

UTILIZAREA SISTEMELOR DE MANAGEMENT AL ÎNVĂȚĂRII LA FORMAREA COMPETENȚEI DE PROGRAMARE ORIENTATĂ PE OBIECTE

Andrei BRAICOV, doctor, conferențiar universitar

Ala GASNAȘ, lector superior

Catedra ITI, Universitatea de Stat din Tiraspol

Rezumat. În acest articol sunt prezentate reflecții privind utilizarea Sistemelor de Management al Învățării (SMÎ) la studierea Programării Orientate pe Obiecte (POO). Sunt determinate avantajele acestei abordări educaționale și caracteristicile care garantează eficientizarea formării și dezvoltării la studenți a conceptului POO cu ajutorul instrumentelor SMÎ.

Cuvinte cheie: concept de programare orientată pe obiecte, sisteme de management al învățării, MOODLE, tehnologii de predare.

USING OF THE LEARNING MANAGEMENT SYSTEMS IN THE TRAINING OF COMPETENCE OF OBJECT ORIENTED PROGRAMMING

Abstract. This article examine the use of Learning Management Systems (LMS) in studying Object Oriented Programming (OOP). The advantages of this educational approach and the characteristics that guarantee the efficiency of student training and development of the OOP concept with the help of LMS tools are determined.

Keywords: Object Oriented Programming concept, Learning Management Systems, MOODLE, teaching technologies.

Dezvoltarea vertiginoasă din ultimii ani a domeniilor Informatică și Tehnologii Informaționale a furnizat o „foame” monoton-crescândă de specialiști în domeniul programării atât în plan cantitativ, cât și în cel calitativ, la nivel național și internațional.

Această ultimă necesitate continuă să impună reactualizarea dens-periodică a curriculei programelor de studii (ciclurile licență și master) la Informatică și Tehnologii Informaționale, dar și revederea formelor și instrumentelor de organizare a instruirii studenților din aceste domenii.

În această ordine de idei subiectele legate de metodologia pregătirii specialiștilor în arealul Informatică și Tehnologii Informaționale au constituit în ultimii ani obiectele cercetării mai multor savanți: S. Hadjerrouit (Norvegia), Keefe Karen (Australia), V. Cabac, S. Gîncu, L. Mihălache, T. Velicova, L. Chiriac, N. Deinego, C. Negară (Republica Moldova) etc.

Există o varietate largă de subdomenii ale Informaticii: Rețele de calculatoare și arhitecturi, Sisteme de operare, Limbaje de programare, Tehnologii Web, Algoritmă, Securitate informațională, Baze de date etc. Acestea nu au un hotar bine conturat, astfel încât abilitățile și competențele profesionale ale unui bun specialist-informatician se referă la câteva subdomenii.

Dilema alegerii-elaborării celor mai adecvate curricule și metodologii de pregătire a unui specialist bun, competent și competitiv pe piață rămâne actuală.

Așa cum majoritatea sistemelor de operare și mediile de dezvoltare a produselor-program sunt bazate pe tehnologia *Programării Orientate pe Obiecte* (POO), conchidem că deținerea competențelor de POO este o caracteristică absolut necesară informaticianului profesionist modern. Competențele POO sunt solicitate în majoritatea subdomeniilor Informaticii menționate anterior.

Competența de Programare Orientată pe Obiecte (CPOO) reprezintă un sistem de cunoștințe, abilități, deprinderi și atitudini, numite resurse, bine structurate și temeinic însușite, prin intermediul cărora individul va putea acționa, în baza conceptelor POO, pentru soluționarea/tratarea anumitor situații specifice [1].

În pofida faptului că POO a detronat programarea procedurală (fiind considerată mai aproape de modelarea situațiilor din viața reală) și, prin esența ei, a condus la transparență în proiectarea și elaborarea produselor program, predarea POO și formarea CPOO sunt mult mai dificile decât formarea deprinderilor și abilităților de programare procedurală.

Motivul acestei dificultăți constă în faptul că pentru predarea POO și formarea CPOO sunt necesare abordări educaționale și metode de predare-învățare și evaluare noi (pe alocuri radicale).

Care sunt problemele, dificultățile întâlnite în predarea și învățarea POO?

În [1] se menționează despre lipsa unui concept comun de selectare a conținuturilor educaționale la disciplina POO și utilizarea diferitor limbaje și medii pentru studierea POO.

Keefe Karen enumeră în [2] problemele care apar la studenți în formarea competenței de programare: descompunerea unei probleme; înțelegerea mecanismului de programare; transferul de parametri; returnarea valorilor de către metode; chiar și declararea variabilelor etc.

În cazul POO Keefe Karen susține că studenții întâmpină dificultăți în înțelegerea și utilizarea celor mai esențiale caracteristici POO, cum ar fi moștenirea, abstractizarea, clasele, polimorfismul etc.

Pe lângă obiectivele de bază, formulate tradițional în curricula din programele universitare, companiile care recrutează specialiști-informaticieni enumeră un șir de competențe specifice legate de POO pe care trebuie să le posede aceștia [3], cum ar fi:

- ✓ Conceperea proiectelor pe baza paradigmei obiectuale cu accent pe interacțiunea cu comunitatea;
- ✓ Programare obiectuală multi-arhitectură;
- ✓ Conceperea programelor cu un limbaj POO utilizând un proiect ce necesită o adaptabilitate crescută etc.

Următoarele argumente vin să confirme afirmația necesității deținerii competenței POO:

- mediile moderne de dezvoltare a produselor-program sunt bazate pe principiul POO;
- sistemele de operare actuale sunt elaborate, de asemenea, utilizând POO;
- limbajele de programare de nivel înalt (inclusiv a celor de elaborare a aplicațiilor web) au implementat conceptul POO și sunt susținute de un șir de biblioteci de obiecte predefinite;
- elementele/componentele unei aplicații (inclusiv web) sunt tratate în codul de program, de regulă, drept obiecte;
- conceptul POO permite optimizarea procesului de programare (în special la extinderea produselor-program, la dezvoltarea proiectelor mari);
- programarea obiectelor și a relațiilor dintre ele are similitudini cu proiectarea obiectelor din viața reală și descrierea unor fenomene complexe reale etc.

S. Hadjerrouit [4] descrie un model pedagogic bazat pe principiile epistemologiei constructiviste pentru predarea Programării Orientate pe Obiecte. Acest model are la bază următoarele șapte principii:

1. Cunoștințele în domeniul POO trebuie să fie construite în mod activ de către elevi, nu transmise în mod pasiv de către profesori.
2. Programul de instruire trebuie să fie ghidat de conceptele POO, nu de aspectele tehnice al limbajului.
3. Dacă cunoștințele anterioare ale studenților despre programare sunt în conflict cu abordarea POO, atunci ele trebuie revizuite.
4. Conceptele POO trebuie să fie bine asociate cu problematica și limbajul utilizat în predare, care, la rândul lor, trebuie să se fie strâns legate între ele.
5. Procesul de producere a cunoștințelor orientate pe obiecte puternic asociate necesită anumite abilități în rezolvarea problemelor.
6. Modalitățile tradiționale de predare, cum ar fi prelegerile, trebuie înlocuite cu un set de activități în care elevii sunt implicați activ în dobândirea cunoștințelor.
7. Pentru a obține implicarea activă a elevilor în rezolvarea problemelor, activitățile trebuie să se concentreze în jurul unui set de probleme reale, motivaționale.

Pornind de la faptul că Sistemele de management al Învățării (SMÎ) au fost concepute în ideea organizării constructiviste a procesului educațional, înțelegem că aceste șapte principii de studiere a POO se potrivesc reușit cu obiectivele unui curs electronic de POO, organizat pe un SMÎ (de exemplu Moodle).

Potrivit lui M. Kölling [5] problemele de predare POO apar din cauza alegerii proaste a limbajului de programare și a materialelor didactice. În același timp el consideră că un *mediu* adecvat de învățare POO trebuie să sprijine șapte caracteristici-cheie: simplitatea în utilizare; instrumente integrate; obiecte de suport; reutilizarea codului suport; tutelarea învățării; echipe de sprijin; disponibilitatea.

Răspândirea Internetului a contribuit la o varietate de materiale de învățare on-line, care ajută la dobândirea unui volum mare de cunoștințe-informații și, datorită interactivității, antrenează utilizatorul în formarea anumitor abilități și competențe.

Așa cum motivația, îndeosebi cea intrinsecă, catalizează studenții spre învățare, iar curiozitatea reprezintă baza acestui tip de motivație, Internetul este mediul cel mai prolific în furnizarea subiectelor de interes înalt, dar și pentru realizarea unor cooperări între utilizatori prin intermediul canalelor de comunicare (de exemplu, rețelelor de socializare). Aceste atu-uri, persistente și în SMÎ, au fost completate cu posibilități de control a informației. Un SMÎ identifică utilizatorii și le oferă drepturi-acces la resurse educaționale relevante.

SMÎ permite, de rând cu educația tradițională (în care rolul cadrului didactic este esențial), organizarea instruirii mixte (în engleză „*blended learning*”), ori astfel, se poate ajunge la o predare de calitate. Instruirea mixtă este una dintre cele mai eficiente metode de predare a programării. Ea permite combinarea diferitor modalități de prezentare a materialului didactic, folosind tehnici de management a cunoștințelor și organizarea eficientă a timpului atât al cadrelor didactice, cât și al studenților. În așa mod, procesul de învățare devine mai interesant și accesibil.

SMÎ este un mediu propice pentru învățarea constructivistă. Biggs J. B. consideră că învățarea are loc prin comportamentul activ al studentului: învățarea este ceea ce face el, ce învață el (studentul), nu ceea ce face profesorul.

Cele șapte principii ale lui Hadjerrouit, enumerate mai sus, se potrivesc cu obiectivele unui curs de POO, predarea căruia se face cu ajutorul unui SMÎ.

Pe parcursul anilor 2015-2017 a fost realizată o cercetare privind eficiența utilizării SMÎ Moodle în procesul de studiu al POO (în cadrul Universității de Stat din Tiraspol).

Grație oportunităților de învățare flexibilă oferite de SMÎ Moodle și a abordării constructiviste (cu accent pe explorarea practică și experimentală), studenții (în pofida pre-achizițiilor de diferit nivel) au salutat forma propusă de organizare a instruirii. O parte a resurselor educaționale digitale pentru cursul POO a constituit-o baza teoretică (arsenalul de concepte, noțiuni și definiții etc.), cealaltă parte – probleme (pe modele reale), a căror soluționare implică utilizarea POO.

Subiectele principale au fost organizate în module, ceea ce a permis o mai bună asimilare a conținutului prin vizualizarea secvențială a materialului digital.

Utilizarea SMÎ Moodle a permis studenților să acceseze cursul ori de câte ori a fost necesar, ajutați în obținerea succeselor de comunicarea sincronă și asincronă (colegială și cu profesorul-tutore).

Testările (îndeosebi cele inițiale) au fost utile pentru determinarea gradului de înțelegere de către studenți a subiectului de învățare, precum și pentru identificarea aspectelor de predare-învățare care urmează să fie îmbunătățite.

S-a constatat că, așa cum nivelul abilităților de programare este direct dependent de cardinalul mulțimii problemelor rezolvate, a algoritmilor elaborați de către studenți, este necesar ca aceștia să fie antrenați în activități practice atât în timpul orelor de contact direct, cât și în timpul destinat lucrului individual. În același timp problemele (proiectele) propuse trebuie să fie relevante, cu modele reale, care de cele mai dese ori cer implicarea câtorva rezolvatori (cu diferite roluri).

În același timp observăm că una dintre cele mai importante considerente privind crearea unei resurse de învățare electronice este identificarea clară a tipului de utilizatori ai acestor resurse. SMÎ, de regulă, sunt utilizate cu succes de către studenții alfabetizați tehnologic. Majoritatea studenților-informaticieni sunt autonomi, independenți și familiarizați cu instrumentele Web 2.0, deci sunt predispuși pentru activități colaborative.

Așadar, s-a constatat că SMÎ Moodle susține reușit cele două considerente: depozitarea-furnizarea unei game largi de resurse digitale (texte, prezentări, materiale interactive audio-video etc), probleme-proiecte și organizarea activităților colaborative de învățare-cercetare. Evident, rolul profesorului, de asemenea, se extinde, îndeosebi spre componenta *ghidare*, inclusiv extra-auditorială.

SMÎ facilitează administrarea întregului proces de studiu prin asistența de administrare, verificare, evaluare, urmărire de progres etc.

Concluzii

Utilizarea platformelor SMÎ pentru învățare este un imperativ al timpului, o abordare care este conformă generației actuale de studenți.

Studenții domeniului Informatică apreciază pozitiv implicarea instrumentelor și tehnologiilor noi de învățare în studierea programării și, valorificându-le pe deplin, reușesc să-și îmbunătățească semnificativ performanțele educaționale.

Organizarea procesului de studiu al Programării Orientate pe Obiecte prin asistarea de instrumentele unui Sistem de Management al Instruirii permite dezvoltarea unui mediu educațional constructivist, interactiv și integrat, centrat pe instruit, care oferă oportunități inovative, determinate de un șir de avantaje: accesul la o gamă largă de abordări formative și strategii de învățare flexibile (formarea participativă, metacogniția, învățarea axată pe probleme etc.), evaluarea diversificată (autoevaluare, evaluarea formativă, cea sumativă etc.), abordarea asincronă în timp și spațiu, conținuturi dinamice, mediu colaborativ, feedback.

Bibliografie

1. Lupu I., Cabac V., Gîncu S. Formarea și dezvoltarea competenței de programare orientată pe obiecte la viitorii profesori de informatică. Ch: UST, 2013. 150 p.

2. Keefe K. et al. Adopting XP Practices for Teaching Object Oriented Programming. In: Proceedings of the 8th Australasian Computing Education Conference, Hobart, Australia. 2006.
3. <http://academyplus.md/>
4. Hadjerrouit S. A constructivist approach to object-oriented design and programming. ITiCSE '99: Proceedings of the 4th Annual Conference on Innovation and Technology in Computer Science Education. Cracow, Poland, 1999. p.171-174. <https://www.deepdyve.com/lp/acm/a-constructivist-approach-to-object-oriented-design-and-programming-yzcmeH0Atz?key=acm>
5. Kölling M. The Problem of teaching object-oriented programming Part II: Environments. Journal of Object-Oriented Programming, 11(9), 1999. pp.6-12.
6. Петрова Ю.А. Дифференцированный подход при обучении объектно-ориентированному программированию в старшей школе: дис. канд. пед. наук. Санкт-Петербург, 2002. 169 с.
7. Angarita A. M., Guerrero C. D., Perera G. Virtual Learning Environment to Support Object Oriented Programming Learning https://www.academia.edu/8872180/Virtual_Learning_Environment_to_Support_Object_Oriented_Programming_Learning
8. Biggs J. Teaching for Quality Learning at University (2nd Edition). Open University Press, Berkshire, 2003. http://www.umweltbildung-noe.at/upload/files/OEKOLOG%202014/2_49657968-Teaching-for-Quality-Learning-at-University.pdf
9. Booch G. et all. Object-Oriented Analysis and Design with Applications, 3rd Edition. Pearson, 2007.