



Ministerul Educației al Republicii Moldova
Centrul de Excelență în Informatică și Tehnologii Informaționale



"Aprob"

Directorul Centrului de Excelență în
Informatică și Tehnologii Informaționale

Vitalie Zavadschi

20 decembrie 2016

Curriculumul modular
S.06.O.021 Asistență pentru programarea vizuală

Specialitatea: 61310 Programare și analiza produselor program
Calificarea: Asistent programator

Chișinău 2016

Curriculumul a fost elaborat în cadrul Proiectului *EuropeAid/133700/C/SER/MD/12*
"Asistență tehnică pentru domeniul învățământ și formare profesională
în Republica Moldova",
implementat cu suportul financiar al Uniunii Europene



Autori:

Gîncu Silviu, doctor în pedagogie, grad didactic superior.

Pîrvan Evgheni, grad didactic superior, Colegiul „Iulia Hașdeu” din Cahul.

Damaschin Ion, grad didactic doi, Centrul de Excelență în Informatică și Tehnologii Informaționale.

Aprobat de:

Consiliul metodic-științific al Centrului de Excelență în Informatică și Tehnologii Informaționale.



Director

Vitalie Zavadschi

20 decembrie 2016

Recenzenți:

1. „EBSIntegrator” SRL, adresa: str.IonInculeț 33, mun.Chișinău, director Aremesu Vitalie.
2. ÎCS „Cedacri International” SRL, adresa: str.Ștefan cel Mare 171/1, mun.Chișinău, manager departament: Francesco Pipio.

Adresa Curriculumului în Internet:

Portalul național al învățământului profesional tehnic

<http://www.ipt.md/ro/produse-educationale>

Cuprins

I. Preliminarii	4
II. Motivația, utilitatea modulului pentru dezvoltarea profesională.....	5
III. Competențele profesionale specifice modulului	5
IV. Administrarea modulului	5
V. Unitățile de învățare	6
VI. Repartizarea orientativă a orelor pe unități de învățare	18
VII. Studiu individual ghidat de profesor	18
VIII. Lucrările practice recomandate	19
IX. Sugestii metodologice	20
X. Sugestii de evaluare a competențelor profesionale	22
XI. Resursele necesare pentru desfășurarea procesului de studii	24
XII. Resursele didactice recomandate elevilor	25

I. Preliminarii

Programarea vizuală se referă la mediile de programare ce oferă utilizatorului posibilitatea de a specifica programului într-o manieră cel puțin bidimensională. Un limbaj vizual manipulează informația vizuală, suportă interacțiune vizuală sau permite programarea prin folosirea expresiilor vizuale. Programarea vizuală este în mod uzual definită ca fiind utilizarea expresiilor vizuale (cum ar fi grafică, desene, animații sau iconițe), ca mijloace prin care programele sunt create și modificate. Aceste expresii vizuale pot fi folosite în mediile de programare ca și interfețele grafice pentru limbajele de programare textuale; acestea pot fi folosite pentru a alcătui sintaxa unor noi limbaje vizuale de programare ducând la apariția unor noi paradigme sau pot fi folosite în prezentări grafice ale comportării sau structurii unui program textual.

Statutul Curriculumului. Curriculumul modular “Asistență pentru programarea vizuală” este un document normativ și obligatoriu pentru realizarea procesului de pregătire a tehnicienilor în învățământul profesional tehnic postsecundar, care vor acorda asistență în elaborarea și dezvoltarea diverselor produse-program în conformitate cu sarcinile de lucru.

Funcțiile Curriculumului. Funcțiile de bază ale Curriculumul sunt:

- act normativ al procesului de predare, învățare, evaluare și certificare în contextul unei pedagogii axate pe competențe;
- reper pentru proiectarea didactică și desfășurarea procesului educațional din perspectiva unei pedagogii axate pe competențe;
- componentă de bază pentru elaborarea strategiei de evaluare și certificare;
- orientare a procesului educațional spre formare de competențe la elevi;
- componentă fundamentală pentru elaborarea manualelor tipărite, manualelor electronice, ghidurilor metodologice, instrumentarului de evaluare.

Beneficiarii Curriculumului. Curriculumul este destinat:

- profesorilor din instituțiile de învățământ profesional tehnic postsecundar;
- autorilor de manuale și ghiduri metodologice;
- elevilor care își fac studiile la specialitatea în cauză;
- membrilor comisiilor pentru examenele de calificare;
- membrilor comisiilor de identificare, evaluare și recunoaștere a rezultatelor învățării, dobândite în contexte non-formale și informale.

Scopul studierii acestui modul constă în formarea și dezvoltarea competenței profesionale specifice de acordare de asistență în elaborarea și dezvoltarea aplicațiilor, bazate pe paradigma programării orientate pe obiecte și în mentenanța și actualizarea produselor-program (softurilor) aplicative și de sistem. De asemenea, modulul contribuie la dezvoltarea competenței profesionale generale de respectare și de promovare a normelor de drept informatic.

Unitățile de curs ce în mod obligatoriu trebuie certificate până la demararea procesului de instruire la modulul în cauză:

- Programarea structurată.
- Programarea procedurală.
- Programarea calculatorului.
- Administrarea sistemelor de operare.

- Asistență pentru baze de date.
- Asistență pentru programarea orientată pe obiecte.
- Limbajul SQL.

II. Motivația, utilitatea modului pentru dezvoltarea profesională

Studierea acestui modul va contribui la formarea și dezvoltarea de competențe profesionale ce corespund nivelului patru de calificare:

- cunoștințe factice, principii, procese și concepte generale din domeniul elaborării produselor-program;
- abilități cognitive și practice necesare pentru elaborarea aplicațiilor de consolă conform tematicilor incluse;
- asumarea responsabilității pentru mentenanța aplicațiilor.

Competențele formate și dezvoltate în cadrul acestui modul vor fi necesare pentru studierea unităților de curs orientate spre elaborarea și dezvoltarea produselor program. De asemenea, ele vor fi de un real folos în activitatea profesională a tehnicianului, în special, în ocupațiile legate de gestiunea produselor-program utilizate în companii.

III. Competențele profesionale specifice modului

În cadrul modului vor fi formate și dezvoltate următoarele competențe profesionale specifice:

- CS1. Acordarea de asistență în gestionarea componentelor.
- CS2. Stabilirea structurii generale a unei aplicații vizuale.
- CS3. Configurarea elementelor de conținut ale unei aplicații vizuale.
- CS4. Dezvoltarea aplicației vizuale conform specificațiilor tehnice.
- CS5. Mentenanța aplicațiilor vizuale.
- CS6. Utilizarea programării vizuale în activitatea profesională.

IV. Administrarea modului

Semestrul	Numărul de ore				Modalitatea de evaluare	Numărul de credite
	Total	Contact direct		Lucrul individual		
		Prelegeri	Practică/ Seminar			
VI	90	30	30	30	examen	3

V. Unitățile de învățare

Unități de competență	Unități de conținut	Abilități
1. Medii de programare vizuală		
UC1. Gestionarea proiectelor din cadrul mediilor de programare vizuală	<p>1. Medii de programare vizuală. Caracteristici de bază.</p> <p>2. Suprafața de lucru:</p> <ul style="list-style-type: none"> - bara de titlu; - bara meniului principal; - bara de instrumente; - paleta de componente; - inspectorul de obiecte; - fereastra formei; - fereastra editorului de cod. <p>3. Proiecte. Noțiuni generale. Operații de:</p> <ul style="list-style-type: none"> - creare; - salvare; - deschidere; - modificare; - compilare; - executare. <p>4. Fișiere caracteristice proiectului:</p> <ul style="list-style-type: none"> - fișierul principal; 	<p>A1. Lansarea mediului de programare vizuală.</p> <p>A2. Gestionarea suprafeței de lucru.</p> <p>A3. Crearea, editarea și salvarea proiectelor.</p> <p>A4. Compilarea și executarea proiectelor.</p> <p>A5. Crearea aplicațiilor de consolă.</p> <p>A6. Crearea aplicațiilor vizuale.</p>

Unități de competență	Unități de conținut	Abilități
	<ul style="list-style-type: none"> - fișiere atașate unei forme; - fișiere de resurse; - fișiere de rezervă; - fișiere de configurație ș.a. <p>5. Noțiuni de aplicație. Tipuri de aplicații:</p> <ul style="list-style-type: none"> - de consolă; - vizuale. 	
2. Gestionarea componentelor		
UC2. Acordarea de asistență în gestionarea componentelor de tip fereastră	<p>6. Noțiunea de componentă. Elementele caracteristice ale componentelor:</p> <ul style="list-style-type: none"> - proprietăți; - metode; - evenimente. <p>7. Proceduri de gestiune a componentelor:</p> <ul style="list-style-type: none"> - modificarea proprietăților; - apelarea metodelor; - prelucrarea evenimentelor - aranjarea componentelor pe suprafața unei ferestre. <p>8. Componente de tip fereastră. Structura componentelor de tip fereastră:</p> <ul style="list-style-type: none"> - bara de titlu; 	<p>A7. Plasarea componentelor pe suprafața ferestrelor.</p> <p>A8. Identificarea proprietăților componentelor.</p> <p>A9. Identificarea evenimentelor componentelor.</p> <p>A10. Identificarea secvențelor de cod destinate prelucrării evenimentelor componentelor.</p> <p>A11. Excluderea componentelor de pe suprafața ferestrelor.</p> <p>A12. Modificarea proprietăților componentelor.</p> <p>A13. Utilizarea handler-ului de evenimente în scopul prelucrării evenimentelor.</p> <p>A14. Gestionarea sub îndrumare a componentelor.</p> <p>A15. Identificarea elementelor componentelor de tip fereastră.</p>

Unități de competență	Unități de conținut	Abilități
	<ul style="list-style-type: none"> - zona client. <p>9. Elementele caracteristice componentelor de tip fereastră:</p> <ul style="list-style-type: none"> - proprietăți; - metode; - evenimente. 	<p>A16. Modificarea proprietăților componentelor de tip fereastră la nivel de cod sursă.</p> <p>A17. Modificarea proprietăților componentelor de tip fereastră cu ajutorul inspectorului de obiecte.</p> <p>A18. Identificarea în cadrul aplicațiilor a evenimentelor ce trebuie prelucrate.</p> <p>A19. Prelucrarea evenimentelor componente de tip fereastră conform specificațiilor propuse.</p> <p>A20. Apelarea metodelor componentelor de tip fereastră.</p> <p>A21. Implementarea sub îndrumare a componentelor de tip fereastră.</p>
3. Componente frecvent utilizate		
UC3. Acordarea de asistență în utilizarea componentelor-tip	<p>10. Componente frecvent utilizate:</p> <ul style="list-style-type: none"> - butoane de comandă; - casete de editare; - etichete; - cutii de editare multilinie; - meniuri; - grupuri de butoane; - cutii de grupare. 	<p>A22. Identificarea proprietăților, a metodelor și evenimentelor componentelor-tip.</p> <p>A23. Modificarea proprietăților componentelor-tip.</p> <p>A24. Apelarea metodelor componentelor-tip.</p> <p>A25. Prelucrarea evenimentelor componentelor-tip.</p> <p>A26. Implementarea procedurilor de colaborare a componentelor-tip din cadrul proiectelor.</p> <p>A27. Implementarea sub îndrumare a componentelor-tip.</p> <p>A28. Utilizarea componentelor-tip în cadrul aplicațiilor vizuale.</p>

Unități de competență	Unități de conținut	Abilități
	11. Elementele caracteristice componentelor frecvent utilizate: <ul style="list-style-type: none"> - proprietăți; - metode; - evenimente. 	A29. Modificarea stării și comportamentul componentelor-tip conform specificațiilor propuse.
UC4. Acordarea de asistență în implementarea obiectelor de tip șir de caractere	12. Tipul de date șir de caractere în mediul de programare vizuală. Structura obiectelor de tip șir de caractere: <ul style="list-style-type: none"> - metode; - operatori. 13. Proceduri frecvent utilizate: <ul style="list-style-type: none"> - inițializarea șirurilor; - conversii către alte tipuri de date; - conversia datelor la nivel de șir de caractere. 	A30. Declararea obiectelor de tip șir de caractere. A31. Inițializarea obiectelor de tip șir de caractere. A32. Conversia șirurilor de caractere. A33. Apelarea metodelor obiectelor de tip șir de caractere. A34. Comunicarea cu și dintre obiecte de tip șir de caractere. A35. Implementarea sub îndrumare a obiectelor de tip șir de caractere.
4. Gestiunea ferestrelor		
UC5. Acordarea de asistență în implementarea ferestrelor	14. Aplicații multiforme. 15. Proceduri de gestiune a ferestrelor. 16. Tipuri de ferestre: <ul style="list-style-type: none"> - pentru citirea datelor; - pentru afișarea datelor; - modale. 	A36. Adăugarea, excluderea și modificarea ferestrelor. A37. Implementarea ferestrelor destinate citirii și afișării mesajelor. A38. Implementarea formelor modale. A39. Crearea sub îndrumare a aplicațiilor MDI.

Unități de competență	Unități de conținut	Abilități
	17. Aplicații MDI. 18. Proceduri de utilizare a variabilelor în cadrul mai multor ferestre. 19. Ferestre de dialog. Proprietăți, metode și evenimente ale ferestrelor de dialog. Ferestre: <ul style="list-style-type: none"> - pentru deschiderea/salvarea fișierelor textual/de tip imagine; - pentru prelucrarea fontului; - pentru prelucrarea culorilor; - pentru căutarea/înlocuirea textelor; - pentru imprimarea conținutului fișierelor textuale. 	A40. Implementarea ferestrelor de dialog destinate selectării fișierelor. A41. Implementarea ferestrelor de dialog destinate modificării fonturilor. A42. Implementarea ferestrelor de dialog destinate modificării culorilor. A43. Implementarea ferestrelor de dialog destinate imprimării fișierelor textuale. A44. Elaborarea sub îndrumare și gestiunea ferestrelor.
5. Elemente de grafică		
UC6. Acordarea de asistență în implementarea ferestrelor obiectelor grafice	20. Suprafețe pentru desenare. 21. Modalități de gestiune a imaginilor. 22. Proprietăți ale obiectul Canvas: <ul style="list-style-type: none"> - pensulă; - pânză; - font; - creion. 23. Metode ale obiectului Canvas:	A45. Identificarea componentelor cu suprafețe de desenare. A46. Încărcarea imaginilor în componente. A47. Modificarea proprietăților componentelor în scopul afișării corecte a imaginilor. A48. Salvarea imaginilor în formatele grafice prestabilite. A49. Modificarea proprietăților obiectului Canvas. A50. Apelarea metodelor obiectului Canvas. A51. Desenarea figurilor geometrice.

Unități de competență	Unități de conținut	Abilități
	<ul style="list-style-type: none"> - elipsă; - linie; - dreptunghi; - poligon; - afișarea textului etc. <p>24. Regimul grafic în mediu de programare vizuală.</p> <p>25. Grafice de funcții. Modalități de desenare:</p> <ul style="list-style-type: none"> - prin puncte; - prin linii. <p>26. Construirea graficelor funcțiilor:</p> <ul style="list-style-type: none"> - trigonometrice (sin, cos etc.); - liniare; - exponențiale; - logaritmice; - definite de utilizator. <p>27. Efecte de animație. Operații de:</p> <ul style="list-style-type: none"> - deplasare pe suprafața formei a unei imagini; - rotire a unei imagini; - reamplasare a unei imagini. <p>28. Integrarea componentei de tip timp cu elemente de grafică.</p>	<p>A52. Desenarea în mediul de programare vizuală a axelor de coordonate.</p> <p>A53. Desenarea de figuri prin apelarea metodelor de desenare a punctului și liniei.</p> <p>A54. Desenarea graficelor.</p> <p>A55. Modificarea stării componentei de tip timp conform cerințelor propuse.</p> <p>A56. Prelucrarea evenimentelor componentelor de tip timp conform cerințelor propuse.</p> <p>A57. Dezvoltarea sub îndrumare a obiectelor grafice.</p>

Unități de competență	Unități de conținut	Abilități
6. Componente specializate		
UC7. Acordarea de asistență în aplicarea componentelor specializate	<p>29. Componente specializate frecvent utilizate:</p> <ul style="list-style-type: none"> - liste de opțiuni și casete combinate; - componente pentru prelucrarea informațiilor de tip data calendaristică; - componente pentru prelucrarea informațiilor de tip tabel; - componente pentru gestiunea directoarelor și fișierelor. <p>30. Elementele caracteristice componentelor specializate frecvent utilizate:</p> <ul style="list-style-type: none"> - proprietăți; - metode; - evenimente. 	<p>A58. Setarea proprietăților componentelor specializate</p> <p>A59. Implementarea metodelor componentelor specializate.</p> <p>A60. Implementarea secvențelor de cod destinate prelucrării evenimentelor componentelor specializate.</p> <p>A61. Implementarea metodelor de colaborare a componentelor specializate.</p> <p>A62. Implementarea componentelor de tip liste de opțiuni.</p> <p>A63. Implementarea componentelor de tip casete combinate.</p> <p>A64. Implementarea componentelor destinate prelucrării datelor calendaristice.</p> <p>A65. Apelarea metodelor caracteristice tipului de date fișier.</p> <p>A66. Afișarea proprietăților de fișiere.</p> <p>A67. Afișarea adreselor de fișiere.</p> <p>A68. Crearea, ștergerea, redenumirea, căutarea de fișiere și directoare.</p> <p>A69. Implementarea sub îndrumare a prelucrărilor de fișiere.</p> <p>A70. Implementarea componentelor destinate prelucrării directoare lor și fișierelor.</p> <p>A71. Apelarea metodelor caracteristice tipului data calendaristică.</p>

Unități de competență	Unități de conținut	Abilități
		<p>A72. Determinarea proprietăților obiectelor de tip data calendaristică.</p> <p>A73. Descompunerea datelor calendaristice în valori numerice.</p> <p>A74. Compunerea valorilor numerice în date calendaristice.</p> <p>A75. Implementarea sub îndrumare a prelucrărilor de date calendaristice.</p>
7. Aplicații complexe		
UC8. Acordarea de asistență în implementarea aplicațiilor complexe	<p>31. Aplicații complexe.</p> <p>32. Proceduri de elaborare a aplicațiilor complexe.</p> <p>33. Simularea aplicației calculator cu numere complexe. Implementarea de funcționalități precum:</p> <ul style="list-style-type: none"> – calcule (+ - * /); – diverse. <p>34. Simularea aplicației gestiune liste. Implementarea funcționalităților:</p> <ul style="list-style-type: none"> – crearea/distrugea listei; – încărcare/salvarea datelor din fișier; – includerea/excluderea unei noi înregistrări; – procesarea (căutare, înlocuire, calculare etc.); 	<p>A76. Identificarea componentelor necesare în funcție de specificul proiectelor.</p> <p>A77. Amplasarea componentelor pe suprafața ferestrelor.</p> <p>A78. Implementarea secvențelor de cod destinate prelucrării evenimentelor.</p> <p>A79. Modificarea proprietăților componentelor din cadrul proiectelor.</p> <p>A80. Apelarea metodelor specifice componentelor din cadrul proiectelor.</p> <p>A81. Implementarea de interfețe grafice conform specificațiilor propuse.</p> <p>A82. Organizarea conlucrării componentelor din cadrul proiectelor.</p>

Unități de competență	Unități de conținut	Abilități
	<ul style="list-style-type: none"> – ordonare a informației; – diverse. 	<p>A83. Adăugarea sub îndrumare de noi funcționalități în proiectele supuse modernizării.</p> <p>A84. Implementarea sub îndrumare a instrumentarului de validare a datelor.</p> <p>A85. Implementarea conceptelor programării orientată pe obiecte: abstractizare, încapsulare, moștenire, polimorfism și agregare sub îndrumare în cadrul aplicațiilor vizuale.</p> <p>A86. Dezvoltarea sub îndrumare a aplicațiilor complexe.</p>
8. Conexiuni cu bazele de date		
UC9. Crearea conexiunilor cu bazele de date	<p>35. Tehnologia ADO.</p> <p>36. Conectarea la bazele de date.</p> <p>37. Etapele de conectare la bazele de date. ODBC pentru realizarea conexiunilor cu bazele de date</p>	<p>A87. Instalarea driverelor pentru conexiunea cu bazele de date.</p> <p>A88. Includerea componentelor de gestiune a bazelor de date în proiecte.</p> <p>A89. Stabilirea conexiunilor cu bazele de date.</p> <p>A90. Verificarea conexiunilor cu bazele de date.</p> <p>A91. Securizarea accesului la bazele de date.</p>
UC10. Implementarea componentelor de tip tabel	<p>38. Componente de tip tabel.</p> <p>39. Proprietăți de:</p> <ul style="list-style-type: none"> – stabilire a conexiunii cu baza de date; – deschidere/închidere a tabelului; 	<p>A92. Stabilirea conexiunii cu tabelele bazelor de date.</p> <p>A93. Implementarea operațiilor de deschidere a tabelelor bazelor de date.</p>

Unități de competență	Unități de conținut	Abilități
	<ul style="list-style-type: none"> – accesare a informației din tabel. <p>40. Metode de:</p> <ul style="list-style-type: none"> – editare a datelor; – parcurgere a datelor; – excludere a datelor; – salvare a datelor. 	<p>A94. Implementarea operațiilor de editare a informațiilor din tabelele bazelor de date.</p> <p>A95. Implementarea operațiilor de parcurgere a înregistrărilor din tabelele bazelor de date.</p> <p>A96. Implementarea operațiilor de excludere a înregistrărilor din tabelele bazelor de date.</p> <p>A97. Implementarea operațiilor de salvare a datelor din tabelele bazelor de date.</p>
UC11. Implementarea componentelor de tip interogare	<p>41. Componente de tip interogare.</p> <p>42. Interogări de:</p> <ul style="list-style-type: none"> – selecție; – inserare; – modificare; – excludere. <p>43. Proceduri de prelucrare a înregistrărilor în bază de tabel.</p> <p>44. Proceduri de prelucrare a înregistrărilor în bază de două și mai multe tabele.</p> <p>45. Proceduri de prelucrare a înregistrărilor cu parametru.</p>	<p>A98. Implementarea interogărilor de selecție.</p> <p>A99. Implementarea interogărilor de inserare.</p> <p>A100. Implementarea interogărilor de modificare.</p> <p>A101. Implementarea interogărilor de excludere.</p> <p>A102. Implementarea interogărilor în bază de un singur tabel.</p> <p>A103. Implementarea interogărilor în bază de două și mai multe tabele.</p> <p>A104. Implementarea interogărilor cu parametru.</p>

Unități de competență	Unități de conținut	Abilități
9. Gestionarea înregistrărilor		
UC12. Implementarea componentelor destinate gestiunii înregistrărilor bazelor de date	<p>46. Componente pentru gestiunea înregistrărilor bazelor de date:</p> <ul style="list-style-type: none"> – navigare; – cutii de editare; – etichete; – liste de opțiuni; – casete combinate; – diagrame; – căutare; – elaborate de utilizator; <p>47. Proceduri de elaborare a formularelor de:</p> <ul style="list-style-type: none"> – adăugare a înregistrărilor; – vizualizare a înregistrărilor; – căutare a informației în baza de date; – excludere a înregistrărilor; – validare a datelor. 	<p>A105. Implementarea operațiilor de inserare a informației în bazele de date.</p> <p>A106. Implementarea operațiilor de extragere a informațiilor din bazele de date.</p> <p>A107. Setarea proprietăților componentelor destinate gestiunii înregistrărilor bazelor de date.</p> <p>A108. Implementarea metodelor destinate gestiunii înregistrărilor bazelor de date.</p> <p>A109. Implementarea secvențelor de cod destinate prelucrării evenimentelor aferente gestiunii înregistrărilor bazelor de date.</p> <p>A110. Implementarea metodelor de colaborare a componentelor destinate gestiunii înregistrărilor bazelor de date.</p> <p>A111. Utilizarea componentelor de navigare în cadrul unui proiect.</p> <p>A112. Implementarea sub îndrumare a componentelor de tip etichete.</p> <p>A113. Implementarea sub îndrumare a componentelor de tip cutii de editare.</p>

Unități de competență	Unități de conținut	Abilități
		<p>A114. Implementarea sub îndrumare a componentelor de tip liste de opțiuni.</p> <p>A115. Implementarea sub îndrumare a componentelor de tip casete combinate.</p> <p>A116. Implementarea sub îndrumare a componentelor de tip diagramă.</p> <p>A117. Implementarea sub îndrumare a componentelor de tip căutare.</p>
UC13. Asistență în elaborarea rapoartelor	<p>48. Componente de tip raport.</p> <p>49. Structura obiectelor de tip raport:</p> <ul style="list-style-type: none"> – proprietăți; – metode; – evenimente. 	<p>A118. Creare rapoartelor în bază de tabele.</p> <p>A119. Crearea rapoartelor în bază de interogări.</p> <p>A120. Crearea rapoartelor în baza componentelor-tip ale mediului de programare vizuală.</p>

VI. Repartizarea orientativă a orelor pe unități de învățare

Nr. crt.	Unități de învățare	Numărul de ore			
		Total	Contact direct		Lucrul individual
			Prelegeri	Practică/ Seminar	
1.	Medii de programare vizuală	6	2	2	
2.	Gestionarea componentelor	6	2	2	2
3.	Componente frecvent utilizate	12	4	4	6
4.	Gestiunea ferestrelor	6	2	2	2
5.	Elemente de grafică	12	4	4	4
6.	Componente specializate	18	6	6	6
7.	Aplicații complexe	6	2	2	2
8.	Conexiuni cu bazele de date	12	4	4	4
9.	Gestionarea înregistrărilor	12	4	4	4
	Total	90	30	30	30

VII. Studiu individual ghidat de profesor

Materii pentru studiul individual	Produse de elaborat	Modalități de evaluare	Termeni de realizare
1. Gestionarea componentelor			
Componente de tip fereastră	Portofoliu: Set de aplicații vizuale cu utilizarea componentelor de tip fereastră	Prezentarea portofoliului	Săptămâna 2
2. Componente frecvent utilizate			
Componente frecvent utilizate	Portofoliu: Set de aplicații vizuale cu utilizarea componentelor frecvent utilizate	Prezentarea portofoliului	Săptămâna 4
3. Gestiunea ferestrelor			
Ferestre	Portofoliu: Set de aplicații vizuale cu utilizarea componentelor de tip ferestre	Prezentarea portofoliului	Săptămâna 5
4. Elemente de grafică			
Grafica în mediu de programare vizual	Portofoliu:	Prezentarea portofoliului	Săptămâna 7

Materii pentru studiul individual	Produse de elaborat	Modalități de evaluare	Termeni de realizare
	Set de aplicații vizuale cu utilizarea elementelor de grafică		
5. Componente specializate			
Componente specializate	Portofoliu: Set de aplicații vizuale cu utilizarea componentelor specializate	Prezentarea portofoliului	Săptămâna 9
6. Aplicații complexe			
Aplicația calculator Aplicații de gestiune a listelor	Aplicația de simulate a unui calculator cu numere complexe Aplicația de gestiune a înregistrărilor unei liste (evidența personalului unei instituții, evidența cărților unei săli de lectură etc.)	Prezentarea aplicațiilor	Săptămâna 11
7. Conexiuni cu bazele de date			
Aplicație de gestiune a înregistrărilor unei baze de date	Fragmente de program cu utilizare tabelelor și interogărilor a unei baze de date din cel puțin 3-4 tabele	Prezentare	Săptămâna 13
8. Gestionarea înregistrărilor			
Aplicație de gestiune a înregistrărilor unei baze de date	Dezvoltarea aplicației de gestiune a datelor unei baze de date cu elaborarea formularelor de adăugare, căutare, prelucrare a datelor, inclusiv rapoarte cu utilizarea diagramelor	Prezentarea proiectului	Săptămâna 15

VIII. Lucrările practice recomandate

Lucrările practice vor fi efectuate în formă de lucrări de laborator. Tematica lucrărilor recomandate:

1. Prelucrarea componentelor de tip fereastră.
2. Prelucrarea componentelor de tip butoane de comandă, casete de editare, etichete, cutii de editare multilinie, meniu, grupuri de butoane, cutii de grupare.
3. Prelucrarea componentelor de fereastră de intrare/ieșire.
4. Prelucrarea componentelor de grafică.
5. Prelucrarea componentelor la nivel de opțiuni și casete combinate, data calendaristică, tabel, directoare și fișiere.
6. Prelucrarea obiectelor create de către utilizator la nivel de componente.

7. Prelucrarea datelor bazelor de date la nivel de tabele/interogări.
8. Elaborarea formularelor bazelor de date.
9. Elaborarea rapoartelor bazelor de date.
10. Gestionarea bazelor de date formate din 2-3 tabele.

IX. Sugestii metodologice

Elementele de bază ale Curriculumului sunt competențele ce trebuie formate și dezvoltate în procesul de formare profesională. Acestea vor fi formate prin organizarea eficientă a procesului de instruire. Pentru aceasta sunt necesare două condiții:

1. Organizarea activităților. Pentru buna organizare a procesului didactic ambii participanți necesită de a-și organiza activitățile. De modul cum sunt organizate acestea depinde în mare măsură nivelul de formare a competențelor. În această ordine de idei, în procesul de organizare a activităților se vor asigura:

- condiții optime pentru buna colaborare dintre elev și profesor;
- un set de procese care duc la îmbunătățirea relațiilor dintre părți;
- un nivel de implicare a părților acționând în baza unor reguli și acțiuni prestabilite.

2. Selectarea adecvată a metodelor de instruire. Se recomandă utilizarea metodelor de instruire precum:

Simularea și modelarea. Simularea este utilizată pentru prezentarea la faza inițială a unor concepte, oferind posibilitatea de ghidare a activității elevului în bază de situații practice. Prin intermediul acestei metode se pot reda, prin analogie, diverse situații, raționamente, care pot să reprezinte relații dintre obiecte, fenomene, procese etc. Aceste metode sunt recomandate pentru studierea materiilor Caracteristici de bază ale mediului de programare vizuală, componente frecvent utilizate ș. a.

Problematizarea mai poate fi denumită și predare prin rezolvare de probleme sau predare productivă de probleme. Conform acestei metode instruitului este pus în fața unor dificultăți create în mod deliberat, și prin depășirea lor învață ceva nou. „Punctul forte” al metodei îl constituie situația-problemă. Din această cauză este necesar de a formula corect situația. La crearea situație de tip problemă se va ține cont de următoarele caracteristici:

- A. Situația trebuie să prezinte o dificultate pentru instruit, iar pentru a găsi soluția, acesta se va confrunta cu efort de gândire;
- B. Situația trebuie să prezinte interes, astfel încât acesta să acționeze spre a rezolva problema;
- C. Situația trebuie să orienteze activitatea instruitului spre a rezolva problema și de a cointeresa pe acesta de a dobândi noi cunoștințe;
- D. Rezolvarea situației nu va fi posibilă fără a apela la resurselor recent dobândite.

Prin intermediul situației create, instruitul este cointerestat de a studia, analiza și a participa la rezolvarea problemei. Aplicarea acestei metode presupune parcurgerea a patru etape:

1. Formularea problemei – este descrisă situația problemă, explicarea, după necesitate a diferitor puncte cheie, care ar permite instruitului să perceapă problema;
2. Studiarea problemei – se lucrează în mod independent, sunt reactualizate anumite resurse;
3. Determinarea soluției – în cadrul acestei etape sunt pregătite resursele necesare, se descoperă mijloacele care duc la rezolvarea problemei și este analizat modul de aplicare a acestora în determinarea soluției;
4. Obținerea rezultatului final – se analizează rezultatul obținut și formate anumite concluzii.

Această metodă este recomandată pentru studiarea materiilor componente frecvent utilizate, elemente de grafică ș. a.

Algoritmizarea reprezintă o metodă de predare-învățare bazată pe utilizarea și valorificarea algoritmilor în procesul de instruire. Algoritmul de instruire se reprezintă sub forma unui grup de scheme, unui set de operații, iar prin parcurgerea lor într-o ordine bine stabilită duce la rezolvarea unui set de probleme caracteristice unei familii de situații. În rezultatul aplicării acestei metode se va oferi posibilitatea elevului de a elabora treptat propriile scheme, aplicabile în diferite circumstanțe didactice. Această metodă este recomandată pentru studiarea materiilor gestionarea componentelor, aplicații complexe ș. a.

Instruirea asistată de calculator este o metodă didactică care valorifică principiile de modelare și analiză cibernetică. Prin intermediul calculatorului se pune la dispoziția elevului un set de probleme, care necesită a fi analizate, completate sau elaborate. Utilizarea metodei va oferi posibilitatea de organizarea informației conform cerințelor programei adaptabile la capacitățile fiecărui elev; stimularea cognitivă a elevului prin secvențe didactice și întrebări ce vizează depistarea unor lacune, probleme, situații-problemă; rezolvarea sarcinilor didactice prezentate anterior prin reactivarea sau obținerea informațiilor necesare de la resursele informatice apelate prin intermediul calculatorului; realizarea unor sinteze recapitulative după parcurgerea unor teme, module de studiu, lecții; asigurarea unor exerciții suplimentare de stimulare a creativității elevului. Această metodă este recomandată pentru studiarea materiilor în cadrul orelor teoretice la tematici precum: componente frecvent utilizate, gestiunea ferestrelor, componente specializate ș. a.

Metoda studiul de caz valorifică o situație reală care se analizează și se rezolvă. Așa cum problemele rezolvate în stilul orientat pe obiecte au un grad sporit de dificultate, sunt cazuri când este necesar de a prezenta elevului probleme deja rezolvate. Avantajul metodei, constă în faptul că fiecare dintre elev își va aduce aportul la analiza și rezolvarea problemei. În utilizarea acestei metode se conturează câteva etape: 1) Selectarea și prezentarea cazului; 2) Organizarea echipelor de lucru; 3) Prelucrarea și conceptualizarea; 4) Structurarea finală a studiului. Această metodă este recomandată pentru studiarea materiilor aplicații complexe, gestionarea înregistrărilor ș. a.

Instruirea prin proiecte reprezintă o modalitate de instruire/autoinstruire grație căreia elevii, dar mai ales elevii efectuează o cercetare orientată spre obiective practice și finalizată într-un produs ce poate fi un obiect, un aparat, o instalație, o culegere tematică, un album, o lucrare științifică etc. Această metodă este recomandată pentru studierea materiilor elemente de grafică, aplicații complexe, gestionarea înregistrărilor, precum și pentru materiile propuse în cadrul orelor de tip studiu individual.

X. Sugestii de evaluare a competențelor profesionale

Evaluarea competențelor profesionale este procesul prin care sunt colectate și analizate dovezile necesare pentru judecarea competenței în raport cu cerințele calificării profesionale. Calificarea profesională este documentul în care se descriu rezultatele învățării în concordanță cu cerințele pieței muncii, specificate în standardul ocupațional/ profilul ocupațional. Evaluarea competențelor profesionale este un proces complet diferit de sistemul tradițional de evaluare a cunoștințelor. Evaluarea competențelor profesionale este un proces care presupune consultarea și colaborarea dintre elev și profesor. Evaluarea competențelor are loc prin furnizarea de către elev a dovezilor de competență care sunt interpretate de către profesor. Dovezile de competență acumulate sunt rezultate considerate parțiale și atât elevul cât și profesorul pot solicita clarificări suplimentare.

Procedura de evaluare a competențelor profesionale pentru modulul *Asistență pentru programarea vizuală*, va oferi elevilor posibilitatea de a-și demonstra atât cunoștințele teoretice și practice. Metodele folosite în procesul de evaluare vor evidenția cunoștințele și deprinderile necesare pentru efectuarea activităților de muncă și, mai ales, capacitatea elevului de a obține rezultatele practice așteptate.

Activitățile de evaluare vor fi orientate spre motivarea elevilor și obținerea unui feedback continuu, fapt ce va permite corectarea operativă a procesului de învățare, stimularea autoevaluării și a evaluării reciproce, evidențierea succeselor, implementarea evaluării selective sau individuale. Pentru a eficientiza procesele de evaluare, înainte de a demara evaluările, cadrul didactic va aduce la cunoștința elevilor tematica lucrărilor, modul de evaluare (bareme/grile/criterii de notare) și condițiile de realizare a fiecărei evaluări.

Evaluarea curentă/formativă se va realiza prin diverse modalități: observarea comportamentului elevului, analiza rezultatelor activității elevului, discuția/conversația, prezentarea proiectelor individuale de activitate. Prin evaluarea curentă/formativă, cadrele didactice informează elevul despre nivelul de performanță; îl motivează să se implice în dobândirea competențelor profesionale.

Evaluarea sumativă se realizează la finele modulului în baza simulării în atelier a unei situații de problemă din contexte profesionale variate, care solicită elevului demonstrarea competenței profesionale. Cadrele didactice vor elabora sarcini prin care vor orienta comportamentul profesional al elevului spre demonstrarea sistemului de cunoștințe și abilități. În acest scop, vor fi clar stabiliți indicatorii și descriptorii de performanță ai procesului și produsului realizat de către elev.

Portofoliul reprezintă o metodă complexă de evaluare în care un rezultat al evaluării este elaborat pe baza aplicării unui ansamblu variat de probe și instrumente de evaluare. Portofoliul, de regulă este realizat pe o perioadă mai îndelungată (în decursul mai multor ore). Conținutul unui portofoliu este reprezentat de rezultatele la: lucrări practice, studiul individual, investigații, referate și proiecte, observarea sistematică la clasă, autoevaluarea elevului, chestionare de atitudini etc. Alegerea elementelor ce formează portofoliul este realizată de către profesor (astfel încât acestea să ofere informații concludente privind pregătirea, evoluția, atitudinea elevului) sau chiar de către elev (pe considerente de performanță, preferințe etc.). Structurarea evaluării sub forma de portofoliu se dovedește deosebit de utilă, atât pentru profesor, cât și pentru elev sau părinții acestuia. Pentru a realiza o evaluare pe bază de portofoliu, profesorul:

- va comunica elevilor intenția de a realiza un portofoliu, adaptând instrumentele de evaluare ce constituie “centrul de greutate” ale portofoliului la specificul unității de învățare;
- va alege componentele ce formează portofoliul, dând și elevului posibilitatea de a adăuga piese pe care le consideră relevante pentru activitatea sa;
- va evalua separat fiecare piesă a portofoliului în momentul realizării ei, dar va asigura și un sistem de criterii pe baza cărora să realizeze evaluarea globală și finală a portofoliului;
- va pune în evidență evoluția elevului, particularitățile de exprimare și de raportare a acestuia la aria vizată;
- va integra rezultatul evaluării portofoliului în sistemul general de notare.

Competențele elevului se manifestă prin produse concrete, care sunt analizate de către profesor în raport cu aspectele critice stabilite pentru unitate/unitățile de competență pentru care este evaluat. Dovezile de competență sunt informațiile produse de un elev din care rezultă că îndeplinește toate aspectele descrise de unitatea/unitățile de competență pentru care este evaluat, respectiv are cunoștințele și deprinderile necesare.

Evaluarea nivelului de dezvoltare a competențelor în cadrul orelor:

teoretice se va realiza prin teste, exemple de aplicare a cunoștințelor teoretice în practică, machete etc.;

de laborator se va realiza prin elaborarea de către elev, în termeni concreți, a aplicațiilor web avînd la bază unitățile de conținut studiate în cadrul orelor teoretice precum și abilitățile anterior dezvoltate;

de studiu individual se va realiza prin studierea de către elev a materialelor suplimentare decît cele oferite în cadrul orelor de tip contact direct și prezentarea de portofolii pentru anumite unități de conținut prin care elevul își va demonstra abilitățile formate.

Probe de evaluare a competențelor, în baza situațiilor de problemă de la viitoarele locuri de muncă:

- elaborarea aplicațiilor cu utilizarea componentelor de tip butoane de comandă, casete de editare, etichete, cutii de editare multilinie, meniu, grupuri de butoane, cutii de grupare;

- elaborarea aplicațiilor cu utilizarea componentelor de tip fereastră de intrare/ieșire;
- elaborarea aplicațiilor cu utilizarea componentelor de grafică;
- elaborarea aplicațiilor cu utilizarea componentelor la nivel de opțiuni și casete combinate, data calendaristică, tabel, directoare și fișiere;
- elaborarea aplicațiilor cu utilizarea obiectelor create de către utilizator la nivel de componente;
- elaborarea aplicațiilor cu utilizarea componentelor de tip tabel/interogare;
- elaborarea aplicațiilor cu utilizarea formularelor ale unei baze de date;
- testarea aplicațiilor de consolă elaborate.

În calitate de **produse pentru măsurarea competențelor** se vor folosi:

- componente prelucrate conform specificațiilor propuse;
- aplicații vizuale elaborate conform specificațiilor propuse;

Criteriile de evaluare a produselor pentru măsurarea competenței vor include:

- Utilizarea mediului de programare vizuală în conformitate cu sarcinile propuse.
- Corectitudinea aplicațiilor elaborate.
- Fundamentarea deciziilor.
- Ținuta lingvistică.
- Respectarea termenilor de elaborare.

XI. Resursele necesare pentru desfășurarea procesului de studii

Cerințe față de sălile de curs	
Pentru orele teoretice	Cabinet de informatică cu 15 calculatoare Proiector
Pentru orele de laborator	Laborator de informatică care asigură fiecărui elev un calculator
Cerințe tehnice	
Parametri tehnici minimi ale calculatorului	Procesor: 2 GHz Memorie operativă: 4 GB Unitate de stocare: 500 GB Afișaj și grafică: size: 22'', resolution: 1366x768 Network: Ethernet, 100 Mb
Software	Sistem de Operare Microsoft Windows C++ Builder Visual Studio

XII. Resursele didactice recomandate elevilor

Nr. crt.	Denumirea resursei	Locul în care poate fi consultată/ accesată/ resursa
1.	A. Braicov, S. Gîncu, Borland C++ Builder. Chişinău, 2009. http://en.calameo.com/read/002801569838bdc5a9164	Internet
2.	I. Lupu, V. Cabac, S. Gîncu Formarea şi dezvoltarea competenţei de programare orientată pe obiecte la viitorii profesori de informatică, Chişinău, 2013. http://en.calameo.com/books/0028015690bbcf6a9e03	Internet
3.	S. Gîncu Metodologia rezolvării problemelor de informatică în stilul orientat pe obiecte, Chişinău, 2012. http://en.calameo.com/read/002801569ce96f41ef59f	Internet
4.	Programarea orientată pe obiect şi programarea vizuală. http://colegiulbratianu.ro/wp-content/themes/theme53309/documente/software/DotNet.pdf	Internet
5.	Архангельский А. Я. Программирование в C++Builder 6. Москва: Изд. БИНОМ 2003. http://a1308.ru/books/id324	Internet
6.	Культин Н. Б. C++ Builder в задачах и примерах. СПб.: БХВ-Петербург, 2005. ftp://ftp.kng.ignix.ru/BOOKS/LANG/C%23/C++_Builder_v_zadachakh_i_primerakh.pdf	Internet