

DEVELOPMENT AND MANAGEMENT SYSTEM FOR WEB APPLICATIONS AND WEBSITES

George-Alex STELEA, Cristinel GAVRILĂ

Ph.D. Students. Eng., Transilvania University of Brasov, e-mail: george.stelea@unitbv.ro, cristinel.gavrila@unitbv.ro
Coordinator: prof. univ. dr. ing. Florin SANDU

Abstract

The World Wide Web is accessed by a plurality of devices at different connections using multiple operating systems. In this paper we propose a solution for the development and management of web applications and websites in order to optimize the programming, administration and management time, using modules, techniques and preintegrated methods to simplify the procedures associated with common and repetitive activities.

Key words: web development system, web application management, back-end development technologies, front-end development technologies.

1. Introducere

Internetul a fost creat ca o soluție la nevoia de comunicare și a devenit acum cel mai utilizat mecanism în această privință. Datorită vitezei de transmitere a datelor și a costurilor reduse, Internetul elimină limitele temporale și spațiale în comunicare, interacționare, monitorizare în timp real [1] și activitățile de afaceri.

Varietatea de rezoluții, browsere și capacități de procesare generează necesitatea ca o aplicație sau un site web să fie accesibil atât din punct de vedere al conținutului, cât și al calității. Interacțiunea cu interfețele se realizează atât în modul clasic, folosind tastatură și mouse, cât și cu ajutorul tehnologiilor touchscreen, în ambele cazuri experiență de utilizare trebuie să fie confortabilă și practică. Integrarea tehnologiilor moderne, cum ar fi principiul de reactivitatea, viteza de navigare, secțiunile video și optimizarea pentru motoarelor de căutare, pot spori interactivitatea cu utilizatorul, în timp ce tehnologiile dedicate compatibilității inter-platformă pot asigura accesibilitate de ultimă oră în mediul online.

Este important să se testeze și să se utilizeze tehnologii compatibile cu toate platformele moderne și să se evite utilizarea

1. Introduction

The Internet has been created as a solution to the need for communication and has now become the most used mechanism in this regard. Due to the speed of data transmission and low costs, the Internet eliminates temporal and spatial boundaries in human communication, interaction, real-time monitoring [1] and business activities.

The variety of resolutions, browsers and processing capability generates the need for an application or a website to be accessible both as content and quality. Interacting with the interfaces in both the classic mode, with keyboard and mouse, or using touchscreens causes the demand to offer a comfortable and practical user experience. Integrating modern technologies like responsiveness, navigation speed, video sections and a search engine optimization structure, can increase the interactivity with the user, while using technologies that are cross-browsing compatibles can assure state of art accessibility in the online environment.

It is important to test and use technologies that are cross-browsing compatibles, and to avoid the use of new or old technologies that can't be accessible to most of the online users. It is futile to develop a software project if it

tehnologiilor fie prea noi sau vechi, care nu pot fi accesibile majorității utilizatorilor online. Este inutil să se dezvolte un produs software dacă acesta poate fi afișat în mod corespunzător doar pe un număr limitat de dispozitive. Succesul unui proiect depinde direct de importanța acordată stabilirii scopurilor și obiectivelor sale. Planificarea optimă a activităților este condiționată de un scop clar care trebuie să fie stăpanit de toți membrii echipei de dezvoltare.

Sistemul își propune să asigure un flux de lucru adecvat, utilizând tehnologii de ultimă generație, atât pe partea de back-end (Server Side Scripting, structuri de baze de date, biblioteci și module) cât și tehnologii front-end (HTML, CSS, cu JavaScript și grafica web) pentru dezvoltarea și mentenanța aplicațiilor, scalabile și compatibile cu platformele moderne, cât și pentru optimizarea timpului de programare, administrare, gestionare și pentru utilizarea mai eficientă a resurselor.

2. Conceptul și funcționalitatea sistemului de dezvoltare și management

Etapele dezvoltării unei pagini web nu pot fi gândite separat, deoarece atunci când se construiește un site web este nevoie de un flux de lucru dinamic cu multe recurențe către etapele anterioare [2]. Un site sau o aplicație web atractivă și dinamică necesită întotdeauna un limbaj de programare solid și puternic. PHP, Java, Python, Ruby sau .NET pot fi alese în dezvoltare pentru a obține funcționalitatea de back-end dorită.

Există mai multe abordări în privința utilizării unui limbaj de programare pentru dezvoltarea web. Două dintre cele mai importante abordări sunt programarea procedurală sau programarea orientată pe obiecte. De obicei, pentru site-urile mici, dezvoltatorii aleg să utilizeze programarea procedurală și nu programarea orientată pe obiecte, care necesită o mai mare complexitate și control. Pentru aplicațiile web/site-uri web care necesită o structură simplă și o scalabilitate redusă, utilizarea

can be properly displayed only to a limited devices. The success of a project depends directly on the importance given to establishing its goals and objectives. Good planning of activities is conditional on a clear purpose that needs to be mastered by all members of the development team.

The system aims to ensure a proper workflow and the use of appropriate, both back-end technologies (Server Side Scripting, Database Structures, Libraries and Plugins) and front-end technologies (HTML, CSS, with JavaScript and Graphics) to develop and maintain solid, scalable and cross-platform compatible applications, to optimize the programming, administration and management time and to make more efficient use of resources.

2. Development and management system concept and functionality

The stages of developing a web page cannot be thought separately because when building a website is required a dynamic workflow with many recurrences to earlier stages [2]. An attractive and dynamic website or web application always require a solid and strong programming language. PHP, Java, Python, Ruby or .NET can be chosen for development in order to achieve back-end functionality.

There are several different approaches in using a programming language for web development. Two of the most important approaches are procedural programming or object-oriented programming. Usually for small websites web developers choose to use procedural programming and not object-oriented programming that requires more complexity and control. For web application/websites requiring a simple structure and a reduced scalability the use of object-oriented programming can lead to unnecessary complexity and pointless consumption of resources. Different development environments, programming languages and programming type are preintegrated in the system in order to assure

programării orientate pe obiecte poate duce la o complexitate inutilă și la un consum inutil de resurse. Diferitele medii de dezvoltare, limbaje de programare și tipuri de programare sunt preintegrate în sistem pentru a asigura libertatea de alegere pentru fiecare dezvoltator.

O aplicație online sau un site web este construit cu un scop, ar putea fi unul comercial, informativ sau educativ, iar dacă nu își atinge scopul, ar putea fi lipsit de valoare [3]. Este foarte important să se țină seamă de publicul țintă căruia i se adresează. În acest fel, se poate analiza ce fel de design, accesibilitate și tehnologii vor fi folosite. O gamă largă de tipuri de aplicații / site-uri web populare și regulat utilizate sunt prezente în configurația inițială a fiecărui proiect pentru a oferi o structură definită și o funcționalitate testată și scalabilă. Procesul de configurare se face printr-o interfață grafică de utilizator, așa cum este prezentat în Fig. 1, unde sunt afișate zece tipuri de aplicații web, însă acest număr poate fi extins, dacă se dorește acest lucru.

freedom of choice for each developer.

An online application or website is built with a purpose, it could be commercial, informational or educational and if it does not accomplish its goal it could be worthless [3]. It is very important to take into account the target audience to whom it is addressed. In this way it can be analyzed what kind of design, accessibility and technologies will be used. A range of popular and common used applications/websites type are configured in the initial setup of each project in order to provide a defined structure and a tested and scalable functionality. The configuration process is made through a graphical user interface as shown in Fig. 1, where ten types of web applications are displayed, but this number can be extended if desired.

System Control Panel

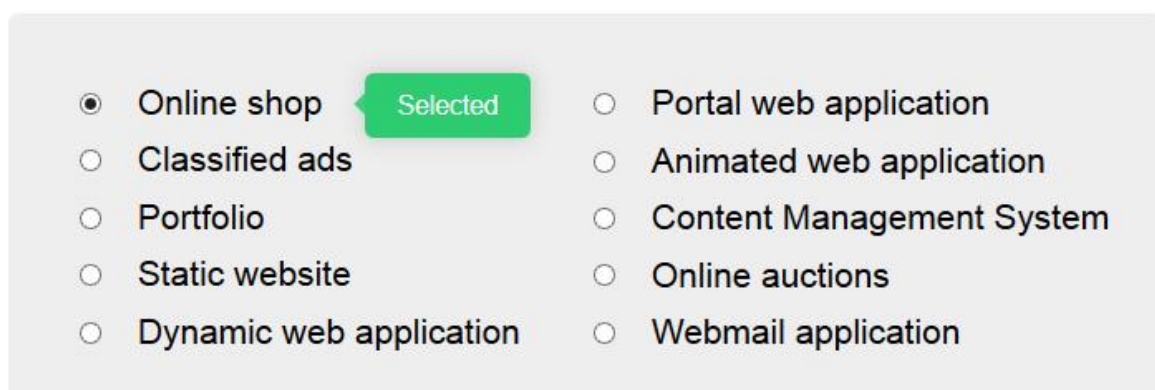


Fig. 1

Pentru gestionarea corectă a unei aplicații web se utilizează de obicei un sistem de template-uri web, acest sistem combinând mai multe șabloane pentru a genera o pagină web îmbunătățită. Secțiunile care sunt identice în mai multe pagini sunt stocate într-un singur fișier care poate fi apelat în

For proper management of a web application it is typically used a web templating system, this system combines several templates to generate an enhanced web page. The sections that are identical in more pages are stored in a single file that can be called in the main template using built-in functions or using

șablonul principal utilizând funcții încorporate sau motoare de șablonare cu diverse modele arhitecturale [4] (de exemplu: MVC-Model View Controller).

Șabloanele pot fi utilizate pentru a genera pagini statice și dinamice în funcție de necesitățile dorite. În Fig. 2 este prezentată interfața grafică de utilizator pentru a selecta din două tipuri de machete/șabloane preconfigurate pentru scopuri de comerț electronic.

template engines with various architectural pattern [4] (for example: MVC - Model View Controller).

Templates can be used to generate both static and dynamic pages according to the desired needs. In Fig. 2 is presented the graphical user interface to select from two preconfigured layouts/templates types for e-commerce purpose.

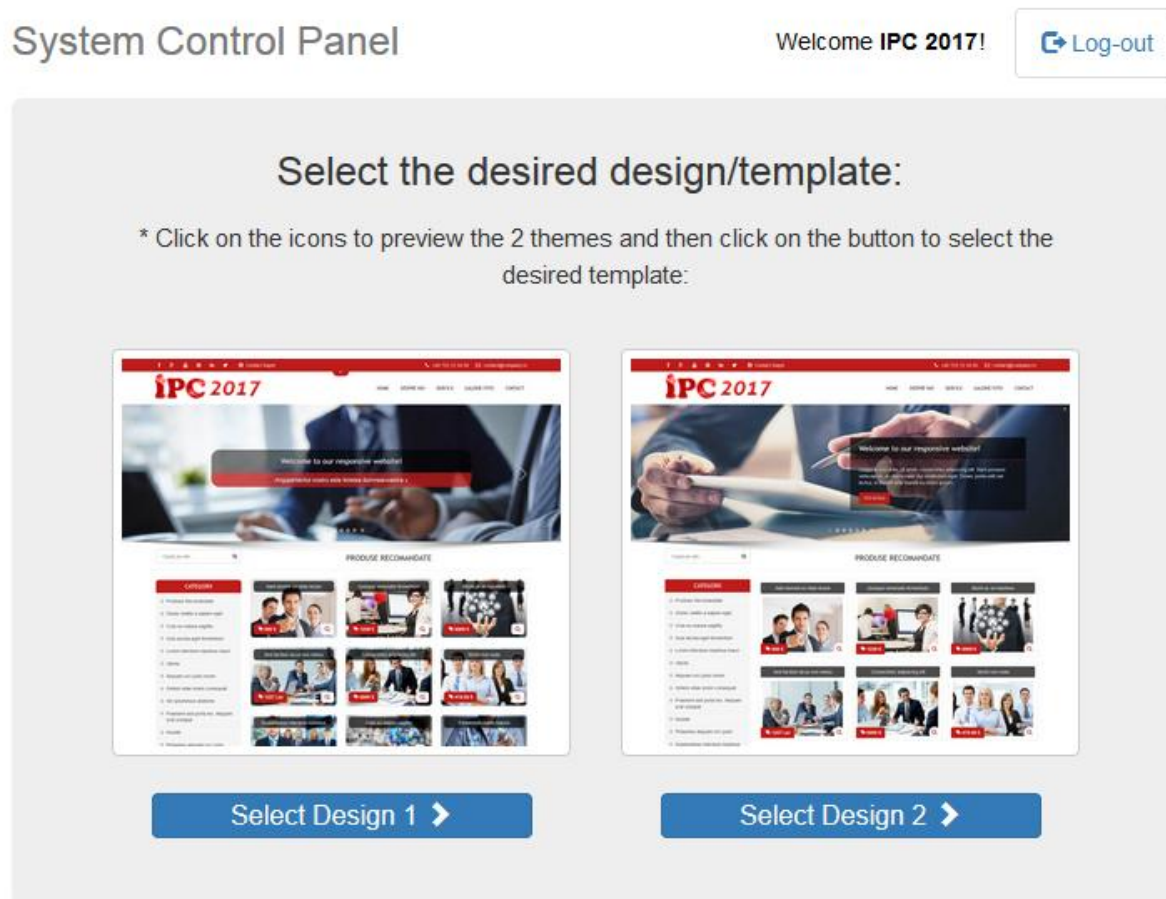


Fig. 2

Indiferent de tipul de aplicație, o alternativă la programarea personalizată este utilizarea de biblioteci, module și plugin-uri (open-source sau comerciale). Utilizarea unor astfel de tehnologii reduce foarte mult costurile de dezvoltare și întreținere, chiar dacă un proiect este continuat de un alt

Regardless of the type of application an alternative to personalized programming is the use of libraries, modules and plugins (open-source or commercial). Using such technologies greatly reduces development and maintenance costs, even if a project is continued by another web developer than the

dezvoltator web decât cel care a construit proiectul inițial.

Sistemul dispune de biblioteci, module și pluginuri preinstalate, testate și optimizate, astfel încât nu mai este necesară instalarea și configurarea manuală a acestora, așa cum se arată în Fig. 3.

one who built the initial project.

The system has pre-installed libraries, modules and plugins, tested and optimized, so that it is no longer necessary to install and manually configure them, as shown in Fig. 3.

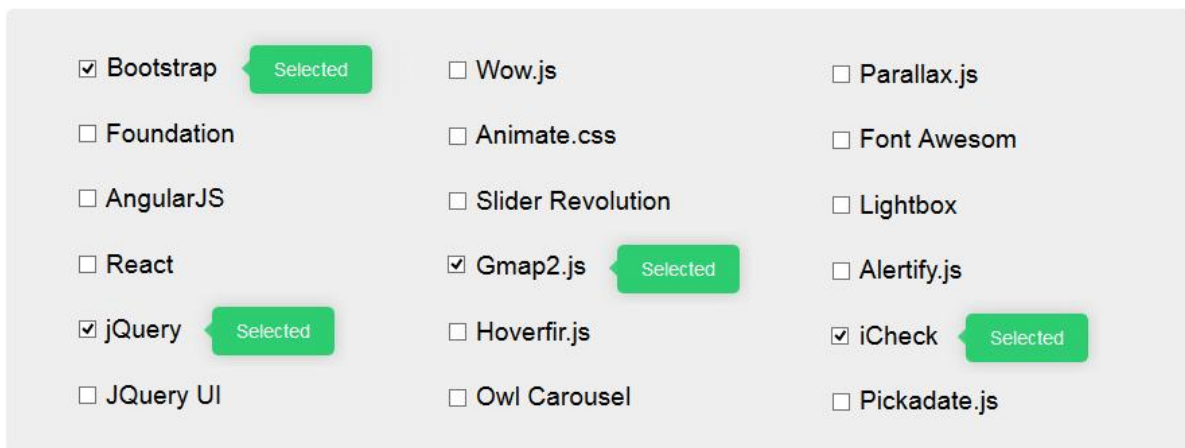


Fig. 3

O altă caracteristică a sistemului propus este prezența secțiunilor pre-integrate, frecvent utilizate precum "date de contact", "galerii foto/video" și "bine ați venit/despre noi". Aceste secțiuni pot fi activate sau dezactivate utilizând meniuri de selecție așa cum sunt prezentate în Fig.4.

Another feature of the proposed system is the presence of pre-integrated commonly used sections like "contact data", "photo/video galleries" and "welcome/about us". These sections can be activated or deactivated using select menus as presented in Fig.4.

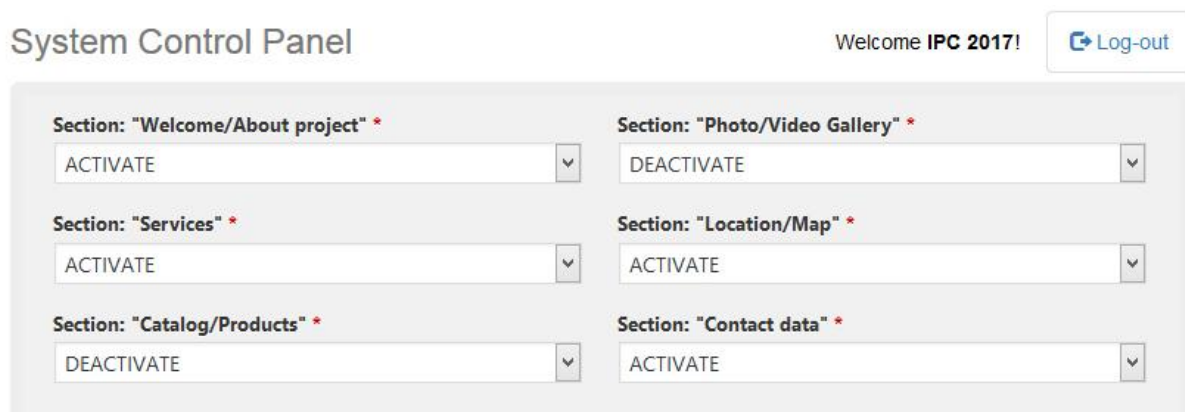


Fig. 4

Chiar examinate separat, paginile unui site web trebuie să fie ușor de recunoscut: au aceleași culori, aceleași simboluri, același logo, același antet, același subsol, folosesc aceleași fonturi și sunt împărțite în mod egal. Site-ul sau aplicația este omogenă, unitară și este prezentată în ansamblu. Un design bun ar trebui să transmită mesaje, să comunice cu vizitatorii și să îndeplinească obiectivele pentru care a fost creat.

În această direcție a fost adăugat un încărcător de imagini (ex: logo) și un selector de culori pentru a alege culoarea diferitelor secțiuni din partea frontală a aplicației. Această funcționalitate este prezentată în Fig.5.

Even examined separately, the pages of a website must be readily recognizable: they have the same colors, the same symbols, the same logo, the same header, the same footer, they use the same fonts and are equally divided. The website or application is homogeneous, unitary and is presented as a whole. A good design should convey messages, communicate with visitors and meet the objectives for which it was created.

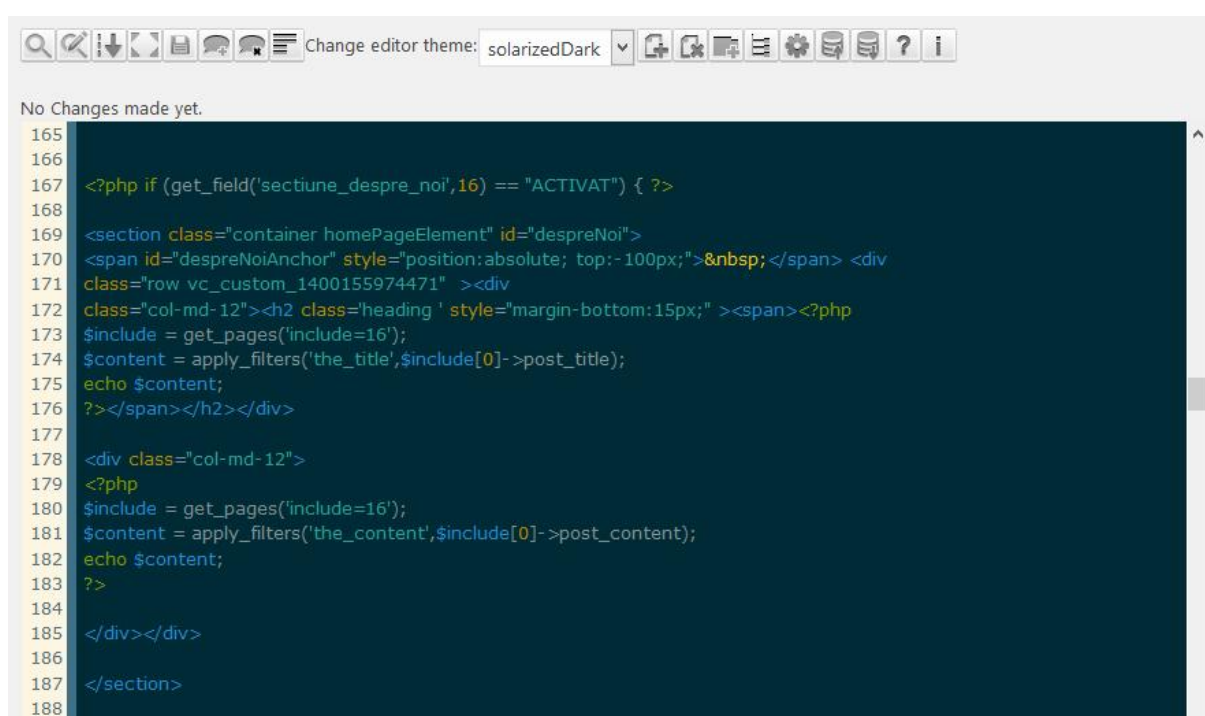
In this direction we added an image uploader (ex: logo) and a color picker to choose the color of the different sections in the front end part of the application. This functionality is presented in Fig.5.



Fig. 5

După configurarea inițială, un sistem IDE (Integrated Development Environment) este încorporat în sistem pentru a putea dezvolta alte caracteristici personalizate utilizând limbajul de programare dorit și preconfigurat. IDE-ul este prezentat în Fig. 6 și este special construit pentru a extinde scalabilitatea și funcționalitatea proiectelor și pentru a evita limitările tehnice.

After initial configurations, an IDE (Integrated Development Environment) is incorporated in the system in order to be able to develop other custom features using the desired and preconfigured programming language. The IDE is presented in Fig. 6 and is specially built to extend the scalability and functionality of the projects and to avoid technical limitations.



```
165
166
167 <?php if (get_field('sectiune_despre_noi',16) == "ACTIVAT") { ?>
168
169 <section class="container homePageElement" id="despreNoi">
170 <span id="despreNoiAnchor" style="position:absolute; top:-100px;">&nbsp;  </span> <div
171 class="row vc_custom_1400155974471" ><div
172 class="col-md-12"><h2 class="heading" style="margin-bottom:15px;" ><span><?php
173 $include = get_pages("include=16");
174 $content = apply_filters('the_title',$include[0]->post_title);
175 echo $content;
176 ?></span></h2></div>
177
178 <div class="col-md-12">
179 <?php
180 $include = get_pages("include=16");
181 $content = apply_filters('the_content',$include[0]->post_content);
182 echo $content;
183 ?>
184
185 </div></div>
186
187 </section>
188
```

Fig. 6

3. Concluzii

Când se începe dezvoltarea unui software web, este foarte important să se planifice scalabilitatea viitoare și lățimea de bandă oferită.

În cazul în care este necesară dezvoltarea și extinderea în continuare a funcționalității aplicației, alegerea tehnologiilor greșite poate provoca nevoia reconstruirii acesteia de la zero. Kevin Skoglund, senior trainer la Lynda.com, a spus că atunci când vine vorba de construirea unui site web: "Este foarte

3. Conclusions

When starting developing a web software it is very important to plan its future scalability and bandwidth. If it is needed to further develop and grow the functionality of the web application, choosing the wrong technologies might cause the need of rebuilding it from the start. Kevin Skoglund, senior trainer at Lynda.com said that when coming to build a web website: "It's very important to plan an adequate workflow and use proper technologies to achieve necessary

important să se planifice un flux de lucru adecvat și să se folosească tehnologii corespunzătoare pentru a obține simplitatea necesară dorită și a se evita complexitatea inutilă"[5].

Sistemul de dezvoltare și management propus oferă un spectru larg de aplicabilitate într-o gamă largă de aplicații compatibile cu platformele moderne (magazine online, site-uri statice, aplicații web dinamice, aplicații de tip web portal, aplicații web animate, site-uri pentru anunțuri, portofolii, aplicații de tip webmail, wiki etc.).

În funcție de tipul de aplicație dorit, timpul de programare, administrare și gestionare poate fi optimizat până la mai mult de 50%, iar după configurarea inițială, datorită modulului IDE (Integrated Development Environment), sistemul poate fi folosit pentru a dezvolta aplicații avansate și personalizate.

simplicity and avoid unnecessary complexity"[5].

The proposed development and management system offers a wide spectrum of applicability in a range of cross-platform compatible applications (online shop, static website, dynamic web application, portal web application, animated web application, classified ads, portfolio, webmail application, wiki etc.).

Depending on the type of application, the programming, administration and management time can be optimized up to more than 50% and after the initial setup, thanks to the IDE (Integrated development environment) module, the system can be used to develop even advanced custom applications.

References

- [1] Iulian Iliescu, Titus Bălan, Oana Garoiu, Sorin Zamfir (2016) *Thin clients for real-time monitoring of communication infrastructure*. Intl Conf on Scientific Research and Education in the Air Force-AFASES. DOI: 10.19062/2247-3173.2016.18.1.35.
- [2] Deane Barker (2016) *Web Content Management:Systems, Features and Best Practices*, O'Reilly Media. ISBN: 978-1491908129.
- [3] Kristina Halvorson (2012) *Content Strategy for the Web*, New Riders. ISBN: 978-321808301.
- [4] Mark Richards (2015) *Software Architecture Patterns*, O'Reilly Media, Inc.
- [5] Kevin Skoglund (2009) *PHP with MySQL Beyond the Basics*, Lynda.com, Inc ISBN:1596715413 9781596715417.