

FISA DISCIPLINEI

**Ingineria programelor
COBD308**

Număr credite4

1. Obiectivele disciplinei

Abordarea sistematică a analizei, proiectării și realizării programelor și însușirea noțiunilor fundamentale de inginerie software.

2. Rezultatele învățării (se exprima în obiective măsurabile ce fac subiectul evaluării)

a. Cunoștințele generale

- Etapele de realizare ale unui sistem software
- Clasificarea aplicațiilor informatice
- Asigurarea și controlul calității unui produs software
- Evaluarea și testarea

b. Cunoștințele de specialitate

- Metodologii și tehnici de analiză și proiectare a sistemelor informatice
- Utilizarea instrumentelor software de proiectare

c. Competențele generale

- Ergonomie și interfață om-mașină
- Calitate

d. Competențele de specialitate

- Metode, norme, instrumente
- Tehnici de dezvoltare
- Concepție, modelare și arhitectura aplicațiilor
- Integrare de programe și sisteme de operare
- Producție și exploatare

e. Abilitățile cognitive specifice

- Lucru în echipă
- Gestiunea timpului

3. Concordanța cu obiectivele planului de învățământ/specializării

a. Contribuția rezultatelor învățării disciplinei la formarea competențelor specializării

Lucrul cu sisteme informatice (proiectare, realizare, testare, evaluare, întreținere) specific oricărui absolvent al acestei specializări, munca în echipă, nu se poate face fără cunoștințele specifice învățate la această disciplină.

Mai mult, proiectul pe care trebuie să-l realizeze fiecare pentru obținerea licenței este în fond un proiect de sistem software cu toate componentele, care trebuie proiectat utilizând metodologia și tehnicile de inginerie software.

b. Cerințele disciplinare prealabile

Cerințele prealabile pentru ingineria programelor sunt cunoștințele solide de baze de date fără de care nu există nici un sistem informatic real, iar ca discipline adiacente, proiectarea interfețelor, rețele de calculatoare, sisteme multimedia și hipermedia..

4. Structura activității didactice

CURS 28 ore
Seminar 0 ore
Lucrări practice 28 ore
Proiect 0 ore

5. Prezentarea conținutului disciplinei

a. Curs

<i>Continutul activității</i>	<i>Nr. de ore</i>
Sistemele software: Probleme și perspective; Ingineria software- Definiții. Criza software. Teorema pentru ingineria software. Clasificarea programelor.	2
Cerinte- specificatii: Procesul de alegere a cerintelor. Specificarea cerintelor. Specificarea formala.	2
Dezvoltarea produselor software: Paradigmele dezvoltarii software. Metode și calitate. Productivitate. Componentele unui instrument CASE.	4
Proiectarea software-ului: Metode și strategii de proiectare. Proiectarea sistemelor concurente. Proiectarea calitatii. Documentatia. Dreptul de proprietate și garanții.	4
Concepte ale metodologiei de analiza și proiectare orientate pe obiect : Modele de lucru. Modelul obiectelor. Modelul dinamic. Modelul functional.	6
Proiectarea sistemelor software in metodologie OMT: Proiectarea sistemului. Exemple de arhitecturi existente.	2
Fiabilitatea și mentenabilitatea software: Metode de realizare a software-ului fiabil. Asigurarea fiabilitatii aplicatiilor software. Modele de studiere "a posteriori" a fiabilitatii.	2
Proiectarea sistemelor utilizând UML: 8.1. Concepte generale de analiză și proiectare UML. 8.2. Diagrame de clase. 8.3. Diagrame de interacțiune . 8.4. Diagrame de stare . 8.5. Diagrame de activitate. 8.6. Diagrame de desfășurare. 8.6. UML - ul și programarea	6

b. Seminar

Continutul activității *Nr. de ore*

c. Lucrări practice

<i>Continutul activității</i>	<i>Nr. de ore</i>
- Realizarea unui sistem software mic lucrând in echipa (2 membri) și ulilizând UML;	10
- Analiza și proiectarea unei interfete prietenoase cu utilizatorul;	8
- Realizarea unui sistem software utilizand un CASE.	10

d. Proiect

Continutul activității *Nr. de ore*

6. Invatare

a. Forme de invatare/predare

Curs magistral, aplicații practice, exemple

b. Resurse educationale

Prezentări PowerPoint, resurse on-line

c. Bibliografie disponibilă

La biblioteca universitatii, a catedrei, pe internetul catedrei și pe CD

d. Alte resurse

Dotari laboratoare, instrumente software disponibile, resurse umane

7. Evaluare

a. Forme de evaluare

A . La activitățile practice studenții vor primi 4-5 teme care constau din proiectarea unor sisteme informatice cu ajutorul unui instrument software și care vor fi prezentate periodic pe parcursul semestrului sub forma de rezultate practice. Toate temele presupun lucrul în echipă formată din minimum 2 membri. De asemenea sistemele realizate vor fi prezentate oral în fața colegilor și vor fi supuse autoevaluării, evaluării colegiale și evaluării cadrului didactic concretizate prin note. În ultima săptămână din semestru se va stabili pe baza notelor obținute în timpul semestrului și a participării la activitățile de laborator nota finală de la laborator.

B Examinarea finală se va face în timpul sesiunii de examene printr-un test grilă care conține 50 de întrebări cu ponderi egale (un punct) alese aleator dintr-o bază de întrebări.

b. Principii de notare

Nota la grila = nr_puncte*2/10 (răspuns corect la întrebare =1 punct, răspuns greșit =0 puncte)

Nota finală la disciplină = (2*nota_la_grila + nota_laborator)/3

c. Informarea studenților cu privire la evaluarea asociată disciplinei

Studenții sunt informați la primul curs despre cerințele disciplinei, cum se face evaluarea și care este programul de evaluare.

8. Responsabil de curs

Nume : Prof. Dr. Ing. Cornelia Novac Ududec

Date de contact : email : cornelia.novac@ugal.ro



Șef cat. CIA,
