will certainly find a system that will meet them.

Identifying the system best suited to your project requires prospecting for needs. Regardless of your knowledge of authoring systems, you need to determine the most important functionality that will help conceptualize your deliverable.

The criteria we will cite will allow for a basic understanding of the authoring systems, while provoking analytical thinking to help identify the capabilities and options that are most important to achieving the desired project.

– *Interoperability and standards*

The ability of an authoring system to build applications that can work with other software and online LMS is called interoperability. The e-learning community has several sets of technology standards and is continually developing additional standards. The four most common standards are AICC, SCORM, IMS Global Learning Consortium, and Microsoft LRN. Authoring tools are distinguished by the standards they support. SCORM is the one supported by MOODLE.

– *Types of questions*

People who develop courses are always interested in the different types of quizzes they can create with an authoring system. Fill in the blanks, matches, true or false, drag and drop, etc. … the type or types of questions that will be included in the learning environment to be produced must be taken into account when choosing the system use.

– *Multiple learning paths*

To account for the differences between learners, some authoring systems have the ability to create variations, which is an important feature of differentiated pedagogy. Courses created using variations are more complex to design, but they take into account a range of knowledge and skills. For example, Adobe Captivate makes it easy, and lets you create variations with just a few clicks using a feature called “status view,” which lets you display different content when you click on the same order. This command can be easily programmed to display content appropriate to the level of the learner, who uses the application.

– *Media and file support*

Most authoring systems support common file types such as JPG, WAV, GIF, PNG etc.; While little software supports various video and audio formats, for example Toolbook only supports AVI format with great care in codecs. To do this, you must pay attention to the question of the formats supported when choosing, or provide an auxiliary authoring system to convert the files that will be used to create the resource.

– *Extensibility*

Some authors have to adapt the software used for specific purposes, which are not available in the original version, to do so requires an open source authoring system. The issue of extensibility is often a concern of organizations and staff specialized in the field, it requires either a great mastery of programming, or the availability of a budget that allows the task to be subcontracted to a programmer, in order to adapt the authoring system to its needs.

– *Output format*

The most important question to ask when choosing an authoring system is probably the following: “What output file format (s) does it produce? It is important to determine the format of exit before starting to choose. This allows the choice to be filtered and targeted considerably, as well as to ensure in advance that the files will work in the training IT infrastructure, including end user platforms (e.g., PC and Mac), operating systems and browsers. Similarly, one should avoid being stuck with a proprietary format that could disappear from the market without the possibility to open and edit files, or run them on the available platforms.

Interestingly, output formats become even more important as we move into the era of mobile learning. Newer authoring systems have features that make it possible to produce files that can be read on mobile devices. In the past, developers had to completely customize the architecture and format of deliverables to be read on mobile devices.

6 The Advantages of Using Authoring Systems

With the shift in the learning paradigm from teacher or subject-centered learning to student-centered learning, there is a tremendous need for change in teaching, learning, pedagogy and assessment. In effect, there is nothing better to take advantage of the technological advancements of the twenty-first century to accompany this change. Also to improve leaning motivation by establishing a suitable learning environment. Therefore, the learner will be extrinsically motivated so as to satisfy his three self-determination theory psychological needs:

- Helping learners to acquire the freedom of decision making in order to encourage their need.
- Guiding the learners by providing them with a pedagogical support.
- To give the learners the opportunity to share a common learning experience and give them the chance to interact with each other by utilizing competitive activities. Hamal and all (2021a, b)

Technology is developing rapidly and participating in education as a teaching and learning tool. In addition to the horizons that technology has enabled them to open through e-learning, the latter has contributed to the democratization of knowledge and make it more accessible, it has also been able to renew the form of direct learning delivery, and increase its attractive potential. Clark & Mayer (2016) assert that the use of multimedia and in particular the mix between word and graphic, (which is the specialty of authoring systems par excellence), optimizes the learning process itself. The use of engaging high profile media can improve efficiency and learning and make it an enjoyable experience.

GUI authoring systems (as opposed to writing code or script directly in a programming editor) reduce the technical burden; they typically use WYSIWYG ("what you see is what you get") interfaces allowing users to easily manipulate and configure digital learning resources, using familiar visual metaphors. Thus, programming editors that facilitate the writing of application code like C++ or scripting languages like JavaScript are not true authoring systems for producing educational resources. Developers can indeed use them to create learning content, but they are not designed to reduce the technical costs of knowing the programming or scripting language. In addition, most teachers and training organizations do not have advanced (and expensive) programming skills and cannot produce learning applications using only programming languages or scripts, and they do not have the infrastructure needed to support the development of traditional code-based software applications.

In today's times, all data and information is disseminated around the world via the internet/web, so that anyone can easily access all the data and information they want. The big problem with the content available online is its relevance and adaptability to the conditions in which the teacher or the educational organization operates. The use of authoring systems can be promising, because their rational use makes it possible to produce resources as desired and well adapted to their own context (target audience, available facilities, pedagogies used and styles of teaching and learning). They allow teachers and institutions to produce their own educational content such as course learning materials or assessments, and also reduce costs and have educational content tailored to their learning objectives and goals.

The use of any authoring system directly influences the teaching practices of a teacher as it requires a high level of organization. Before looking into the creation of each digital learning environment, the teacher or the educational organization must every time specify their objectives, make a prospection of what they know how to do and what their target audience knows about the object of study to be presented. They will also be invited to support the various resources likely to be used in their project, and also to define their sequences and their uses. In short, the process of editing educational resources using an authoring system requires the implementation of instructional engineering and scripting techniques each time.

6.1 Licensing Fees for Authoring Systems

A software license, in general, is a document that specifies what can and cannot be done with the software. It grants authorizations, imposes restrictions on users and specifies the financial compensation (Zaatri et al. 2020). The most popular authoring systems are proprietary software, the user, whether natural or legal, must pay a consideration for being able to use them. The costs vary from a few hundred to some tens of thousands of dirhams. However, there are copyright-free authoring systems, but they have many limitations.

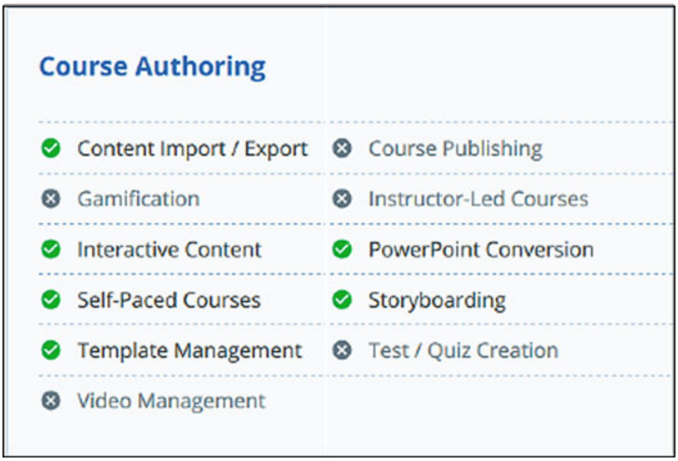
Free software is defined by the fact that its source code is made available free of charge and that users are allowed to study, modify it and distribute it to anyone and for any reason. However, the acquisition and use of this software is not always free.

Open source options are obviously attractive to buyers because they do not involve any licensing costs or minimal fees. However, the pros and cons of purchasing an open source authoring system should be carefully considered, as in the long run the cost could be equal to or greater than that of a commercial solution. It's easy to get seduced by the free license aspect and ignore the other aspects of the solution that can cripple Our authoring project. For example Courselab is a free authoring system, which costs 2000 MAD for a personal use license, however it offers less functionality (see Fig. 1) compared to proprietary software of the same price range, such as Mediator.

It is oftentimes the case where the potential advantage of open source authoring system scan be overlooked as the product can be fully tailored to the user's particular needs. If well managed, this advantage can make an open source solution less expensive, not only because the license is free, but also because development and customization efforts can be focused solely on user needs and nothing more.

In a less developed country like Morocco, the acquisition of a paying authoring system, even open source, remains too expensive for an individual. The solution may be to buy the licenses in batches by the state and provide them to teachers and trainers, which will lower consider the purchase costs. Indeed, the system designers, authors of educational application productions, provide special offers for batch purchases whether by states or by institutions.

Adobe for example, provides the ability to purchase shared device licenses, Institutions can purchase shared device licenses to give students and teachers the ability to access Adobe products in shared environments such as labs, libraries and



Course Authoring

✓ Content Import / Export	⊗ Course Publishing
⊗ Gamification	⊗ Instructor-Led Courses
✓ Interactive Content	✓ PowerPoint Conversion
✓ Self-Paced Courses	✓ Storyboarding
✓ Template Management	⊗ Test / Quiz Creation
⊗ Video Management	

Fig. 1 Characteristics of CourseLab (CourseLab Reviews and Pricing - 2020, s. D.)

classrooms. Through these licenses, authorized users can access the latest versions of Creative Cloud applications and services on machines installed with a shared device license.

Purchases in batches often subject to negotiation and the suppliers of the authoring systems, Morocco intends to modernize its education systems and generalize the use of ICT because authoring systems have a promoting potential and can effectively contribute to this modernization, provided that these systems are made available to teachers and trained in this subject. Hamal and All (2021a, b).

7 Conclusion

This paper aimed at confirming and defending the precise choice of authoring systems that will be highlighted through our MMOOC. Indeed, the training that we are going to make available to our learners does not require in-depth knowledge of computer science, hence the choice of authoring systems with a graphical interface, which will allow our registrants lacking computer knowledge to continue and to easily benefit from our training, while being able to produce their own quality educational applications. In this chapter, we have tried to categorize these systems according to the tasks they perform as well as their execution environment (cloud or computer). Given their diversity and variety, we have suggested certain criteria to be respected in order to make the relevant choice of the appropriate authoring system with its context and expectations. As we have highlighted the contribution and the positive impact of this software in the production of educational applications as well as in the practices of teachers. As a substitute, we have not ruled out the problem of expensive costs and the need for the purchase of licenses in batches by the state to circumvent this problem which imposes the use of these authoring systems and their accessibility for teachers.

References

R.C. Clark, R.E. Mayer (eds.), *Applying the Multimedia Principle: Use Words and Graphics Rather than Words Alone*. in E-Learning and the Science of Instruction (John Wiley & Sons, Inc., Hoboken 2016), pp. 67–87. https://doi.org/10.1002/9781119239086.ch4

CourseLab Reviews and Pricing—2020. (s. d.). Consulté 29 mai 2020, à l'adresse https://www.capterra.com/p/128438/CourseLab/

F. Dağ, L. Durdu, S. Gerdan, Evaluation of educational authoring tools for teachers stressing of perceived usability features. Procedia Soc. Behav. Sci. **116**, 888–901 (2014). https://doi.org/10.1016/j.sbspro.2014.01.316

L. Masterman, M. Vogel, Practices and processes of design for learning. In: Rethinking Pedagogy for a Digital Age: Designing for 21st Century Learning (2013)

T. Murray, An overview of intelligent tutoring system authoring tools: updated analysis of the state of the art. In: Authoring tools for Advanced Technology Learning Environments: Toward Cost-effective Adaptive, Interactive, and Intelligent Educational Software (2003)

R. Steven, S.B. Stephen, W. Leslie, Authoring tools for component-based learning environments, ed. by T. Murray, S.B. Blessing, S. Ainsworth, Authoring Tools for Advanced Technology Learning Environments. (Springer, Dordrecht, 2003). https://doi.org/10.1007/978-94-017-0819-7_16

L. Yahiaoui, Redocumentation des traces d'utilisation d'un environnement informatique [Thesis, Lyon 1] (2012). http://www.theses.fr. http://www.theses.fr/2012LYO10140

I. Zaatri, S. Margoum, R. Bendaoud, I.L. El Malti, D. Burgos, K. Berrada. Open educational resources in Morocco, ed. by R. Huang, D. Liu, A. Tlili, Y. Gao, R. Koper, Current State of Open Educational Resources in the Belt and Road Countries (Springer, Singapore, 2020), pp. 119–134. https://doi.org/10.1007/978-981-15-3040-1_7

O. Hamal, Boosting e-learner's motivation through identifying his/her emotional States. Iraqi J. Sci. Spec. Issue 127–132 (2021a). https://doi.org/10.24996/ijs.202.SI.1.17

O. Hamal, Review and taxonomy of MOOCs and the significance of digital transformation of learning in higher education institutions: prospects and opportunities. Int. J. Soc. Sci. Hum. Res. **04**(08) (2021b). https://doi.org/10.47191/ijsshr/v4-i8-12