

FIȘA DISCIPLINEI

Concepte avansate de testare

Anul universitar 2026-2027

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Babeș-Bolyai
1.2. Facultatea	Facultatea de Matematica și Informatica
1.3. Departamentul	Departamentul de Informatica
1.4. Domeniul de studii	Informatica
1.5. Ciclul de studii	Master
1.6. Programul de studii / Calificarea	Baze de date
1.7. Forma de învățământ	Cu frecvență

2. Date despre disciplină

2.1. Denumirea disciplinei	Concepte avansate de testare			Codul disciplinei	MME8150
2.2. Titularul activităților de curs	Conf. Dr. Vescan Andreea				
2.3. Titularul activităților de seminar	Conf. Dr. Vescan Andreea				
2.4. Anul de studiu	2	2.5. Semestrul	3	2.6. Tipul de evaluare	Examen
2.7. Regimul disciplinei	Opțional	2.8. Tipul disciplinei		Disciplină de specializare (DS)	

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

3.1. Număr de ore pe săptămână	4	din care: 3.2. curs	2	3.3. seminar/ laborator/ proiect	2
3.4. Total ore din planul de învățământ	56	din care: 3.5. curs	28	3.6 seminar/laborator	28
Distribuția fondului de timp pentru studiul individual (SI) și activități de autoinstruire (AI)					ore
Studiul după manual, suport de curs, bibliografie și notițe (AI)					84
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					26
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					24
Tutoriat (consiliere profesională)					4
Examinări					6
Alte activități					0
3.7. Total ore studiu individual (SI) și activități de autoinstruire (AI)				144	
3.8. Total ore pe semestru				200	
3.9. Numărul de credite				8	

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	
4.2. de competențe	

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Videoproiector, acces la internet
5.2. de desfășurare a seminarului/ laboratorului	Videoproiector, acces la internet

6.1. Competențele dobândite în urma absolvirii programului de studii (se preiau din planul de învățământ)¹

¹ Se vor prelua din Planul de învățământ al programului de studii acele competențe profesionale și/sau transversale la dezvoltarea cărora contribuie disciplina pentru care se elaborează fișa disciplinei. Pentru fiecare competență se va prelua întregul enunț, inclusiv codul competenței, cu formularea care apare în planul de

Competențe profesionale	
Codul competenței	Competență
CP1	înțelegerea și operarea cu conceptele de bază din domeniul ingineriei software
CP3	înșușirea conceptelor matematice și modelelor formale care să faciliteze înțelegerea, verificarea și validarea funcționării sistemelor
CP5	folosirea metodologiilor și instrumentelor specifice limbajelor de programare și ingineriei programării
Competențe transversale	
Codul competenței	Competență
CT1	capacitatea de analiză și sinteză a informației; comportarea onorabilă, etică, respectarea deontologiei profesionale
CT2	abilități de muncă în echipă, cu preluarea diferitelor roluri de execuție și conducere pentru realizarea unor proiecte
CT3	abilități de comunicare profesională: descrierea clară, concisă, verbală și în scris, a rezultatelor profesionale

6.2. Rezultatele învățării specifice programului de studii (se preiau din planul de învățământ)²

Rezultatele învățării vizate prin disciplină		
Codul competenței	Cunoștințe și înțelegere (Knowledge and understanding)	Abilități academice specifice (Specific academic skills)
CP1	Absolventul are cunoștințe necesare pentru a concepe, modela și proiecta sisteme software complexe.	Absolventul are abilități de a realiza demersului de educare și pregătire pe diverse teme legate de dezvoltarea sistemelor software.
CP3	Absolventul este capabil să realizeze cercetări în inginerie software, în special în domeniul gândirii algoritmice și gândirii critice.	Absolventul este capabil să folosească limbajul de specialitate și terminologia specifică domeniului ingineriei software, astfel încât să poată comunica și interacționa cu membrii unor echipe de lucru.
CP5	Absolventul este în măsură să aplice cunoștințe avansate de inginerie software, plecând de la studierea la un nivel ridicat de abstractizare a diferitelor sisteme, fiind capabil să ofere soluții de implementare pentru aplicații la sisteme informatice complexe, integrate.	Absolventul cunoaște și respectă norme și reguli etice și deontologice în cercetarea științifică.
CT1	Absolventul posedă cunoștințe fundamentale de modelare prin care analizează probleme din viața reală, le transpune în cerințe concrete și elaborează un model software corespunzător.	Absolventul demonstrează că posedă cunoștințe relative la cerințele specifice demersului de cercetare în domeniul informaticii în general și al domeniului ingineriei software în special și înțelege rolul cercetării în promovarea progresului.
CT2, CT 3	Absolventul este capabil să realizeze cercetări în inginerie software, în special în domeniul gândirii algoritmice și gândirii critice.	Absolventul este capabil să folosească limbajul de specialitate și terminologia specifică domeniului ingineriei software, astfel încât să poată comunica și interacționa cu membrii unor echipe de lucru.

7. Rezultatele învățării specifice disciplinei

Cunoștințe și înțelegere (Knowledge and understanding)

învățământ, fără modificări. Dacă nu se preia nici o competență din oricare din cele două categorii, se șterge linia din tabel aferentă acelei categorii.

² Se menționează rezultatele învățării specifice programului de studiu la dezvoltarea cărora contribuie disciplina pentru care se elaborează fișa. Enunțurile, preluate fără modificări din Planul de învățământ în funcție de tipul disciplinei (DF/DS/DC) se trec în dreptul competenței asociate.

1. Absolventul este capabil de a prezenta și a explica metodele, algoritmi, paradigmele și tehnicile folosite în diferite ramuri ale informaticii.
2. Absolventul are abilitatea de a înțelege și comunica eficient informațiile.
3. Absolventul are capacitatea de a observa și obține informații din diverse surse.
Abilități academice specifice (Specific academic skills)
1. Absolventul cunoaște metodele de testare și verificare a sistemelor software.
2. Absolventul este familiarizat cu instrumentele de management de proiect, sistemele de control al versiunilor, precum și conceptele, metodele, instrumentele de continuous integration/continuous delivery (CI/CD).

8. Conținuturi

8.1 Curs	Metode de predare - învățare	Observații ³
Lecture 1. Introduction in Testing and Web Internals <ul style="list-style-type: none"> • General notions • Web Internals explained • (HTTP) Request Structure • Web System in test (Structure of web application, clients etc) • Application Sample and what we will test • Different Tools and frameworks - awareness on tools on market • Project Lifecycle - Scrum Agile methodologies System Architecture Analysis - DB	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	
Lecture 2. Web GUI Automation - Selenium <ul style="list-style-type: none"> • Programming languages testing context- C#, Javascript, Java, Ruby • Selenium IDE - WebDriver (1-2) • CSS selectors • XPath Page Object patterns	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	
Lecture 3 – 4. Web GUI Automation - Selenium <ul style="list-style-type: none"> • Maven Config • Data Driven tests • GUI Automation FWKs - Serenity, Cucumber, Specflow • BDD pros and cons • DB NoSql + Big Decimal Use of Oracles	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	

³ De exemplu aspecte organizatorice, recomandări pentru studenți, aspecte specifice legate de curs/seminar cum ar fi invitarea unor practicieni în domeniu etc.

<p>Lecture 5. JS Unit Tests</p> <ul style="list-style-type: none"> • Data generation • Coverage - examples • Javascript + unit tests • Leverage between GUI - UNIT - API tests (maybe moved in a web context course) 	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	
<p>Lecture 6. API Testing - REST, SOAP</p> <ul style="list-style-type: none"> • What you want to test • Execution types • SOAP and REST • Pros and cons in API testing 	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	
<p>Lecture 7. API Testing - REST, SOAP</p> <ul style="list-style-type: none"> • SOAP + JMeter (JAVA + jMeter jMeter + jar) • Use of APIs for Test Data setup • Test using IMAP, POP3, SMTP, FTP, HTTP Client • Proxy Tools - fiddler etc • Libraries 	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	
<p>Lecture 8. Performance Testing</p> <ul style="list-style-type: none"> • Request Analysis - yslow, page speed • Load testing • User experience - practices • Browser tools 	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	
<p>Lecture 9. Performance Testing</p> <ul style="list-style-type: none"> • Report analysis • Stress, Volume, Spyke 	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	
<p>Lecture 10- 11 Security Testing</p>	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	
<p>Lecture 12-13 Mobile Testing</p> <ul style="list-style-type: none"> • Issues • Native + Web + Embedded (hybrid) • API clients • Responsiveness • Segmentation <p>Analytics</p>	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	

<p>Lecture 14. Continuous Integration</p> <ul style="list-style-type: none"> • Jenkins, TeamCity, Bamboo • Master - Slave Setup • Parallel test execution • Selenium Grid • Continuous Delivery 	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Didactic 	
<p>Bibliografie</p> <p>Bibliography</p> <p>Books</p> <p>[Eri15] Bayo Erinle, Performance testing with JMeter, 2015 [Eri14] Bayo Erinle, JMeter CookBook, Packt Publishing, 2014 [Ava14] S. Avasara, SeleniumWebDriver Practical Guide, 2014 [Kov14] Dima Kovalenko, Selenium Design Patterns ad Best Practices, Packt Publishing , 2014 [Bur12] David Burns, Selenium 2 Testing Tools: Beginner’s guide, 2012 [Unm12] G. Unmesh, Selenium Testing CookBook, 2012 [Gra12] D. Graham, M. Fewster, Experiences of test automation: Case studies of Software Test Automation, 2012</p> <p>[Pres10] R. S. Pressman, Software engineering: a practinioner’s approach, seventh edition, Higher Education, 2010 [Kan99] C. Kaner, J. Falk, H. Nguyen, Testing Computer Software, 1999</p> <p>[Crs09] L. Crispin, J. Grecory, Agile testing: a practical guide for testers and agile teams, Addison-Wesley, 2009 [You08] M. Pezzand, M. Young, Software Testing and Analysis: Process, Principles and Techniques, John Wiley & Sons, 2008 [Nai08] K. Naik, P. Tripathy, Software testing and quality assurance. Theory and Practice, A John Wiley & Sons, Inc., 2008 [Pat05] R. Patton, Software Testing, Sams Publishing, 2005 [Mye04] Glenford J. Myers, The Art of Software Testing, John Wiley & Sons, Inc., 2004 [Brn02] I. Brnstein, Practical software testing, Springer, 2002</p> <p>Articles</p> <p>[1] Meszaros, G., Smith, S., Andrea, J, The test automation manifesto, LNCS vol 2753, pp. 73-81, 2003</p> <p>Internet resources</p> <ul style="list-style-type: none"> • Serenity, http://thucydides.info/docs/serenity-staging/ • Selenium, http://www.seleniumhq.org/ • CSS Selector, http://www.w3schools.com/cssref/css_selectors.asp • Selenium tutorial, http://software-testing-tutorials-automation.blogspot.ro/2014/01/selenium-webdriver-tutorials-basic.html <p>Tutorials</p> <p>During lectures/seminars/laboratories tutorials will be given for each assignment.</p>		

8.2 Seminar / laborator	Metode de predare - învățare	Observații
Seminar 1: Selenium IDE, CSS Selectors, Webdriver project setup (Page Object)	Presentation, Conversation, Problematizations, Discovery, Other methods – individual study, exercises	
Seminar 2 Webdriver project DDT BDT Page Object Reporting (FWKs) DB NoSql + BigDecimal – Optional (Bonus)	Presentation, Conversation, Problematizations, Discovery, Other methods – individual study, exercises	
Seminar 3 API testing in Java/JMeter	Presentation, Conversation, Problematizations, Discovery, Other methods – individual study, exercises	
Seminar 4 Performance test using JMeter	Presentation, Conversation, Problematizations, Discovery, Other methods – individual study, exercises	
Seminar 5 Security testing	Presentation, Conversation, Problematizations, Discovery, Other methods – individual study, exercises	
Seminar 6 Mobile testing Appium	Presentation, Conversation, Problematizations, Discovery, Other methods – individual study, exercises	
Bibliografie		
See from Course bibliography		
<ul style="list-style-type: none"> Cursul există în programul de studii al tuturor universităților importante din România și din străinătate; Conținutul cursului este considerat de companiile de software ca fiind important pentru abilitățile medii de programare. 		

9. Evaluare



















Tip activitate	9.1 Criterii de evaluare ⁴	9.2 Metode de evaluare ⁵	9.3 Pondere din nota finală
9.1 Curs	- cunoaște principiul de bază al domeniului;	Examen scris	50%
9.2 Seminar/laborator	- aplica conceptele cursului rezolvarea problemelor	Examinare practica	50%
9.3 Standard minim de promovare			
Observație.			

⁴ Criteriile de evaluare trebuie să reflecte direct rezultatele învățării vizate la nivel de program de studii, respectiv la nivel de disciplină. Mai concret, se evaluează achizițiile de învățare menționate în rezultatele anticipate ale învățării.

⁵ Se recomandă stabilirea atât a metodelor de evaluare finală, cât și a strategiei de evaluare pe parcurs.

- Lucrările pentru seminar/laborator nu pot fi refăcute în sesiunea de restanta.
 - Examenele scrise și practice pot fi susținute în timpul sesiunii de restanta.
 - Studenți din anii anteriori până în anul universitar curent
- o Toate regulile de mai sus se aplică studenților din anii anteriori.
- o Temele de seminar/laborator trebuie refăcute în timpul activității didactice (în cele 14 săptămâni înainte de sesiunea normală).
- Cel puțin nota 5 (de la o scară de la 1 la 10) la examenul scris. Nota finală calculată cu formula dată trebuie să fie de minim 5 pentru a promova examenul. Cel puțin nota 5 (de la o scară de la 1 la 10) la examenele scrise și practice și la activitatea de laborator/seminar.

10. Etichete ODD (Obiective de Dezvoltare Durabilă / Sustainable Development Goals)⁶

	<input type="radio"/>	Eticheta generală pentru Dezvoltare durabilă						
								
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
								Nu se aplică nici o etichetă
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Data completării:

22.05.2026

Semnătura titularului de curs

Conf. Dr. Vesca Andreea

Semnătura titularului de seminar

Conf. Dr. Vesca Andreea

Data avizării în departament:

...

Semnătura directorului de departament

Conf.dr. Adrian STERCA

⁶ Selectați o singură etichetă, cea care, în conformitate cu [Procedura de aplicare a etichetelor ODD în procesul academic](#), se potrivește cel mai bine disciplinei. Dacă disciplina tratează tema dezvoltării durabile la modul general (de ex. prin prezentarea/introducerea cadrului general al dezvoltării durabile etc.) atunci se poate alocă eticheta generală de Dezvoltare Durabilă. Dacă niciuna dintre etichete nu descrie disciplina, selectați ultima opțiune: „Nu se aplică nici o etichetă”.