

Analiza właściwości, funkcjonalności i ergonomii nowoczesnych narzędzi dedykowanych do szybkiego tworzenia kursów

Barbara Dębska
Politechnika Rzeszowska
bjdebska@prz.edu.pl

Lucjan Dobrowolski
Politechnika Rzeszowska
ldobrowolski@prz.edu.pl

Karol Hęclik
Politechnika Rzeszowska
kheclik@prz.edu.pl

Streszczenie: W artykule dokonano analizy narzędzi przeznaczonych do szybkiego tworzenia kursów e-learningowych. Powstała ona na bazie doświadczeń zdobytych przez autorów podczas realizacji cyklu szkoleń z zakresu e-nauczania, zorganizowanych dla nauczycieli akademickich Politechniki Rzeszowskiej. W treści publikacji zawarto spostrzeżenia, które pojawiły się głównie podczas części praktycznej szkoleń. Omówiono potencjalne problemy i trudności, na które napotykają osoby, które tworząc kursy e-learningowe korzystają z obecnie dostępnych narzędzi do szybkiego tworzenia kursów. Skoncentrowano się na analizie właściwości i funkcjonalności najbardziej popularnych narzędzi RLD (Rapid e-Learning Development), ergonomii pracy podczas korzystania z tych aplikacji oraz ich niedoskonałościach, co ostatecznie zaowocowało opracowaniem zbioru cech definiujących narzędzie „idealne”. Podczas oceny istniejących narzędzi rozpatrzono te ich funkcjonalności, które pozwalają na włączenie najnowszych technologii internetowych w proces tworzenia e-kursów.

Słowa kluczowe: e-learning, e-kurs, narzędzia do szybkiego tworzenia kursów, RLD

1. Wprowadzenie

Od dłuższego już czasu na rynku dostępne są rozbudowane narzędzia przeznaczone do szybkiego tworzenia kursów e-learningowych. Aplikacje te oczywiście różnią się między sobą funkcjonalnościami oraz ergonomią pracy. Jednak większość z nich posiada pewien wspólny lub zbliżony zbiór cech, który pozwala generować treść kursów w postaci umożliwiającej wyświetlanie jej w przeglądarce WWW.

Obecnie oczekuje się, że kurs e-learningowy będzie zawierał treści multimedialne i interaktywne. Przykładowo, w kursach nauczania chemii istotne jest umieszczanie w e-lekcji treści takich jak: struktury chemiczne, widma, wykresy, diagramy, schematy poglądowe czy nagrania wideo prowadzonych eksperymentów laboratoryjnych. Te treści można przedstawić zarówno w postaci grafiki statycznej (ilustracje, fotografie) opatrzonej komentarzami głosowymi, grafiki dynamicznej (nagrania wideo), jak i grafiki interaktywnej (animacji). Wiedza podana w postaci animacji wydaje się łatwiejsza w przyswojeniu przede wszystkim dlatego, że pozwala w jasny i uporządkowany sposób zaprezentować te zjawiska i procesy, które mają charakter złożony czy wieloetapowy. Animacja pozwala dokładnie prześledzić ich etapy, np. poprzez zatrzymanie jej wykonania lub pracę krokową. Przekazanie tych treści wymaga użycia odpowiednich technologii. W ciągu ostatnich kilku lat pojawiły się i ugruntowały się odpowiednie do zrealizowania tego celu technologie internetowe. Należą do nich między innymi języki: HTML 5 (w tym znaczniki: Canvas, Audio, Video), arkusze stylów CSS 3, SVG, MathML i JavaFX. Ponadto technologie już ugruntowane zyskały nowe własności, np. technologia Flash obecnie płynnie obsługuje złożone animacje 3D. Wszystkie te technologie są dobrze obsługiwane przez przeglądarki WWW, które są głównym narzędziem stosowanym podczas kształcenia zdalnego. Obecnie technologie te są

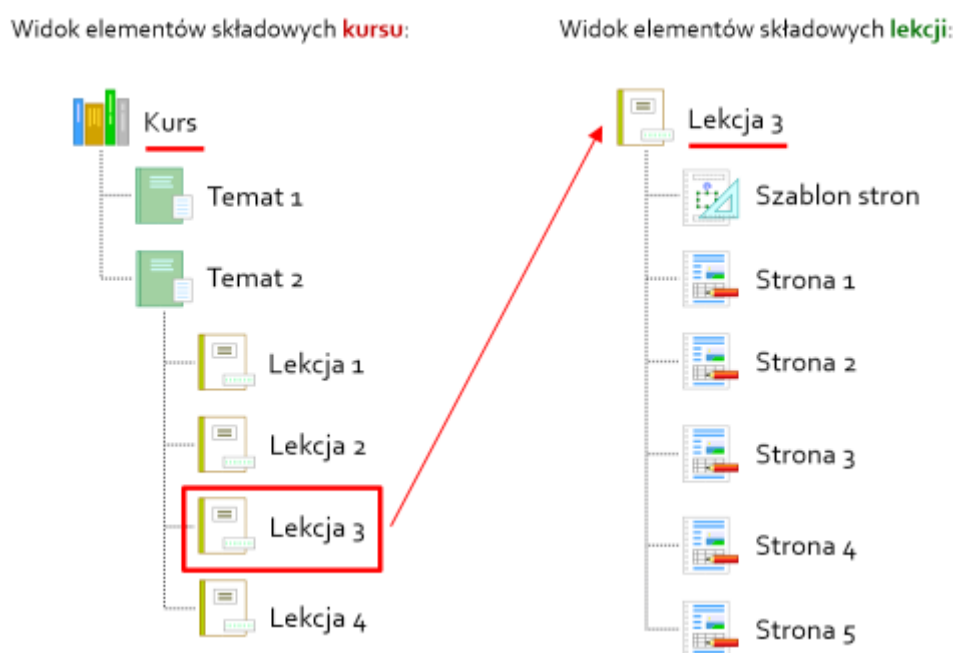
dostępne nie tylko w komputerach klasy PC (stacjonarne i przenośne) ale także w urządzeniach mobilnych, które są wyposażane w różne systemy operacyjne (np.: Android, iOS, Windows RT, Windows Phone). Sytuację nieco komplikuje fakt, że tablety i smartfony różnią się między sobą dostępnymi zasobami sprzętowymi: mocą obliczeniową procesorów, wielkością pamięci RAM, pojemnością pamięci masowych (często opartych na modułach pamięci typu Flash, np. karty SD) oraz rozdzielczością ekranów.

1.1. Edytory RLD

Aplikacje RLD są rodzajem edytorów graficznych, które pozwalają na ustalenie struktury kursu, podział na tematy, lekcje oraz strony wypełniane treścią. Strony mogą być wyświetlane sekwencyjnie lub według aranżacji zaprojektowanej przez autora kursu. Treść stron jest lokowana w obiektach mających postać ramki, w której umieszcza się m.in.: tekst, grafikę, nagrania wideo, pliki audio, animacje. Wybiera się je z dostępnej palety obiektów. W trakcie redagowania kursu można dostosowywać wygląd i zachowanie obiektów do opracowanego projektu. Po zakończeniu redagowania kursu, zawartość jest eksportowana do formatu akceptowanego przez docelowy system LMS (Learning Management System) zarządzający kursami, tak aby strony e-kursu mogły zostać uruchomione w przeglądarce WWW. Najczęściej kurs jest eksportowany do pakietu zgodnego ze standardem SCORM. W takim wypadku każda strona kursu jest zamieniana na stronę HTML. Dodatkowo narzędzie edycji kursu generuje stosowne pliki manifestu oraz pliki skryptów języka JavaScript, niezbędne do współpracy z systemem LMS. Dostępne są też narzędzia, które eksportują kurs do formatu Flash.

1.2. Kurs e-learningowy – elementy składowe

Zarządzanie kursem zazwyczaj wykorzystuje jego podział na tematy, które składają się ze zbioru lekcji. Każda lekcja zawiera ciąg stron. W edytorach kursów dla wygody użytkowników ta struktura często dzielona jest na dwie listy, jak pokazano to na Rys. 1. Lista po lewej stronie pokazuje podział kursu na tematy i lekcje. Wybór węzła lekcji skutkuje wyświetleniem listy jej stron. Często można też zdefiniować szablon dla stron lekcji. Strony mogą także zawierać meta-informacje, np.: słowa kluczowe, daty ostatniej aktualizacji, nazwiska autorów treści lub właściciela kursu.



Rysunek 1. Przykładowy widok struktury e-kursu

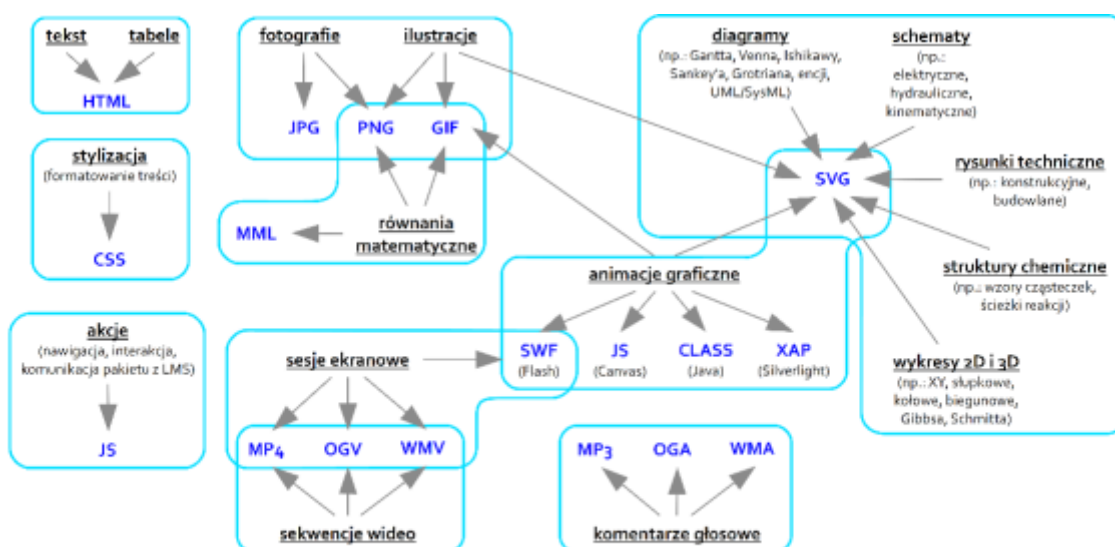
1.3. Rodzaje treści umieszczanych w kursach

Pojedyncza strona lekcji może zawierać różne rodzaje treści. Możemy je podzielić na:

1. obiekty podstawowe: tekst, tabele, grafika, wyrażenia matematyczne,
2. obiekty prezentujące dane naukowe/techniczne: wykresy, schematy, diagramy, rysunki techniczne, struktury chemiczne, mapy i plany,
3. obiekty multimedialne: sekwencje wideo, komentarze głosowe, sesje ekranowe, animacje graficzne.

1.4. Środowisko używane do wyświetlania kursów

Docelowo e-kursy są umieszczane w systemie LMS, a ich treść użytkownik końcowy wyświetla w przeglądarce WWW. Schemat przedstawiony na Rys. 2 ilustruje powiązania pomiędzy rodzajami treści a formatami danych obsługiwanych przez przeglądarki WWW lub współpracujące z nimi moduły dodatkowe (wtyczki).



Rysunek 2. Typowe rodzaje treści umieszczane w e-kursach oraz formaty danych pozwalające je przechowywać

Niektóre rodzaje treści mogą być obsługiwane przez jedno z wielu dostępnych rozwiązań. W szczególności dotyczy to równań matematycznych, ilustracji, komentarzy głosowych, sekwencji wideo, sesji ekranowych i animacji. Także treści takie jak: równania matematyczne, wykresy, schematy, diagramy, rysunki techniczne, struktury chemiczne można zapisać jako grafikę rastrową albo wektorową. Sama przeglądarka posiada ograniczenia narzucane przez standardy takie jak: HTML, CSS, JavaScript i technologie im towarzyszące (np. SVG, HTML5 Canvas itd.). Narzędzie do tworzenia kursów również jest od nich uzależnione. Szczególnie dotyczy to modułu eksportu kursu, który dokonuje przekształcenia treści zapisanych w formatach wewnętrznych narzędzia RLD na formaty czytelne dla przeglądarek internetowych (np. podczas generowania pakietu kursu w standardzie SCORM). Im lepsza obsługa formatów danych przez przeglądarki WWW, tym większe możliwości mają twórcy narzędzi RLD, bowiem mogą wtedy wyposażać swoje narzędzia w nowe własności, które z kolei pozwolą na umieszczenie atrakcyjnych treści w e-kursach.

1.5. Obsługa technologii internetowych w przeglądarkach internetowych

Technologie przedstawione na Rys. 2 są dość dobrze obsługiwane przez większość przeglądarek. W Tabeli 1 podano informacje o obsłudze technologii internetowych istotnych dla e-kursów przez najnowsze wersje najpopularniejszych przeglądarek internetowych (Wayner, 2010) przeznaczonych dla komputerów PC. Z przytoczonego zestawienia wynika, że pewne problemy występują z obsługą nagrań wideo i audio, bo producenci przeglądarek nie doszli do porozumienia w sprawie ujednoczenia obsługiwanych formatów. Zauważa się natomiast, że nie występują już żadne problemy z obsługą zarówno grafiki rastrowej, jak i wektorowej. W technologiach takich jak Flash czy JavaFX obiekty graficzne obsługiwane są przez zainstalowane wtyczki.

Tabela 1. Obsługa technologii internetowych w przeglądarkach WWW dedykowanych dla komputerów PC (stan: 10-06-2014). Opracowano na podstawie Wayner (2010), MathML (2014) oraz Mobile HTML (2014)

L.p.	Technologia	MS IE 11	Mozilla Firefox 29	Google Chrome 35	Opera 20	Apple Safari 7	Konqueror 4.11
1	HTML 5 Basic	tak	tak	tak	tak	tak	tak
2	HTML 5 Audio	MP3	MP3, OGG	MP3, OGG	OGG	MP3	MP3, OGG
3	HTML 5 Video	MP4	MP4, OGG	MP4, OGG	OGG, WebM	MP4	MP4, OGG
4	HTML 5 Canvas	tak	tak	tak	tak	tak	tak
5	CSS 3	częściowo	częściowo	częściowo	częściowo	częściowo	częściowo
6	JavaScript 1.8	tak	tak	tak	tak	tak	tak
9	SVG statyczny	tak	tak	tak	tak	tak	tak
10	SVG animacje	częściowo	częściowo	częściowo	częściowo	częściowo	częściowo
11	MathML	wtyczka	częściowo	—	—	częściowo	—
12	WebGL	tak	tak	tak	tak	tak	tak
13	AJAX	tak	tak	tak	tak	tak	tak
14	Flash	wtyczka	wtyczka	wtyczka	wtyczka	wtyczka	wtyczka
15	JavaFX	wtyczka	wtyczka	wtyczka	wtyczka	wtyczka	wtyczka

Nowe technologie, takie jak HTML5 Canvas czy powiązana z nią technologia WebGL, są już dostępne w nowych wersjach przeglądarek WWW. Niestety technologia MathML, która w kształceniu zdalnym jest bardzo ważna, nadal nie jest obsługiwana w stopniu zadowalającym, tj. zaimplementowano ją tylko w dwóch przeglądarkach, ale w stopniu ograniczonym wyłącznie do znaczników prezentacyjnych. Ponadto standard CSS 3, odpowiedzialny na formatowanie treści strony WWW, jeszcze nie jest w pełni zaimplementowany w przeglądarkach WWW (być może także ze względu na fakt, że niektóre jego moduły wciąż znajdują się w fazie opracowywania).

W Tabeli 2 uwzględniono przeglądarki WWW występujące w urządzeniach mobilnych (w nawiasach podano system operacyjny). W tych urządzeniach przeglądarka jest dostarczana razem ze system operacyjnym, chociaż na niektórych z nich istnieje możliwość zainstalowania przeglądarki internetowej niezależnego dostawcy.

Tabela 2. Obsługa technologii internetowych w przeglądarkach WWW dedykowanych dla urządzeń przenośnych (stan: 10-06-2014). Opracowane na podstawie World Wide Web Consortium (W3C, 2014), MathML (2014) oraz Mobile HTML (2014)

L.p.	Technologia	Internet Explorer (Windows Phone)	Mozilla Firefox (Android)	Google Chrome (Android)	Opera Mini (Android)	Apple Safari (iOS)	Android Browser (Android)
1	HTML 5 Basic	tak	tak	tak	tak	tak	tak
2	HTML 5 Audio	zależne od OS: Android – tak; iOS, WindowsRT, Windows Phone – problematyczne					
3	HTML 5 Video	zależne od OS: Android – tak; iOS, WindowsRT, Windows Phone – problematyczne					
4	HTML 5 Canvas	tak	tak	tak	tak	tak	tak
5	CSS 3	częściowo	częściowo	częściowo	częściowo	częściowo	częściowo
6	JavaScript 1.8	tak	tak	tak	tak	tak	tak
9	SVG statyczny	tak	tak	tak	tak	tak	tak
10	SVG animacje	tak	tak	tak	tak	tak	tak
11	MathML	—	—	—	—	—	—
12	WebGL	—	tak	tak	—	—	tak
13	AJAX	tak	tak	tak	—	tak	tak
14	Flash	zależne od OS: Android – tak; iOS, WindowsRT, Windows Phone – problematyczne					
15	JavaFX	zależne od OS: Android – tak; iOS, WindowsRT, Windows Phone – problematyczne					

Dostępność poszczególnych technologii internetowych jest zależna od systemu operacyjnego zainstalowanego w urządzeniu mobilnym. Możliwość wyświetlania nagrań wideo i audio oraz uruchamiania animacji Flash i apletów Java jest zależna od systemu operacyjnego. W urządzeniach z systemem Android nie ma większych przeszkód z zainstalowaniem dodatkowego oprogramowania, o ile ono istnieje. Natomiast w przypadku urządzeń z systemem iOS czy Windows Phone oraz Windows RT jest to utrudnione. W tych ostatnich, instalacja wtyczek Flash i Java oraz kodeków audio i wideo jest utrudniona lub wręcz niemożliwa. Również dostępność kodeków dedykowanych dla poszczególnych formatów jest różna dla różnych systemów operacyjnych zainstalowanych w urządzeniach mobilnych.

2. Współczesne narzędzia RLD – funkcjonowanie i ergonomia

W narzędziach do szybkiego tworzenia kursów bardzo istotna jest możliwość zdefiniowania właściwej nawigacji pomiędzy stronami lekcji i wpływanie na jej interakcje z użytkownikiem. Często jest to zrealizowane jako przypisanie akcji określonego typu do strony lub obiektu. Narzędzie do tworzenia kursów zwykle dostarcza zbiór gotowych akcji. Te akcje po eksporcie kursu do pakietu, np. SCORM, są przekształcane na zbiór funkcji zapisany w języku JavaScript. Tworzenie własnej niestandardowej akcji jest dostępne jedynie w bardzo nielicznych programach. Takim programem jest Authorware (<http://www.adobe.com/products/authorware>). Firmy tworzące narzędzia RLD na ogół wychodzą z założenia, że jest to zbyt trudne do opanowania i obsługi dla użytkowników. Być może powodem jest też to, że programiści w tych firmach nie potrafią opracować skutecznego i efektywnego rozwiązania, pozwalającego użytkownikowi tworzyć diagramy interakcji, które byłyby przekształcane podczas eksportu e-kursu do postaci

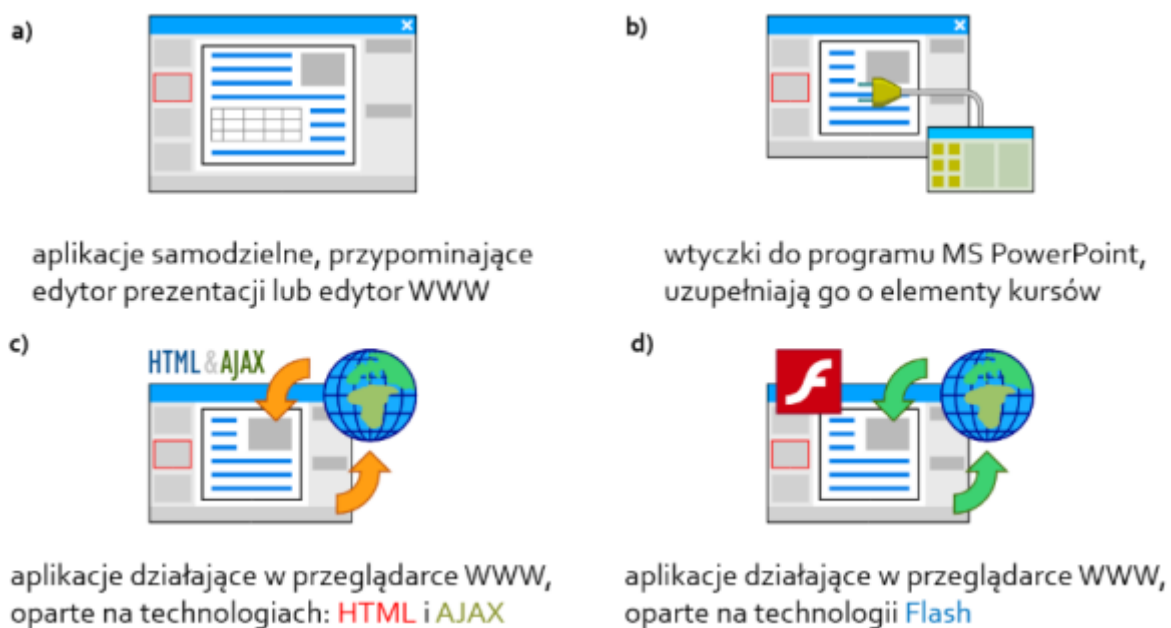
skryptów języka JavaScript. Tymczasem możliwość definiowania własnych akcji byłaby bardzo użyteczna dla nauczycieli tworzących kursy.

2.1. Authorware – przodek narzędzi RLD

Jednym z pierwszych programów przeznaczonych dla e-learningu jest Authorware (Lakens, 2005), stworzony w firmie Authorware Inc. Pozwala na definiowanie własnych akcji: interakcji obiekt–użytkownik oraz nawigacji pomiędzy stronami kursu. Definiowanie akcji przypomina tworzenie programu. W tym przypadku instrukcje mają postać graficzną, nie tekstową. Ułatwia to opanowanie obsługi programu przez osoby nie posiadające wiedzy z zakresu programowania. Okno programu zawiera paletę obiektów i instrukcji, wykorzystywanych w edytorze kodu. Interakcje i nawigacja są przedstawiane w postaci diagramu. Wskazany obiekt można dostosować poprzez panel właściwości. Ponadto w kodzie można używać: funkcji, zmiennych i obiektów wiedzy zgromadzonych w standardowej bibliotece. Interaktywne prezentacje mogą łączyć w sobie: dźwięk, tekst, grafikę, filmy i proste animacje. Program zawiera także narzędzia oceny pracy studenta. Pozwala na generowanie pakietów SCORM, a utworzone w nim kursy mogą być umieszczane na dowolnej platformie LMS obsługującej ten standard. Niestety od wersji 7 aplikacja Authorware nie jest już rozwijana.

2.2. Sposób pracy współczesnych narzędzi RLD

Funkcjonalności dostępnych współcześnie narzędzi RLD można podzielić na cztery grupy, co pokazano na Rys. 3. Sposób ich pracy jest zbliżony do działania edytorów prezentacji (Rys. 3a). Z tego względu, oprócz samodzielnych aplikacji do tworzenia kursów, powstało sporo rozwiązań bazujących na programie MS PowerPoint, które dodają do niego brakujące elementy (np. możliwość tworzenia testów i quizów, moduł eksportu do pakietu SCORM). Narzędzia te działają jako wtyczki (Rys. 3b). Dostępne są też narzędzia działające jako aplikacje uruchamiane w przeglądarce WWW (Rys. 3c). Do działania wymagają połączenia internetowego oraz przeglądarki zdolnej do pracy asynchronicznej (AJAX). Wyjątkiem są bardzo nieliczne aplikacje przeglądarkowe, wykorzystujące technologię Flash (Rys. 3d) zamiast rozwiązań bazujących na języku HTML i AJAX.



Rysunek 3. Typowe sposoby pracy narzędzi do tworzenia e-kursów

2.3. Najpopularniejsze narzędzia do szybkiego tworzenia kursów

Na rynku jest obecna spora liczba aplikacji do projektowania e-kursów. Nie wszystkie z nich można nazwać narzędziami do szybkiego tworzenia kursów (RLD). Z tego względu do analizy została wybrana grupa, która pozwala łatwo budować e-lekcje. Narzędzia te zgrupowano w Tabeli 3. Wszystkie wymienione aplikacje są dostępne komercyjnie. Część z nich występuje także w wersjach darmowych, które mają jednak ograniczone funkcjonalności. Natomiast tego typu narzędzia, udostępniane na licencji OpenSource, cechuje ubogi zbiór funkcjonalności i niska ergonomia pracy. Nie obsługują one też wielu niezbędnych technologii internetowych.

Spośród aplikacji wymienionych w Tabeli 3 pierwszy jest ActivePresenter (Atomi Systems, 2014). Posiada on graficzny edytor slajdów, panel zawierający listę utworzonych slajdów, listwę czasową pozwalającą na manipulację treścią oraz panel właściwości, przeznaczony do dostosowania obiektów umieszczonych na poszczególnych slajdach. Inne podobne narzędzia to Captivate (<http://www.adobe.com/pl/products/captivate.html>) i Storyline (<https://www.articulate.com/products/storyline-overview.php>). Cechują je funkcjonalności analogiczne jak ActivePresenter. Ponadto Storyline posiada bibliotekę akcji, którą można wykorzystać podczas realizacji interakcji użytkownika z treścią kursu.

Tabela 3. Dostępne obecnie narzędzia do szybkiego tworzenia kursów (stan: 10-06-2014)

L.p.	Nazwa narzędzia	Producent	Sposób działania	Cena netto
1	ActivePresenter	Atomi Systems	desktop, samodzielne	399 \$
2	Authorware	Adobe	desktop, samodzielne, nierozwijane	80 \$
3	Captivate	Adobe	desktop, samodzielne	356 €
4	Courselab	WebSoft	desktop, samodzielne	699 \$
5	Impatica	Impatica Inc.	wtyczka do MS PowerPoint	200 \$
6	iSpring Suite	iSpring Solutions	wtyczka do MS PowerPoint	587 \$
7	Lectora	Trivantis	desktop, samodzielne	2 495 \$
8	Mediator	Matchware	desktop, samodzielne	599 \$
9	Office Mix	Microsoft	wtyczka do MS PowerPoint	0 \$
10	Presenter	Adobe	wtyczka do MS PowerPoint	593 €
11	SmartBuilder	Strategic Technology Solutions	desktop, samodzielne	1 970 \$
12	Storyline	Articulate	desktop, samodzielne	1 398 \$
13	Toolbook	Sumtotal	desktop, samodzielne	2 795 \$
14	ViewletBuilder	Qarbon	desktop, samodzielne	599 \$
15	Workplace Collaborative Learning Authoring Tool	IBM Lotus	desktop, samodzielne, nierozwijane	—
16	WBT Express	4systems	desktop, samodzielne	249 \$
17	ZebraZapps	Allen Learning Technology	usługa w Internecie	200 \$/m-c
18	Zenler Studio	Zenler	wtyczka do MS PowerPoint	499 \$

Inną ważną aplikacją jest IBM Lotus Workplace Collaborative Learning Authoring Tool (Thompson, 2014), która została stworzona przez firmę Lotus (część koncernu IBM). Program był dostarczany jako dodatkowe wyposażenie dla systemu LMS firmy IBM. Posiada wizualny edytor struktury logicznej kursu oraz graficzny edytor stron, co nadal jest rzadkością wśród narzędzi RLD. Wyposażony jest w: panel zawierający listę utworzonych tematów i lekcji oraz listę stron w lekcji, a także listwę narzędzi, która zawiera komponenty umieszczane na stronach. Manipulacja wyglądem, treścią i zachowaniem komponentów odbywa się poprzez wywołanie menu kontekstowego wybranego komponentu strony. Aplikacja posiada rozbudowane narzędzia pozwalające tworzyć testy i quizy. Utworzone kursy mogą być eksportowane do pakietu SCORM. Niestety program nie jest już rozwijany.

Bardzo ważnym narzędziem jest aplikacja Lectora (<http://lectora.com/solutions/create-elearning>), stworzona przez firmę Trivantis. Jest ona wyposażona w liczne udogodnienia oraz biblioteki gotowych komponentów. Posiada również graficzny edytor stron, panel zawierający listę utworzonych stron i listę obiektów umieszczonych na stronach, listwę narzędzi oraz bibliotekę gotowych obiektów. Pozwala na manipulowanie nawigacją oraz interakcją poprzez przypisywanie gotowych akcji do slajdów i obiektów. Podobną aplikacją jest ToolBook (SumTotal Systems, b.d.). Cechuje ją funkcjonalność bardzo zbliżona do wymienionych narzędzi. Nieco mniejszą od nich funkcjonalność posiadają programy: Courselab (WebSoft, b.d.), Mediator (<http://www.matchware.com/en/products/mediator/productinfo/features.htm#05>), SmartBuilder (<http://www.smartbuilder.com/product/smartbuilder>), ViewletBuilder (<http://www.qarbon.com/presentation-software/viewletbuilder>) i WBT Express (http://www.4system.com/wbts_pro.html). Użytkownik jest zmuszony do używania okien dialogowych zamiast swobodnie dostępnych palet narzędziowych. Najmniej wygodny interfejs użytkownika posiada program WBT Express.

Ciekawym oprogramowaniem jest pakiet iSpring (<http://www.ispringsolutions.com/ispring-suite?ref=home>), który został stworzony przez firmę iSpring Solutions. Składa się z kilku modułów. Najważniejszy z nich jest wtyczka, która integruje się z programem MS PowerPoint i uzupełnia go o funkcjonalność pozwalającą na tworzenie e-kursów. W szczególności dotyczy to tworzenia testów i quizów oraz zapewnia eksport kursu do pakietu SCORM. Podobnie działają narzędzia takie jak: Impatica (Impatica Inc., 2012), Presenter (<http://www.adobe.com/pl/products/presenter/features.html>) i Zenler Studio (<http://www.zenler.com/authoring-tool.php>). Od niedawna pojawił się ich konkurent o nazwie Office Mix (Office Team, 2014). Jest to dodatek do programu PowerPoint, opracowany przez firmę Microsoft. Jest dostępny bezpłatnie dla zarejestrowanych użytkowników MS Office. Niestety nie posiada on niektórych funkcjonalności, m.in. nie obsługuje standardu SCORM.

Ostatnie z prezentowanych narzędzi to ZebraZapps (<http://www.alleninteractions.com/products/zebrazapps>), stworzone przez firmę Allen Learning Technology. Działa jako aplikacja webowa, wykorzystująca technologię Flash. Nad jego projektem czuwał Michael W. Allen, twórca programu Authorware, który przeniósł do niego wiele wcześniej opracowanych rozwiązań. Korzystając z tego programu, autor kursu może wpływać na nawigację oraz interakcję treści stron z użytkownikiem, poprzez łączenie obiektów „wirtualnymi przewodami”, tworząc w ten sposób diagram przepływu danych i sterowania. Pozwala to na osiągnięcie złożonych animacji bez konieczności posiadania wiedzy na temat programowania. Narzędzie oferuje funkcję tworzenia prostych obiektów, które mogą komunikować się ze sobą poprzez wirtualne „okablowanie”. Bardziej skomplikowane obiekty mogą być tworzone poprzez łączenie obiektów prostszych. Dostęp do aplikacji ZebraZapps opiera się na modelu sprzedaży SaaS (Software as a Service), który wymaga uiszczenia miesięcznego abonamentu.

W Tabeli 3 podano także orientacyjne ceny. W większości wypadków dotyczą one licencji typu Professional, kierowanych do nauczycieli. W przypadku narzędzi działających jako wtyczka

do programu MS PowerPoint, przy obliczaniu kosztów narzędzia należy doliczyć koszt zakupu programu bazowego.

2.4. Zalety i wady współczesnych narzędzi RLD

Większość z analizowanych narzędzi do szybkiego tworzenia kursów posiada pewne wspólne cechy do których należą:

1. graficzny edytor stron kursu,
2. łatwe nawigowanie pomiędzy stronami poprzez listę stron kursu,
3. biblioteka gotowych obiektów, przeznaczonych do osadzania na stronach,
4. możliwość osadzania treści multimedialnych: filmów, dźwięku, animacji,
5. kontekstowy panel właściwości przeznaczony do manipulowania obiektami zawierającymi treść,
6. wbudowane biblioteki prostych treści graficznych i dźwiękowych,
7. gotowe szablony styli dla tworzonych kursów,
8. prosty nadzór nad nawigacją i interakcją pomiędzy stronami,
9. kontrola nad interakcją za pomocą listwy czasowej (tylko niektóre narzędzia),
10. wbudowane biblioteki gotowych akcji do nadzorowania nawigacji i interakcji,
11. eksport kursu do pakietu w akceptowanym standardzie, np.: SCORM,
12. wygoda obsługi i możliwość względnie szybkiego tworzenia prostych kursów,
13. łatwość opanowania dla początkujących użytkowników.

Daje się jednak zauważyć zastój w rozwoju narzędzi do tworzenia e-kursów. Większość z analizowanych programów działa w sposób podobny do programu MS PowerPoint. Jak na dzisiejsze oczekiwania użytkowników jest to zdecydowanie za mało. Znaczna część z w/w narzędzi nie pozwala m.in. na:

1. tworzenie złożonej struktury e-kursu (wyjątek: IBM Lotus Workplace Collaborative Learning Authoring Tool),
2. tworzenie własnych akcji złożonych z prostych instrukcji (tj.: nawigacji pomiędzy stronami, interakcji pomiędzy obiektami i użytkownikiem) za pomocą np.: edytora diagramu przepływu danych i sterowania (wyjątek: ZebraZapps),
3. obsługę niektórych złożonych technologii internetowych, np. animacji SVG, HTML 5 Canvas czy tworzenie złożonych testów i quizów (e-egzaminów),
4. wprowadzanie treści bezpośrednio z poziomu narzędzia, np. nagrywanie audio i wideo, pobieranie tekstu ze skanera poprzez OCR, dyktowanie tekstu głosem, rysowanie ilustracji wektorowych, tworzenie wykresów, itd. (wyjątek: nagrywanie sesji ekranowych w niektórych z wymienionych aplikacji),
5. edycję treści multimedialnych i złożonych obiektów graficznych bezpośrednio z poziomu narzędzia, np.: obróbka audio czy wideo, redagowanie schematów czy diagramów.

Są to dość uciążliwe braki, które powodują, że autor e-kursu musi posiłkować się dodatkowymi narzędziami (np. edytorami audio, wideo, edytorami wykresów, schematów, itd.). Komplikuje to i wydłuża proces tworzenia kursu e-learningowego. Użytkownik jest zmuszony instalować dodatkowe aplikacje oraz nauczyć się je obsługiwać. Bardzo często musi także zagłębiać się w niuanse informatyczne, dotyczące formatów wymiany danych pomiędzy dodatkowymi aplikacjami a narzędziem RLD. Zapewne, także podniesie to koszty tworzenia e-kursów, bo konieczne będzie zakupienie dodatkowych programów. Wprawdzie użycie narzędzi OpenSource może wyeliminować konieczność zakupu takich narzędzi, ale często jest to okupione niską jakością i stabilnością takich aplikacji (np. ShotCut) albo trudnością ich obsługi przez osoby bez przygotowania informatycznego (np. VirtualDub). Rozwiązaniem wydaje się uzupełnienie obecnych narzędzi RLD o funkcjonalności dostępne w narzędziach dodatkowych. Podejście to jest już od dawna stosowane w pakietach graficznych (Adobe Photoshop, Adobe Premiere, Avid Studio),

pakietach do montażu materiałów multimedialnych (Adobe Encore, Adobe Flash), pakietach biurowych jak MS Office, a nawet w pewnym zakresie w pakiecie LibreOffice.

3. „Idealne” narzędzie do szybkiego tworzenia kursów

W oparciu o doświadczenie autorów zdobyte podczas użytkowania niewymienionych w artykule aplikacji: graficznych (np. Adobe Photoshop, Adobe Illustrator, CorelDraw, Corel Photopaint, MS Visio, Blender), multimedialnych (np.: Adobe Audition, Audacity, Avid Studio, Pinnacle Studio, Camtasia) i programistycznych (MS Visual Studio, Delphi, NetBeans, LabVIEW) oraz narzędzi do szybkiego tworzenia kursów, powstała idea zdefiniowania cech hipotetycznego „idealnego” narzędzia. Wśród aplikacji, których cechy należałoby wziąć pod uwagę, są przede wszystkim narzędzia takie jak RAD oraz VI.

3.1. Narzędzia RAD i VI

Narzędzia typu RAD (Rapid Application Developing) są zintegrowanymi pakietami przeznaczonymi do szybkiego tworzenia oprogramowania. Porównanie narzędzi RLD z narzędziami RAD jest o tyle uprawnione, że e-kursy (zwłaszcza multimedialne i interaktywne) można potraktować jak aplikacje, z tą różnicą, że działają w przeglądarce internetowej, a nie bezpośrednio w systemie operacyjnym.

Typowe narzędzie RAD posiada: graficzny edytor okien, paletę komponentów, panel zawierający listę obiektów umieszczonych w oknie, panel edycji właściwości i zdarzeń komponentów, panel zarządzania plikami i ustawieniami projektu. Ponadto zawiera tekstowy edytor kodu źródłowego oraz kompilator i debugger.

Narzędziami pokrewnymi do RAD są narzędzia typu VI (Virtual Instrumentation – wirtualne oprzyrządowanie). Są to graficzne środowiska projektowania dla aplikacji, które przetwarzają strumień danych lub sterują urządzeniami. Zastosowano w nich metodę programowania graficznego, które polega na tworzeniu diagramu przepływu danych i sterowania.

3.2. Podobieństwa i różnice pomiędzy RAD i VI a RLD

Pomiędzy narzędziami typu RAD i VI a narzędziami do szybkiego tworzenia kursów (RLD) można wskazać pewne analogie. Zostały one przedstawione w Tabeli 4. W kolumnie drugiej podano funkcjonalności zaimplementowane obecnie w narzędziach RAD i VI, natomiast w kolumnie trzeciej podano funkcjonalności które są charakterystyczne dla narzędzi RLD. Odpowiadające sobie analogiczne funkcjonalności dostępne w obu grupach, zestawiono wierszami.

Tabela 4. Analogie pomiędzy narzędziami RAD i VI a „idealnym” narzędziem RLD

L.p.	Istniejąca funkcjonalność narzędzi RAD i VI	Postulowana funkcjonalność narzędzi RLD
1	Graficzny edytor okien – komponenty umieszczone w oknie podczas projektowania są wyświetlane w takiej postaci, jak będą wyglądać po kompilacji i uruchomieniu programu.	Graficzny edytor stron kursu – obiekty umieszczone na stronie podczas projektowania są wyświetlane w takiej postaci, jak będą wyglądać po eksporcie do pakietu i uruchomieniu kursu.
2	Komponenty używane do tworzenia okien programu mają postać ramki i są zgrupowane w paletce.	Obiekty używane do tworzenia stron kursu mają postać ramki i są zgrupowane w paletce.

L.p.	Istniejąca funkcjonalność narzędzi RAD i VI	Postulowana funkcjonalność narzędzi RLD
3	Wizualny edytor kodu sterującego działaniem programu ma postać diagramu przepływu danych – obiekty są reprezentowane przez bloczki, przepływ danych i sterowania jest symbolizowany przez linie łączące bloczki (wirtualne przewody).	Wizualny edytor kodu sterującego działaniem kursu, ma postać diagramu przepływu danych – obiekty są reprezentowane przez bloczki, przepływ treści i sterowania jest symbolizowany przez linie łączące bloczki (wirtualne przewody).
4	Edycja wyglądu i zachowania komponentów w oknie programu w czasie projektowania, odbywa się przez panel właściwości obiektu.	Edycja wyglądu i zachowania obiektów na stronie kursu w czasie projektowania, odbywa się poprzez panel właściwości obiektu.
5	Zarządzanie modułami, oknami i komponentami obydwoma się poprzez panel zarządzania projektem, który zawiera listy: (a) plików wykonywalnych, (b) modułów i okien w pliku wykonywalnym, (c) komponentów w oknie programu.	Zarządzanie tematami, lekcjami i stronami oraz obiektami na stronach obydwoma się poprzez panel zarządzania kursem, który zawiera listy: (a) tematów i lekcji w kursie, (b) stron w lekcji, (c) obiektów na stronie.
6	Kompilowanie gotowego programu do postaci wykonywalnej (np. plików EXE, DLL).	Eksport treści i pakowanie gotowego kursu do pakietu uruchamianego przez LMS (np. SCORM).
7	Testowanie działania aplikacji pod kontrolą narzędzia z użyciem debuggera.	Testowanie działania kursu pod kontrolą narzędzia z użyciem przeglądarki WWW.

Jednakże pomiędzy narzędziami RAD i VI a narzędziami do szybkiego tworzenia kursów istnieją pewne istotne różnice, a dotyczą one funkcjonalności, z których jedne można uznać za nadrzędne, inne za pomocnicze. Ilustruje to tabela 5.

Tabela 5. Różnice pomiędzy narzędziami RAD i VI a „idealnym” narzędziem RLD

Funkcjonalność	Narzędzia RAD i VI	Narzędzia RLD
Nadrzędna	Implementacja procedur programu z użyciem tekstowego edytora kodu (RAD) w wybranym języku programowania lub graficznego edytora diagramów przepływu danych i sterowania (VI).	Wizualne redagowanie treści kursu z użyciem graficznego edytora stron kursu: wprowadzanie i redagowanie tekstu, wstawianie grafiki i treści multimedialnych.
Pomocnicza	Wizualne projektowanie okien, wprowadzanie tekstu, wstawianie grafiki i treści multimedialnych.	Wizualne projektowanie diagramu akcji: nawigacja pomiędzy stronami, interakcja z obiektami.

Implementacja procedur programu w RAD lub tworzenie diagramu w VI jest nadrzędną funkcjonalnością, natomiast graficzne projektowanie okien, wprowadzanie tekstu i grafiki to operacje pomocnicze. Dokładnie odwrotnie jest w przypadku e-kursu. Tutaj najważniejsza jest funkcjonalność dotycząca wprowadzania i redagowania treści kursu, natomiast projektowanie diagramu akcji pomiędzy obiektami to funkcjonalność pomocnicza. Różnice te wynikają

z tego, że przeznaczeniem programu jest pobieranie i przetwarzanie różnych treści (danych), które mogą się zmieniać w trakcie jego działania. W przypadku e-kursu treści mogą nie ulegać zmianie po jego uruchomieniu przez użytkownika. Jednak w wielu obecnych narzędziach RLD skrypty wbudowane w pakiet e-learningowy przetwarzają niewielkie porcje danych wprowadzanych przez studenta, np. podczas rozwiązywania testów czy quizów. Wprowadzenie diagramów akcji rozszerzyłoby funkcjonalność e-lekcji. Skoro obecne narzędzia RLD generują skrypty obsługujące testy i quizy to mogłyby generować skrypty w oparciu o diagram akcji utworzony przez użytkownika. Tak wygenerowane skrypty, po wygenerowaniu pakietu i umieszczeniu w LMS mogłyby pobierać dane z plików zasobów, włączonych do pakietu (tekst, tabele, grafika). Zatem poszczególne strony e-lekcji nie miałyby charakteru statycznego jak klasyczne strony HTML, lecz część ich treści byłaby generowana stosownie do akcji użytkownika poprzez skrypty wbudowane w pakiet e-learningowy.

3.3. Zestawienie funkcjonalności „idealnego” narzędzia

Aby naprawdę możliwe było szybkie tworzenie kursu, „idealne” narzędzie winno zatem posiadać taki GUI (Graphic User Interface – graficzny interfejs użytkownika), który:

- grupuje i udostępnia wszystkie polecenia programu w jednym scentralizowanym miejscu – najlepiej w menu głównym, gdzie użytkownik zawsze znajdzie wszystkie polecenia dostępne w narzędziu,
- posiada uniwersalny i kontekstowy pasek narzędziowy, zawierający podstawowe operacje (otwórz, zapisz, cofnij, kopiuj, wklej, usuń itd.),
- pozwala na jak najszybsze wprowadzanie i modyfikację treści e-kursu – np. poprzez użycie paneli edycyjnych zamiast okien dialogowych,
- pozwala na dostosowanie GUI w przypadku ekranów o niższej rozdzielczości poprzez przemieszczanie i dokowanie oraz przełączanie paska narzędziowego i paneli edycyjnych przez użytkownika w zależności od potrzeb,
- pozwala na kontekstowe włączanie widoczności tylko tych paneli, które odpowiadają elementom kursu aktualnie wybranym przez użytkownika,
- wykonuje automatyczny zapis zmodyfikowanej treści kursu co pewien czas, z możliwością jego wyłączenia lub zmiany interwału,
- pozwala na kontekstowe wyświetlanie podpowiedzi i podglądu elementów po ich wskazaniu kursorem myszy, z możliwością jego wyłączenia,
- posiada kontekstowy panel przeznaczony do modyfikacji złożonych treści multimedialnych, np.: obróbki audio, wideo, animacji.

Bardzo poważnym udogodnieniem byłaby możliwość wprowadzania treści dowolnego rodzaju w samym narzędziu RLD. Przykładowo, wprowadzanie tekstu powinno być dostępne nie tylko poprzez jego wpisanie do aktywnej ramki za pomocą klawiatury alfanumerycznej, ale także poprzez:

- a. wstawianie tekstu z użyciem skanera i modułu OCR,
- b. odręczne wpisywanie tekstu z użyciem rysika tabliczki graficznej lub tabletu,
- c. dyktowanie tekstu głosem z użyciem mikrofonu oraz modułu analizy głosu i jego zamianę na tekst.

Funkcjonalność wymieniona w pkt. (a) jest stosowana od dawna w wielu typach aplikacji. Z kolei cecha wymieniona w pkt. (b) jest obecnie dostępna w kilku programach dedykowanych dla komputerów PC, a także jest powszechnie dostępna w tabletach i smartfonach. Natomiast cechą wymienioną w pkt. (c) w chwili obecnej obsługuje kilka edytorów tekstu, np.: MS Word (języku angielskim), a pierwszy raz taka możliwość pojawiła się w 1996 roku w pakiecie Lotus Smart Suite (wykorzystywał on moduł analizy głosu opracowany przez firmę Lernout & Hauspie). W takim przypadku program pełniłby funkcję sekretarza notującego dyktowany tekst.

Z kolei wprowadzanie grafiki statycznej winno być możliwe nie tylko przez wstawianie zawartości z pliku graficznego, ale także poprzez użycie skanera, aparatu fotograficznego lub rysowanie figur w aktywnej ramce z użyciem: myszy, rysika tabliczki graficznej lub tabletu. W chwili obecnej większość nawet dość prostych edytorów graficznych obsługuje tę funkcjonalność.

Osobnym problemem jest modyfikacja już istniejącej treści kursu. W tym przypadku narzędzie winno umożliwiać bezpośrednio wprowadzanie i redagowanie treści, które reprezentują dane naukowe lub techniczne a także multimedialne, np.:

- wyrażenia matematyczne,
- wykresy (np. XY, słupkowe, powierzchniowe, kołowe, biegunowe),
- podstawowe typy schematów (np. blokowe),
- podstawowe typy diagramów (np. Gantta, Venna, Ishikawy, Sankeya),
- proste rysunki techniczne,
- struktury chemiczne (wzory cząsteczek),
- komentarze głosowe (audio),
- nagrania wideo,
- sesje ekranowe.

Wprowadzanie i redagowanie treści obiektów mogłoby zostać zrealizowane z użyciem zagnieżdżonego kontekstowego edytora graficznego, uruchamiającego się po zaznaczeniu obiektu określonego rodzaju. Użyteczna byłaby możliwość wyboru formatu danych, jaki przechowywałby wymienione treści podczas eksportu kursu do pakietu, np. w standardzie SCORM. Przykładowo, dla wyrażen matematycznych możliwymi mogłyby być formaty MathML lub (jako tymczasowe rozwiązanie) SVG, GIF czy PNG. W przypadku treści multimedialnych ważna byłaby możliwość wykonania podstawowej obróbki audio i wideo, np.: regulacja głośności dźwięku, jasności obrazu, przycięcia nagrania, wyboru formatu kompresji podczas eksportu do pakietu SCORM.

4. Wnioski – terażniejszość i przyszłość RLD

Treści multimedialne i interaktywne umieszczane w kursach wymagają obecnie użycia odpowiednich narzędzi do ich zarejestrowania i obróbki. Zatem oprócz samego edytora (Authoring Tool) niezbędne jest stosowanie szeregu programów pomocniczych (np. do obróbki wideo, do przygotowania animacji SVG, Flash, itp.).

W typowych narzędziach przeznaczonych do tworzenia kursów brakuje (albo są one niewystarczająco funkcjonalne) rozwiązań pozwalających autorowi kursu na łatwe i szybkie wprowadzanie i modyfikowanie treści złożonych lub multimedialnych, a następnie ich eksport do pakietu zapisanego w standardzie SCORM. Wysoce pożądane byłoby ich obsługiwanie bezpośrednio w samym narzędziu, gdyż skróciłoby to czas i zmniejszyłoby poziom złożoności procesu tworzenia e-kursu, a także znacząco obniżyłoby koszty zakupu narzędzi niezbędnych do przygotowania kursów nasyconych treściami multimedialnymi.

Niektóre z narzędzi posiadają opcję importu gotowych pakietów SCORM. Narzędziem, które pozwala na import pakietu SCORM jest np. IBM Lotus Workplace Collaborative Learning Authoring Tool. Taka opcja wydaje się być przydatna w sytuacji, gdy źródło e-kursu nie jest dostępne. Jednak pakiety zawierające bardziej złożoną zawartość, nie zawsze są importowane bez błędów lub braków.

Wciąż wiele narzędzi podczas eksportu kursu do pakietu SCORM zamienia rysunki (grafika statyczna) na grafikę rastrową (PNG, JPG) zamiast na wektorową (SVG – w wielu wypadkach takie rysunki mają mniejszy rozmiar, a w przypadku skalowania są czytelniejsze). Żadne nie pozwala na tworzenie złożonych, graficznych i tekstowych egzaminów poza prostymi testami i quizami. Do tego większość z nich jeszcze nie obsługuje standardu TinCan API (Rustici Software, b.d.). Daje się jednak zauważyć pewne zmiany i powoli wykształca się pewien standard budowy oraz funkcjonowania tych aplikacji, podobnie jak nastąpiło to kiedyś dla innych typów

narzędzi informatycznych (np.: RAD, CAD, DTP). Zapewne za kilka lat niektóre z wymienionych wcześniej programów zostaną przekształcone w prawdziwe, rozbudowane narzędzia do szybkiego tworzenia kursów i pojawią się w nich cechy postulowane przez autorów niniejszego artykułu.

5. Bibliografia

1. Atomi Systems. (2014). *ActivePresenter User's Manual Version 4.0*. Pobrane z: http://atomisystems.com/apdownloads/ActivePresenter4.0_UserManual.en.pdf
2. Impatica Inc. (2012). *Impatica for PowerPoint 5.0. User Manual*. Pobrane z: <http://www.impatica.com/products/imp4ppt/manual/Imp4ppt5-User-Manual.pdf>
3. Lakens, D. (2005, marzec). *Authorware Beginner's Guide for Building Experiment*. Pobrane z: [http://dtserv2.compsy.uni-je-na.de/_C1257C12004D57F7.nsf/0/FF88E82DDF5088EAC1257C2200379248/\\$FILE/Authorware%20Beginners%20Guide.pdf](http://dtserv2.compsy.uni-je-na.de/_C1257C12004D57F7.nsf/0/FF88E82DDF5088EAC1257C2200379248/$FILE/Authorware%20Beginners%20Guide.pdf)
4. *MathML*. (2014, październik). Pobrane z: <http://caniuse.com/#feat=mathml>
5. Mobile HTML. (2014). *HTML5 compatibility on mobile and tablet browsers with testing on real devices*. Pobrane z: <http://mobilehtml5.org>
6. Office Team. (2014, 6 maja). *Meet Office Mix*. Pobrane z: <http://blogs.office.com/2014/05/06/meet-office-mix>
7. Rustici Software. (b.d.). *What is the Tin Can API?* Pobrane 8 grudnia 2014, z: <http://tincanapi.com/overview>
8. SumTotal Systems. (b.d.). *The New ToolBook 11.5*. Pobrane 8 grudnia 2014, z: <http://tb.sumtotalsystems.com/index.html>
9. Thompson, K. (2014, 26 stycznia). *Introducing the Lotus Workplace Collaborative Learning Authoring Tool*. Pobrane z: <http://www.ibm.com/developerworks/lotus/library/LMS-LWP>
10. Wayner, P. (2010, 8 grudnia), *HTML5 in the browser: Canvas, video, audio, and graphics*, InfoWorld. *Info World*. Pobrane z: <http://www.infoworld.com/d/application-development/html5-in-the-browser-canvas-video-audio-and-graphics-808>
11. WebSoft. (b.d.). *CourseLab 2.7. User Manual*. Pobrane 8 grudnia 2014, z: <http://store.courselab.com/en-US/Course-Lab.2.7.en-US.pdf>
12. World Wide Web Consortium. (2014). *Standards for Web Applications on Mobile: current state and roadmap*. Pobrane z: <http://www.w3.org/Mobile/mobile-web-app-state>

Analysis of the properties, functionalities and ergonomics of modern tools dedicated to rapid e-learning development

Summary

Keywords: e-learning, e-course, rapid e-learning development, RLD

Article contains an analysis of the rapid e-learning development tools. It was based on the experience gained by the authors during implementation of a series of trainings for Rzeszow University of Technology teachers in the field of e-learning. Publication contains insights that emerged mainly during the practical part of the training. Potential problems and difficulties encountered by people who create e-learning courses using currently available tools for rapid e-learning development were discussed. The main emphasis was put on the analysis of the features, ergonomics, functionalities and also imperfections of the most popular RLD tools. It is-sued a set of characteristics that define „ideal” tool. In assessing existing tools, ex-aminated the features that allow to incorporate the latest web technologies in the pro-cess of creating e-courses.