

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Babeș-Bolyai Cluj-Napoca
1.2 Facultatea	Facultatea de Matematică și Informatică
1.3 Departamentul	Departamentul de Informatică
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii	Postuniversitar
1.6 Programul de studiu / Calificarea	Program postuniversitar de formare si dezvoltare profesionala in Informatica

2. Date despre disciplină

2.1 Denumirea disciplinei	Proiect						
2.2 Titularul activităților de curs							
2.3 Titularul activităților de seminar							
2.4 Anul de studiu	2	2.5 Semestrul	1	2.6. Tipul de evaluare	C	2.7 Regimul disciplinei	Obligatoriu
Codul disciplinei		MLR5097					

3. Timpul total estimat (ore pe semestru al activităților didactice)

3.1 Număr de ore pe săptămână	2	Din care: 3.2 curs	-	3.3 seminar/ laborator	2
3.4 Total ore din planul de învățământ	28	Din care: 3.5 curs	-	3.6 seminar/ laborator	28
Distribuția fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie și notițe					10
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					15
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					5
Tutoriat					2
Examinări					3

Alte activități:		-
3.7 Total ore studiu individual	36	
3.8 Total ore pe semestru	64	
3.9 Numărul de credite	4	

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	<input type="checkbox"/> Metode avansate de programare <input type="checkbox"/> Baze de date <input type="checkbox"/> Programare Web
4.2 de competențe	<ul style="list-style-type: none"> • Abilitatea de a programa într-un limbaj de nivel înalt • Concepte de baza despre baze de date • Abilitatea de a analiza si proiecta si implementa aplicatii software

5. Condiții (acolo unde este cazul)

5.1 De desfășurare a cursului	
5.2 De desfășurare a seminarului/laboratorului	<input type="checkbox"/> Laboratoare cu calculatoare <input type="checkbox"/> Medii de programare pentru Java <input type="checkbox"/> Sisteme de gestiune a bazelor de date

6. Competențele specifice acumulate

Co mp ete nțe pro fesi ona le	C2.1 Identificarea de metodologii adecvate de dezvoltare a sistemelor software. C2.2 Identificarea și explicarea mecanismelor adecvate de specificare a sistemelor software.
--	---

Competențe transversale	<p>CT1 Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională.</p> <p>CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar și dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare și colaborare cu grupuri diverse.</p> <p>CT3 Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare în limba română</p>
--------------------------------	---

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	<input type="checkbox"/> Competențe îmbunătățite de proiectare și programare
7.2 Obiectivele specifice	<input type="checkbox"/> Insusirea conceptelor si tehnicilor existente pentru dezvoltarea aplicațiilor web. <input type="checkbox"/> Familiarizarea cu concepte moderne din dezvoltarea proiectelor web.

8. Conținuturi

8.1 Curs	Metode de predare	Observații
8.2 Seminar / laborator	Metode de predare	Observații
1. Discutii generale legate de activitatile de la laborator. Prezentarea modului de lucru. Stabilirea cerintelor tehnice. Prezentarea temelor. Repartizarea proiectelor. Discutii/Explicatii individuale cu fiecare student legat de cerintele proiectului asociat. Stabilirea termenelor de predare.	Explicare, exemplificare, dezbatere, dialog	
2. Definiere plan de proiect - vedere de ansamblu. - Discutii generale. - Discutii/Explicatii individuale la fiecare student. - Prezentare GIT, sistem de gestiune a surselor.	Explicare, exemplificare, dezbatere, dialog	
3. Configurare proiect. - Modele conceptuale. - Modelare domeniu. - Prezentare sabloane enterprise. - Stabilire tehnologii/framework pentru partea de client (React, AngularJs, jQuery, Bootstrap, GWT, etc.) si server.(JSP, Node, Servlet, JSF, EJB)	Explicare, exemplificare, dezbatere, dialog	

<p>4. Definirea arhitecturii proiectului.</p> <ul style="list-style-type: none"> - Prezentare arhitectura stratificata. - Detalii domeniu, servicii, repository. - Prezentare sablon: Model-View-Controller/Model-View-Presenter - Definire infrastructura, model de reprezentare a datelor. 	Explicare, exemplificare, dezbateri, dialog	
<p>5. Detaliere plan proiect.</p> <ul style="list-style-type: none"> - Stabilirea functionalitatilor. - Definire taskuri/subtaskuri pentru fiecare functionalitate. - Estimarea efortului fiecarui task. 	Explicare, exemplificare, dezbateri, dialog.	
<p>6. Modelare</p> <ul style="list-style-type: none"> - Rafinare model obiectual. - Predare plan proiect. cu toate functionalitatile estimate. - Predare model de reprezentare a datelor. 	Explicare, exemplificare, dezbateri, dialog, evaluare.	
<p>7. Code review</p> <ul style="list-style-type: none"> - Definire teste automate backend si UI - junit, Selenium, etc. - Continuous delivery - Jenkins/Hudson, TeamCity, etc. - Prezentare concept code review - Configurare: Gerrit, Upsource, etc. 	Explicare, exemplificare, dezbateri, dialog.	
<p>8. Code review</p> <ul style="list-style-type: none"> - Analizare/discutii cod - partea de persistenta. - Validarea respectarii documentelor si modelelor stabilite precedent. 	Dezbateri, dialog. Evaluare.	
<p>9. Code review</p> <ul style="list-style-type: none"> - Analizare/discutii cod - partea de servicii. - Validarea respectarii documentelor si modelelor stabilite precedent. - Validare separare clara intre straturile arhitecturale. 	Dezbateri, dialog. Evaluare.	
<p>10. Code review</p> <ul style="list-style-type: none"> - Analizare/discutii cod - partea de prezentare - UI - Validarea respectarii documentelor si modelelor stabilite precedent. - Verificare evaluare folosire sabloane. 	Dezbateri, dialog. Evaluare.	
<p>11. Discutii arhitectura</p> <ul style="list-style-type: none"> - Documentarea aplicatiei la nivel inalt, sa reiasa separarea stratificata - Prezentarea diagrame de deploy. - Discutii despre cum sa facem deploy automat. 	Explicare, exemplificare, dezbateri, dialog.	
<p>12. Design UI</p> <ul style="list-style-type: none"> - Prezentare concept EventBus. - Demo componente decuplate, sincronizate doar prin evenimente. - Discutii testare componente cu Selenium. 	Explicare, exemplificare, dezbateri, dialog.	
<p>13-14. Review proiect.</p> <ul style="list-style-type: none"> - Evaluare per ansamblu. - Evaluare functionalitati. - Evaluare progres pentru fiecare etapa folosind istoricul din Git/Mercurial. 	Evaluare.	

Referinte:

1. Gamma, E., Vlissides, J., Johnson, R. and Helm, R., Design Patterns CD: Elements of Reusable Object-Oriented Software.(CD-ROM)(Boston, MA, USA 1998).
2. Fowler, M., 2002. Patterns of enterprise application architecture. Addison-Wesley Longman Publishing Co., Inc..
3. Fields, D.K., Kolb, M.A. and Bayern, S., 2001. Web Development with Java Server Pages. Manning Publications Co..
4. Humble, J. and Farley, D., 2010. Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation (Adobe Reader). Pearson Education.
5. Beck, K., 2003. Test-driven development: by example. Addison-Wesley Professional.
6. Martin, R.C., 2003. Agile software development: principles, patterns, and practices. Prentice Hall PTR.

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

--

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.4 Curs			
10.5 Seminar/laborator	<ul style="list-style-type: none"> • Capacitatea de a proiecta și dezvolta diferite tipuri de aplicații web. 	Documentarea proiectului: impartirea pe task-uri, alcatuirea diagramelor	50%
		Evaluarea implementarii	40%
		Respectarea termenelor de prepare	10%
10.6 Standard minim de performanță			
<input type="checkbox"/> Pentru a promova disciplina studentul trebuie să obțină cel puțin nota cinci.			

Data completării

Semnătura titularului de curs

Semnătura titularului de seminar

11 iunie 2016

Data avizării în departament

Semnătura directorului de departament

.....

Prof. Dr. Andreica Anca