

# IMPLEMENTAREA ACTIVITĂȚILOR DE EXERSARE LA INFORMATICĂ, CLASA A X-A, ÎN MEDIUL DE PROGRAMARE DELPHI<sup>1</sup>

## THE IMPLEMENTATION OF PRACTICAL ACTIVITIES IN DELPHI PROGRAMMING ENVIRONMENT FOR INFORMATICS GRADE X

*Marina Bostan, lector universitar,  
UPS „Ion Creangă” din Chișinău*

*Marina Bostan, univ. lecturer,  
„Ion Creanga” SPU of Chisinau  
ORCID: 0000-0002-1191-9501*

**CZU: 004.432.2:373.5**

**DOI: 10.46728/c.v2.25-03-2022.p226-233**

### **Abstract**

Using the Delphi programming environment in Computer Science lessons helps students to understand more quickly the basic concepts of visual programming, to develop their skills in coding applications in different fields, and not least such an approach motivates students to study Computer Science, because it arouses their interest.

**Key-words:** programming language, programming environment, digital textbook, Pascal, Delphi

Evoluția continuă și durabilă a societății este asigurată de mai mulți factori, în special de informatizarea învățământului, care a devenit o acțiune pedagogică caracterizată prin formarea culturii electronice, edificarea spațiului informațional și integrarea noilor tehnologii informaționale în procesul de învățământ.

În sistemul educațional sunt implementate tot mai multe inovații tehnice și electronice și, în primul rând, sunt utilizate în procesul didactic tehnologii informaționale de ultimă oră care, practic, odată la 3-5 ani, se modifică conceptual și oferă avantaje uimitoare pentru învățământ.

Schimbările s-au produs într-un mod firesc și încă continuă să se producă, fapt care nu poate decât să bucure pe cei care manifestă interes pentru acest domeniu și sunt în asentiment cu modificările ce au loc. De la metoda tradițională de conversație, expunere didactică, demonstrație, lucru cu manualul, exercițiul, scrierea pe tablă, la cea modernă de modelare, algoritmizare, problematizare, instruire programată, studiu de caz, metode de simulare (jocuri), învățare prin descoperire, a fost o cale destul de lungă, însă se merită așteptarea.

În epoca revoluției digitale elevul preferă tehnologiile moderne/inovatoare, deoarece grație acestora orele sunt captivante, interactive, asimilarea cunoștințelor devine mai ușoară, fiind pe placul și pe măsura așteptărilor elevilor.

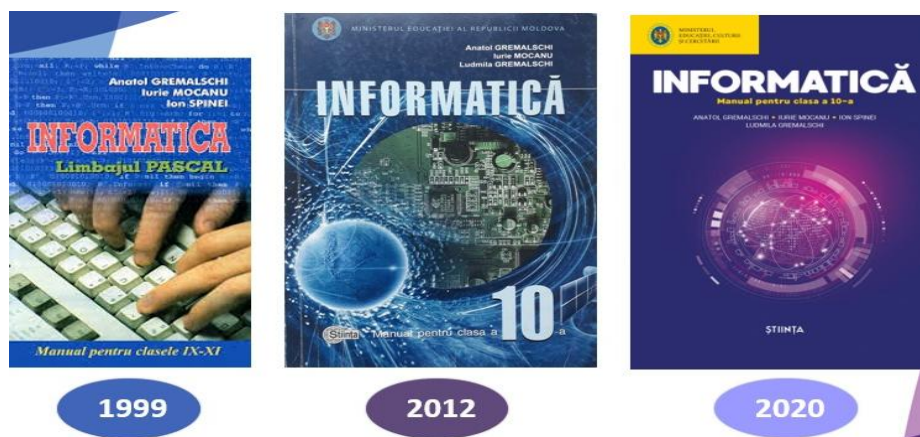
---

<sup>1</sup> Articolul a fost elaborat în cadrul proiectului de cercetare și inovare „Elaborarea și implementarea manualelor digitale interactive în învățământul preuniversitar”, implementat în baza contractului de finanțare încheiat între Agenția Națională pentru Cercetare și Dezvoltare și Universitatea Pedagogică de Stat „Ion Creangă” și aprobat prin Decizia Colegiului Autorității contractante nr. 01-PC din 10.01.2020 din cadrul Priorității strategice IV. Provocări societale, înscris în Registrul de stat al proiectelor din sfera științei și inovării cu cifrul 20.80009.0807.25.

Există o vastă varietate de tehnologii educaționale, atât inovatoare, cât și tradiționale. Cert este faptul că nu putem spune despre niciuna că este perfectă. Utilizarea uneia sau alteia depinde de mai mulți factori: particularitățile de vârstă ale elevului (dar și ale profesorului), contingentul de elevi vs. pregătirea cadrului didactic, subiectul lecției, utilitatea lecției, specificul programului de formare profesională etc. Important însă este ca aplicarea tehnologiei didactice selectate să-i motiveze pe elevi și pe profesori, pentru a atinge nivelul maxim de realizare a obiectivelor propuse.[1, p.34]

Educația generală în ultimii ani a suferit o mulțime de schimbări și adaptări, și în special, modificarea curriculumului disciplinar și reînnoirea (adaptarea) manualelor școlare la curriculum modernizat din anul 2020.

Manualul școlar la Informatică, treapta liceală, în ultimii 20 de ani a evaluat după cum urmează:



**Figura 1. Manualele școlare la Informatică**

În Curriculum național la Informatică pentru clasele X-XII este activ propuse activitățile practice la calculator, la metodologii de predare se menționează principiul de „îmbinare proceselor de predare – învățare a cunoștințelor teoretice cu activitățile practice la calculator;” [2, p.63].

La metodele recomandate se pune accent pe: „...lucrul la calculator, exercițiul, rezolvarea problemelor, lucrarea practică, lucrarea de laborator, studiile de caz, elaborarea proiectelor individuale și/sau în grup. O atenție deosebită se va acorda elaborării de către elevi a proiectelor cu caracter trans- și interdisciplinar, asigurându-se, astfel, implementarea în demersurile didactice a abordărilor STEM și STEAM.” [2, p.64]. Greu de impresionat implementarea astfel de metode fără utilizarea softurilor specializate și medii de dezvoltare.

„Alegerea instrumentarului informatic necesar pentru implementarea Curriculumului este la liberă alegere a instituțiilor de învățământ, a cadrelor didactice și a elevilor. Având drept reper specificul competențelor de format și/sau de performat în domeniul Informaticii, se recomandă utilizarea mijloacelor informatice prezentate în tabelul de mai jos.” [2, p.64]

Clasa	Module	Mijloace informatice recomandate
X	1. Metode de descriere a limbajelor naturale și a limbajelor formale	Aplicații de prezentări electronice.
	2. Vocabularul și sintaxa unui limbaj de programare de nivel înalt 3. Conceptul de dată. Tipuri de date simple 4. Conceptul de acțiune. Instrucțiunile unui limbaj de programare de nivel înalt	Turbo Pascal, Free Pascal; Lazarus, Delphi; Visual Basic, Visual Basic for Applications; Java; C, C++, C#; Python; Online GDB (onlinegdb.com); CSAcademy.com (workspace).

**Figura 2. Mijloace informatice recomandate în Curriculum național**

Toate exemplele din modulele de programare în manualele școlare au fost rezolvate și ulterior prezentate în limbajul Pascal. În manualul de Informatică, anul 2020, pe lângă varianta programului rezolvată în limbajul Pascal s-a adăugat și varianta programului realizată în limbajul C++.

Este cazul de menționat, că pe parcursul dezvoltărilor limbajelor de programare, s-au dezvoltat și au apărut nu doar alte limbaje de programare, dar a apărut și medii de programare în baza limbajelor de programare deja existente. Mediile de programare oferă cu mult mai multe posibilități în domeniul dezvoltării competențelor de programare. Mediile de programare reprezintă pachete de programe care asigură integrarea următoarelor funcții: introducerea și editarea programului sursă, interpretarea sau compilarea, respectiv editarea de legături, încărcarea și lansarea în execuție, depanarea programului. [6, p. 5]

În baza limbajului de programare Pascal s-a dezvoltat mediul de programare Delphi. Delphi a fost creat de compania Borland. A început ca un proiect de cercetare internă în Borland, apoi a evaluat într-un produs comercial. Unul dintre principalele obiective ale acestui nou mediu de dezvoltare a fost interfața ușoară cu principalele motoare de baze de date. A fost unul dintre primele instrumente de dezvoltare recunoscute ca instrumentul RAD, *Rapid Application Development*, când a fost lansat în 1995: era primul mediu de dezvoltare complet vizual, iar compilatorul său era extrem rapid în comparație cu cele existente la acea vreme.

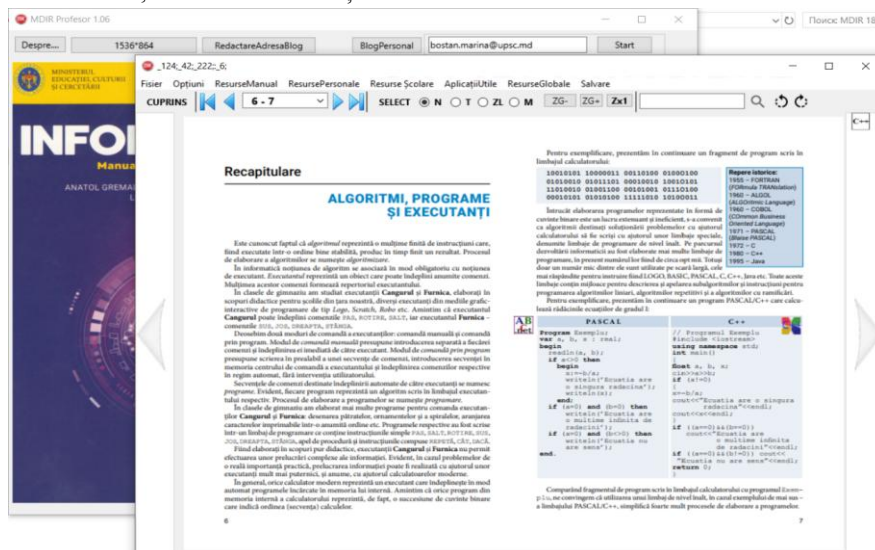
Embarcadero Delphi 10 Seattle face parte din pachetul de dezvoltare RAD STUDIO 10 SEATTLE, este cea mai rapidă metodă de a crea și actualiza aplicații intensive, cu o interfață de utilizare bogată vizuală pentru Windows 10, Mac, dispozitive mobile, IoT și alte platforme. Puteți actualiza ușor aplicațiile VCL și FMX pentru Windows 10 utilizând noile controale și stiluri VCL pentru Windows și componentele serviciilor de platformă Windows. Un mediu de dezvoltare ușor utilizat, cu o memorie disponibilă de două ori pentru proiectele mari, cu suport extins pentru mai multe monitoare și cu inspector de obiecte îmbunătățit. Folosind Delphi 10, dezvoltatorii pot crea aplicații pentru platforme desktop, mobile, cloud și platforme de baze de date de 5 ori mai rapid, inclusiv versiunile pe 32 de biți și pe 64 de biți ale Windows 10. [7]

Elaborarea și implementarea activităților de exersare ni le-am propus să le elaborăm în baza prototipului de manual digital interactiv, elaborat de dr. conf. Univ. dl. Balmuș Nicolae în mediul de programare Delphi 10 Seattle. Manualul digital cuprinde integral conținutul manualului tipărit, având în plus (sau în locul ilustrațiilor din manualul tipărit) elemente specifice: exerciții interactive de diversă natură, animații, secvențe video, imagini și simulări. Manualul digital reprezintă un model nou de integrare a TIC în procesul educațional. El

trebuie să extindă toate componentele procesul de instruire prin includerea de activități multimedia interactive.

Manualul digital(MD) – reprezintă o mapă cu toate fișierele și resursele necesare pentru rularea corectă a aplicației. Fișierul executabil este *DemoMDIR\_ProfesorV\_2022* care pornește aplicația dată și la prima dată necesită introducerea parolei după codul generat de către aplicația, pe calculatorul personal. La momentul dat, versiunea prototipului manualului digital presupune o variantă pentru profesor și o variantă de manual pentru elev.

Prima fereastră este împărțită abstract în două zone: în partea stânga se afișează prima pagină a documentului de bază \*.pdf (la noi variantă \*.pdf a manualului de Informatică pentru clasa a 10-a, 2020), iar în partea dreapta se afișează logoul care este creat de către utilizator. Pentru aceasta în mapa de bază al modelului dat este fișierul LogoMDI, care reprezintă o imagine cu extensia\*.bmp în care poate fi adăugată informația necesară. După tastarea butonului *Start* se deschide fereastră principală de bază care conține antetul aplicației, meniul principal, zona cu instrumente, zona pentru documentul de bază și câmpuri(stâng și drept) pentru plasarea activităților interactive și arată în felul următor:



**Figura 3. Prima fereastră și fereastră principală al manualului digital**

Asamblarea modelului dat cu resursele specifice pentru a avea ca rezultatul final Manual școlar cu suportul didactic digital, e necesar de parcurs mai multe etape și anume: machetarea documentului de bază \*.pdf; scrierea codurilor(listing-uri, textelor) a tuturor programelor(scrise în Pascal, C++, medii de programare) din manual care trebuie să fie accesate direct din MD în mediul corespunzător; proiectarea și elaborarea activităților interactive(în afară SDD) și introducerea lor în SDD; proiectarea, elaborarea și adăugarea activităților interactive (de tip test) în cadrul SDD.

Exemplele de program din manual sunt elaborate în limbajele de programare Pascal și C++ și pot fi accesate direct din MD prin deschiderea automată a aplicațiilor PascalABC și CodeBlocks. Avem posibilitatea de formatat secvențele de program pentru a fi deschise și medii de programare, cum ar fi Delphi.

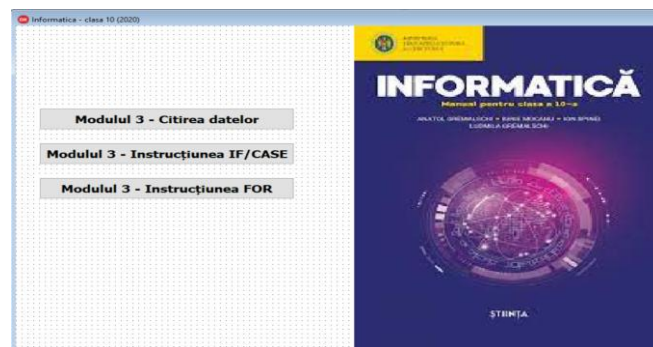
Există de asemenea posibilitatea de a accesa linkuri utile tot direct din MD, de exemplu compilatoarele online pentru diferite limbaje de programare, cum ar fi: <https://www.onlinegdb.com/>, [https://rextester.com/l/pascal\\_online\\_compiler](https://rextester.com/l/pascal_online_compiler), în caz când nu avem la dispoziție instalate limbajele de programare necesare.

Unul din avantajele majore mediului de programare Delphi este crearea unui proiect complex, în care pot fi utilizate nu doar componente-container și multipagini, dar și mai multe forme-ferestre, în afară de forma principală, dintre care se fac legăturile de acces.

Din perspectiva această putem crea o aplicație complexă cu un set de exerciții și probleme rezolvate, care pot fi organizate după diferite criterii, de exemplu după cum urmează modulele unui manual sau paragrafele a unui modul, etc.

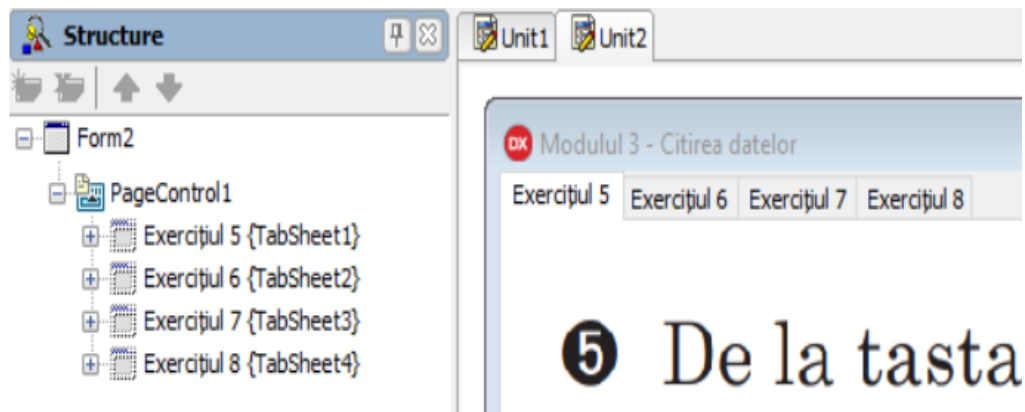
Vom prezenta o varianta de aplicație, care reprezintă câteva paragrafe de probleme și exerciții rezolvate din modulul 3 din manualul de Informatică pentru clasa a 10-a, ediția 2020.

Pe forma principală am plasat butoanele cu denumirea paragrafelor din studiu, la apăsarea cărora se face legătura - se deschid alte forme-secundare și o imagine, care reprezintă manualul la Informatică pentru clasa a 10-a:



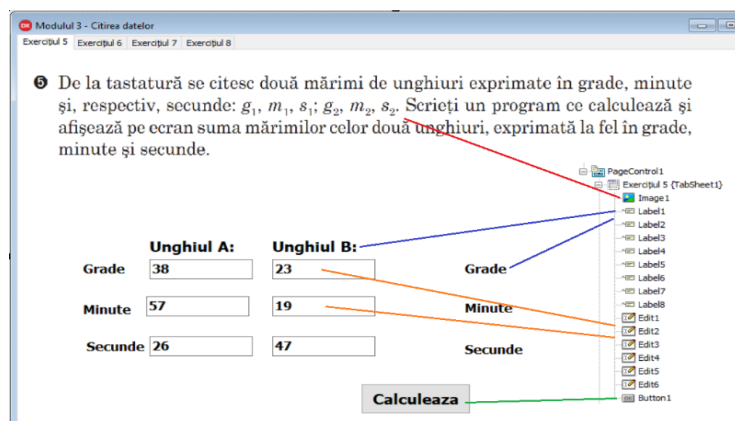
**Figura 4. Fereastra principală a aplicației**

Pentru fiecare paragraf creăm câte o formă-secundară. Pentru paragraful *Citirea datelor* am creat Form2, pe care am plasat componenta PageControl, care reprezintă un set de pagini, care la rândul său sunt obiectele TabSheet. În cazul nostru numărul de pagini reprezintă numărul de exerciții:



**Figura 5. Vizualizarea componentei PageControl cu 5 pagini**

Pentru exercițiul 5 din pag. 128 a manualului vom dezvolta următoare interfață:



**Figura 6. Interfața Exercițiului 5, pag. 128**

Cum se observă din Figura 6, sunt utilizate doar 4 tipuri de obiecte: Image, Label, Edit și Button. În obiectul Image1 am afișat exercițiul din manual, cu ajutorul componentelor Label am etichetat casetele – Edit, pentru introducerea datelor respective și afișarea rezultatelor așteptate și componenta Button1 prelucrează datele introduse în casetele Edit și afișează în etichetele respective rezultatele obținute. Pentru aspectul mai plăcut modificăm unele proprietăți ale componentelor, cum ar fi: Caption, Text, Height, Width, Align, Font, Color, Stretch, ș.a. Menționăm că majoritatea proprietăților se regăsesc la mai multe componente, respectiv cunoașterea acestora nu este deloc dificilă.

Programul însuși se scrie într-o procedură, generată automat de către mediul, atunci când alegem din lista evenimentelor pentru componenta Button1 – evenimentul *Button1Click*. Nu uităm să declarăm variabilele necesare. Pentru citirea datelor folosim funcția de conversie datelor în valori *StrToInt*, introduse în casete *Edit*, proprietatea *Text*, respectiv pentru afișarea rezultatelor folosim funcția de conversie valorilor în text *IntToStr*, care atribuim componentelor *Label*, proprietatea *Caption*:

```

procedure TForm2.Button1Click(Sender: TObject);
var g1,g2,g3,m1,m2,m3,s1,s2,s3:integer;
begin
    g1:=StrToInt(Edit1.Text);   g2:=StrToInt(Edit4.Text);
    m1:=StrToInt(Edit2.Text);   m2:=StrToInt(Edit5.Text);
    s1:=StrToInt(Edit3.Text);   s2:=StrToInt(Edit6.Text);

    g3:=0; m3:=0;
    s3:=s1+s2;
    if s3>60 then begin m3:=1; s3:=s3-60; end;
    m3:=m3+m1+m2;
    if m3>60 then begin g3:=1; m3:=m3-60; end;
    g3:=g3+g1+g2;
    Label6.Caption:='Grade'+IntToStr(g3);
    Label7.Caption:='Minute'+IntToStr(m3);
    Label8.Caption:='Secunde'+IntToStr(s3);
end;

```

**Figura 7. Codul programului pentru Exercițiul 5, pag. 128**

Pentru exercițiul 6 din pag. 128, interfața aplicației arată în felul următor:

Modulul 3 - Citirea datelor  
Exercițiul 5 Exercițiul 6 Exercițiul 7 Exercițiul 8

6 De la tastatură se citește coordonatele a două puncte:  $x_1, y_1; x_2, y_2$ . Scrieți un program care calculează și afișează pe ecran lungimea segmentului cu extremitățile în aceste două puncte și coordonatele mijlocului acestuia.

Coordonata X: Coordonata Y: **Prezentarea grafică:**

Punctul A: -5 -3

Punctul B: 7 6

Formule

**Calculeaza**

Lungimea segmentului: 15

Coordonatele mijlocului segmentului: 1:1,5

**Figura 8. Interfața Exercițiului 6, pag. 128**

În această aplicație am utilizat reprezentarea grafică a exercițiului, care este descrisă într-o procedură separată și prin apăsarea butonului *Prezentarea grafică* doar se apelează. Prin această idee arătăm avantajul programării structurate și dependența interdisciplinară în dezvoltarea aplicațiilor.

Pentru prelucrarea a unui număr de valori introduse de la calculator sau fie generate de calculator, ar fi bine și aceste să fie vizibile pe ecranul calculatorului. În acest scop poate fi utilizată componenta StringGrid, care reprezintă un tabel cu rânduri și coloane, care pot fi setate în proprietăți sau în program. De exemplu pentru exercițiul 8 din pag. 144, interfața aplicației poate fi prezentată în felul următor:

Modulul 3 - Instrucțiunea FOR  
Exercițiul 8 Exercițiul 9 Exercițiul 10

6 Scrieți un program care citește de la tastatură  $n$  numere întregi și afișează pe ecran:

a) câte dintre numerele citite de la tastatură sunt pare;  
b) suma numerelor pozitive citite de la tastatură;  
c) suma tuturor numerelor citite de la tastatură;  
d) media aritmetică a numerelor citite de la tastatură.

n= 5

**Introduceți numerele:**

5 3 2 4 1

**Calculeaza**

a)Nr pare=2  
b)Suma nr poz=15  
c)Suma nr=15  
d)Media aritmetica=3

**Figura 9. Interfața Exercițiului 8, pag. 144**

În final aplicații dezvoltate în cadrul lecțiilor de către elevi cu suportul profesorului, trezesc un mare interes în crearea acestora. Mediul de programare Delphi permite să dezvolte totodată și cunoștințele legate de programare, deprinderi de lucru cu aplicații, dezvoltarea unui design prietenos al aplicațiilor, organizarea și structurarea fișierelor utilizate.

Prin dezvoltarea o astfel de aplicații ajutam elevilor, nu doar să acumulăm cunoștințele ale bazelor programării, dar și formăm la ei în partea majoră competența specifică 4: *Elaborarea produselor digitale grafice, audio și video, demonstrând creativitate și respect față de valorile culturale naționale și universale*. Utilizarea mediului de programare Delphi la orele de Informatică ajută elevilor să înțeleagă mai rapid conceptele de bază de programare vizuală, să-și dezvolte abilitățile în elaborarea aplicațiilor din diferite domenii, și nu în ultimul timp o astfel de abordare motivează elevii să studieze Informatica, deoarece aceasta le trezește interesul de învățare.

## BIBLIOGRAFIE

1. BUDURIN-FURCULIȚĂ, C., ȚIBULEAC, M., ROTARU, I. Motivația valorificată dintr-o perspectivă triadică: profesor-elev-proces educațional. În: *Didactica Pro...*, 2021, nr.1 (125). ISSN 1810-6455, [https://ibn.idsi.md/sites/default/files/imag\\_file/30-34\\_36.pdf](https://ibn.idsi.md/sites/default/files/imag_file/30-34_36.pdf), (vizitat 20.02.2022).
2. GREMALSCHI, A., CHEPTĂNARU, D., CHISTRUGA, Gh., et alii. *Curriculum național: Clasele 10-12. Curriculum disciplinar: Ghid de implementare*. Ministerul Educației, Culturii și Cercetării al Republicii Moldova. Chișinău: „Tipografia centrală”, 2020. 120p. ISBN 978-9975-3436-2-6.
3. GREMALSCHI, A., MOCANU, Iu., SPINEI, I., GREMALSCHI, L. *Informatică: Manual pentru clasa a 10-a*. Chișinău: Știința, 2020. 212 p. ISBN 978-9975-85-234-0.
4. GREMALSCHI, A., MOCANU, Iu., GREMALSCHI, L. *Informatică. Manual pentru clasa a 10-a*, Chișinău: Știința, 2012. 188 p. ISBN 978-9975-67-818-6.
5. GREMALSCHI, A., MOCANU, Iu., SPINEI, I. *Informatică. Manual pentru clasele IX-XI*, Chișinău: Știința, 1999. 256 p. ISBN 9975-67-103-9.
6. MELNIC, A. *Medii de programare: Suport de curs*. Universitatea „George Bacovia” Bacău, 2005, 104p. <https://pdfcoffee.com/qdownload/medii-de-programare-pdf-free.html> (vizitat 15.03.2022).
7. *Delphi software-ul de dezvoltare de instrumente*, <https://materiale.pvgazeta.info/utilizator-143/delphi-software-ul-de-dezvoltare-de-instrumente-de.html>, (vizitat 20.02.2022).

## TEHNOLOGII DE CREARE A APLICAȚIILOR HIBRID: DELPHI – JAVASCRIPT

### HYBRID APPLICATION DEVELOPMENT WITH DELPHI & JAVASCRIPT

*Ilie Mușinschi, programator superior,  
UPS „Ion Creangă” din Chișinău  
Nicolae Balmuș, doctor, conf. univ.  
UPS „Ion Creangă” din Chișinău*

*Ilie Mușinschi, senior programmer  
„Ion Creanga” SPU of Chisinau  
ORCID iD: 0000-0001-7329-5995  
Nicolae Balmuș, PhD, assoc. prof.,  
„Ion Creanga” SPU of Chisinau  
ORCID: 0000-0002-0491-2918*

**CZU: 004.43**

**DOI: 10.46728/c.v2.25-03-2022.p233-238**

#### Abstract

The paper contains brief insights into the history of the development of Delphi and JavaScript programming languages and some ways to create hybrid source code applications. The application design is created in Delphi and the JavaScript code is implemented through the TWebBrowser component. This combines the Delphi concept of fast application creation with the multitude of open-source JavaScript libraries.

**Key-words:** Delphi-JavaScript hybrid applications, digital educational resources, computer-aided simulation.