

Ministerul Educației al Republicii Moldova  
Universitatea de Stat „Alec Russo” din Bălți  
Facultatea de Științe Reale, Economice și ale Mediului  
Catedra de matematică și informatică

**Curriculum pentru unitatea de curs**

# **Managementul proiectelor**

**Program de profesionalizare: *Programarea WEB***

**Bălți, 2015**

Curriculumul a fost discutat la ședința Catedrei de  
matematică și informatică

*Procesul verbal nr. 9 din 7 mai 2015*

Șeful catedrei dr. conf. univ. \_\_\_\_\_

Eugeniu Plohotniuc

Curriculumul a fost aprobat la ședința Consiliului  
Facultății de Științe Reale, Economice și ale Mediului

Procesul verbal nr. 15 din \_\_\_\_ iunie 2015

Decanul facultății, dr. hab., prof. univ. \_\_\_\_\_

Pavel Topala

## Informații de identificare a disciplinei

**Facultatea:** Științe Reale, Economice și ale Mediului

**Catedra** de matematică și informatică

**Domeniului general de studii:** 444 Științe exacte

**Tipul programului:** Program de profesionalizare

**Denumirea programului (specializarea):** Programarea WEB

**Denumire disciplinei:** *Managementul proiectelor.*

### Administrarea unității de curs:

Codul unității de curs	Credite ECTS	Total ore	Repartizarea orelor				Forma de evaluare	Limba de predare
			Prel.	Sem.	Lab.	L. ind		
F.01.O.15	5	150	24	16		110	Ex. scris	română

**Anul de studii și semestrul în care se studiază disciplina:** anul II, semestrul III

**Regimul disciplinei:** obligatorie

**Categoria formativă:** unitate de curs fundamentală.

## Informații referitoare la cadrele didactice



Titularul cursului - *Mircea Petic*, dr. în informatică, conferențiar universitar. Absolvent al Universității de Stat „A. Russo” din Bălți, specialitatea „Informatica și limba engleză aplicată”. A efectuat stagii în diverse universități, inclusiv peste hotare, unde s-a specializat în domeniul informaticii, utilizarea tehnologiei

informației și a comunicațiilor în învățământ, procesarea limbajului natural, programarea obiect orientată, programarea logică și funcțională, ingineria programării, managementul proiectelor.

E-mail: [petic.mircea@yahoo.com](mailto:petic.mircea@yahoo.com)

Orele de consultații - miercuri: 15.00 -17.00. Consultațiile se oferă atât în regim „față-în-față”, cât și prin utilizarea poștei electronice, YahooMessenger (petic.mircea) și Skype sau rețeaua socială facebook. Numele în Skype – [mir\\_cescu](#).

## Descrierea disciplinei

Disciplina "*Managementul proiectelor*" este un curs fundamental care permite aprofundarea cunoștințelor în programare. Realizarea unui sistem informatic reprezintă o activitate complexă și de durată, ce antrenează mari resurse materiale, umane și de timp. IT-ul este doar o industrie printre alte industrii care întreprind proiecte. Asemenea altor domenii, proiectele IT au specificul său metodologic. Este nevoie de proiect management de calitate pentru a livra valoare în mod predictibil, pentru a oferi informații reale atât echipei, cât și managementului companiei și pentru a putea lua decizii bazate pe rezultate și metrice concrete. Pînă la urmă, "*Managementul proiectelor*" este o disciplină care își propune să crească gradul de siguranță al produselor software în condițiile unui mediu în continuă schimbare.

În cadrul disciplinei se studiază noțiuni de: proiect, ingineria programării, ingineria cerințelor, limbajele de modelare, testarea software, drepturile de autor. Viitorii programatori vor obține cunoștințe referitor la diverse modele de dezvoltare software, vor căpăta deprinderi de lucru cu limbajul de modelare UML.

## Integrarea în programul de formare (planul de învățămînt)

Studierea unității de curs „*Managementul proiectelor*” se sprijină pe cunoștințele, capacitățile și competențele dezvoltate în cadrul disciplinei „*Programarea Java*”, „*Programarea obiect orientată*”, „*Arhitectura calculatoarelor și sisteme de operare*”, „*Baze de date*”, „*Programarea Web*”, „*Rețele de calculatoare*”, studiată la ciclul I. Finalitățile și conținutul unității de curs sînt corelate cu finalitățile și conținuturile unităților de curs menționate mai sus.

## **Competențele dezvoltate în cadrul disciplinei**

### *Competențe generice*

1. Competențe cognitive: căutarea, procesarea și analiză critică a informației din diverse surse și de prezentare a informației în diferite forme.
2. Competențe de cooperare și lucru în echipă: utilizarea cunoștințelor și deprinderilor specifice pentru a crea oportunități de realizare a sarcinilor de învățare în colaborare cu colegii.
3. Competențe în domeniul TIC: integrarea TIC în procesul de căutare, prelucrare și asimilare a noilor cunoștințe în cadrul disciplinei.

### *Competențe specifice*

1. Competențe cognitive: de cunoaștere și înțelegere a principalelor noțiuni de ingineria programării și modelarea vizuală în proiectarea sistemelor software.
2. Competențe de aplicare: de proiectare a resurselor și evaluarea prețului de proiect IT.
3. Competențe de analiză: analiză a calității produselor software.

## Finalitățile disciplinei

La finalizarea studierii disciplinei studentul va fi capabil

1. Să determine resurselor necesare realizării proiectului unui sistem software;
2. Să analizeze atributele măsurabile ale calității produsului software;
3. Să explice modul de distribuire a forței de muncă pentru elaborarea unui proiect software.
4. Să proiecteze și să aplice setul de teste care ar verifica produsul software elaborat.

## Prealabilele/precondițiile

- a) *Algoritmi și programare*: Tipuri de date simple și structurate. Instrucțiunile simple și compuse (condiționale și repetitive).
- b) *Programarea obiect orientată*: Clase. Moștenire. Polimorfism. Obiecte.
- c) *Arhitectura calculatoarelor și sisteme de operare*: componentele calculatoarelor, sistemele de operare Windows, Linux.
- d) *Baze de date*: SGBD, proiectarea bazelor de date.
- e) *Programarea Web*: client, server, site.

## Calendarul unității de curs

*Prelegeri (Lecții teoretice)*

Prelegerea	Conținuturi	Nr. de ore
1	Noțiuni de proiect. Exemple de proiecte IT.	2
2	Motivația studierii ingineriei programării. Erori celebre în programare.	2
3	Etapele de dezvoltare a software. Modele de dezvoltare software.	2
4	Ingineria cerințelor	2

5	Modelarea arhitecturii programului.	2
6	Limbajele de modelare.	2
7	Asigurarea calității software.	2
8	Testarea software.	2
9	Atributele măsurabile în asigurarea calității programelor.	2
10	Distribuirea forței de muncă în timp.	2
11	Drepturile de autor.	2
12	<b>Lucrare de control.</b>	1
	<b>Prezentarea sarcinii individuale nr. 1</b>	1
<b>Total</b>		<b>24</b>

*Seminare (Lección practice)*

<b>Seminarul</b>	<b>Tematica lecțiilor de seminar</b>	<b>Nr. de ore</b>
1	Interfața aplicației Microsoft Project. Crearea listei de sarcini.	2
2	Planificarea resurselor. Evaluarea prețului de proiect.	2
3	Modelarea diagramelor UseCase. Diagramele de clasă UML.	2
4	Diagramele de secvență UML.	2
5	Diagramele comportamentale UML.	2
6	Diagrame de activități UML.	2
7	Diagrame de deployment UML.	2
8	<b>Prezentarea sarcinii individuale nr. 2</b>	2
<b>Total</b>		<b>16</b>



### **Sarcină nr.1 pentru lucrul independent**

Elaborarea unui micro-referat în baza unei surse alternative de informare asupra unor aspecte noi cu referire la tematica "*Managementului proiectelor*". Referatul vor fi prezentate într-un raport editat într-un document Word pe 3-5 pagini format A4, font #12, 1.5 intervale. Activitatea sa va fi evaluată atât de către colegi cât și de către titularul disciplinei într-o ședință aparte.

Criterii de evaluare:

1. Exactitatea expunerii materialului;
2. Corectitudinea (logică, ortografică) a raportului prezentat.

**Termenul limită (deadline) de prezentare a sarcinii – perechea a 10-a**

**(prelegeri)**

### **Sarcină nr.2 pentru lucrul independent**

Elaborarea diagramelor UML pentru un proiect de sistem informațional stabilit. Rezultatele vor fi prezentate într-un raport editat într-un document Word pe 2-3 pagini format A4, font #12, 1.5 intervale. Activitatea sa va fi evaluată atât de către colegi cât și de către titularul disciplinei într-o ședință aparte.

Criterii de evaluare:

1. Exactitatea proiectării diagramelor UML;
2. Argumentarea abordării propuse spre implementare;
3. Corectitudinea (logică, ortografică) a raportului prezentat..

Studentul va prezenta raportul însoțit de fișierele sursă care demonstrează editarea diagramelor UML.

**Termenul limită (deadline) de prezentare a sarcinii – perechea a 10-a (seminar)**



## ***Resursele informaționale la disciplină***

### **A. Literatura de bază**

1. Todoran E., Inginerie SoftWare, Editura MediaMira, Cluj-Napoca, 2001, ISBN: 973-9358-73-X, 240 p.
2. Гласс Р., Факты и заблуждения профессионального программирования, Символ+, Москва, 2008, 232 с.
3. Мараско Д., IT проекты. Фронтные очерки, Символ+, Москва, 2008, 376 с.

### **B. Literatura suplimentară**

4. Sommerville I. Software Engineering, Addison Wesley, Boston, 2001, 773 p.
5. Larman C. Applying UML and Patterns, Addison Wesley, Boston, 2002, 615 p.
6. Gamma E., Helm R., Johnson R., Vissides J., Design Patterns, Elements of Reusable Object-Oriented Software, Addison Wesley, Boston, 1998, 431 p.

### **C. Resurse Internet**

7. Situl oficial Argo UML [online] data 05.05.2015 Disponibil pe internet  
< <http://argouml.tigris.org/> >

## Evaluare

Cunoștințele, capacitățile și competențele studenților vor fi evaluate:

1. La prelegeri (**PR**):
  - 1.1. *Lucrare de control scrisă*: perechea a 12-a (**LC**).
  - 1.2. *Sarcina nr. 1 pentru lucru independent*: perechea a 12-a (**SLI1**)
2. În cadrul seminarilor (lecțiilor practice) (**SM**):
  - 2.1. *conform calendarului disciplinei*, de tot 6 note (**S1, S2, S3, S4, S5, S6**);
  - 2.2. *Sarcina nr. 1 pentru lucru independent*: perechea a 8-a (**SLI2**)
3. La examenul final, conform orarului întocmit de decanat (**Ex**).

Nota finală la disciplina „Managementul proiectelor” se calculează conform formulelor:

$$N_{\text{evaluarea curentă}} = 0,5 \times \text{PR} + 0,5 \times \text{SM}$$

$$N_{\text{finală}} = 0,6 \times N_{\text{evaluarea curentă}} + 0,4 \times N_{\text{examen}}$$

Unde **PR**=(**LC**+**SLI1**)/2 și **SM**=0.5×(**S1**+**S2**+**S3**+**S4**+**S5**+**S6**)/6+0.5×**SLI2**

Examenul final se susține scris, care va include un test din 30 de itemi, cu 4 variante de răspuns fiecare. **Principiile de lucru în cadrul disciplinei**

1. Calendarul cursului (termenii-limită de prezentare a sarcinilor propuse spre rezolvare, momentele de evaluare etc.) este corelat cu calendarele la alte discipline din semestru. De aceea prezentarea sarcinilor după termenul-limită indicat în calendar nu este salutăată, iar studenții care amână frecvent prezentarea sarcinilor își formează o imagine nefavorabilă.
2. Nu este salutăată întârzierea la ore.
3. Este salutăată poziția activă a studentului, care studiază din propria inițiativă noi conținuturi, propune soluții (aplicații, instrumente Web), formulează întrebări în cadrul prelegerilor și a orelor practice.

4. În cadrul disciplinei o atenție sporită va fi oferită respectării principiilor *etice*. Prezentarea unor soluții a sarcinilor, preluate de la colegi sau din alte surse, preluarea informațiilor din diverse surse, fără a face trimitere la sursă, va fi considerată *plagiat* și va fi sancționată prin note de „1” .



**Exemplu de itemi pentru testul de la examen**  
la disciplina „Managementul proiectelor”

1. Care diagramă UML nu există?
  - a. *diagrama de citire*
  - b. *diagrama de colaborare*
  - c. *diagrama de activități*
  - d. *diagrama de secvență*
2. Limbajul UML este un ...
  - a. limbaj de programare
  - b. limbaj de modelare
  - c. limbaj de marcare
  - d. limbaj de descriere
3. Cea mai costisitoare etapă într-un proiect IT o reprezintă ...
  - a. testarea
  - b. scrierea de cod
  - c. analiza
  - d. proiectare
4. Cele mai multe erori se comit la ...
  - a. scriere de cod
  - b. proiectare
  - c. analiză
  - d. verificare
5. Care etapă ne asigură că programul este stabil și că funcționează corect din punctul de vedere al dezvoltatorilor?
  - a. validare
  - b. verificare
  - c. testare
  - d. întreținere
6. În care model de dezvoltare a programelor are loc cooperarea strânsă între client și programatori?
  - a. Scrum
  - b. Agile
  - c. Cascadă
  - d. XP
7. Ce nu se refera la atributele unui program bun?
  - a. ușor de folosit
  - b. resurse hardware mari
  - c. modificabil
  - d. sigur

- 8 Limbajul artificial care poate fi folosit să exprime informații sau cunoaștere sau sisteme este ...
- limbaj de programare
  - limbaj de modelare
  - limbaj de marcare
  - limbaj de descriere
9. Abordarea sistematică a dezvoltării, funcționării, întreținerii, și retragerii din funcțiune a programelor reprezintă ...
- informatica
  - ingineria programării
  - managementul proiectelor
  - testarea programelor
10. Care este scopul modelării?
- vizualizarea unui sistem
  - cheltuirea de bani
  - verificarea programului
  - nici un răspuns corect

**Baremul**  
**de convertire a punctajului în note**  
**pentru examenul de evaluare a cunoștințelor la disciplina**  
***„Managementul proiectelor”***  
**(în baza REGULAMENTULUI**  
**cu privire la evaluarea rezultatelor academice ale studenților în**  
**Universitatea de Stat „Alec Russo” din Bălți)**

Procentajul	Nota
100 – 91	10
90 – 81	9
80 – 71	8
70 – 66	7
65 – 61	6
60 – 51	5
50 – 41	4
40 – 31	3
30 – 16	2
15 – 0	1

## Note