

A TANTÁRGY ADATLAPJA

1. A képzési program adatai

1.1 Felsőoktatási intézmény	Babeş–Bolyai Tudományegyetem
1.2 Kar	Matematika és Informatika Kar
1.3 Intézet	Magyar Matematika és Informatika Intézet
1.4 Szakterület	Informatika
1.5 Képzési szint	Magiszteri
1.6 Szak / Képesítés	Vállalati szoftvertervezés és alkalmazásfejlesztés

2. A tantárgy adatai

2.1 A tantárgy neve	Komponens alapú és szolgáltatásorientált architektúrák Arhitecturi orientate pe servicii si bazate pe componente Component-based and service-oriented architectures						
2.2 Az előadásért felelős tanár neve	dr. Simon Károly						
2.3 A szemináriumért felelős tanár neve	dr. Simon Károly						
2.4 Tanulmányi év	1	2.5 Félév	2	2.6. Értékelés módja	vizsga	2.7 Tantárgy típusa	Kötelező – alap

3. Teljes becsült idő (az oktatási tevékenység féléves óraszama)

3.1 Heti óraszám	3	melyből: 3.2 előadás	2	3.3 szeminárium/labor	1/0
3.4 Tantervben szereplő össz-óraszám	42	melyből: 3.5 előadás	28	3.6 szeminárium/labor	14/0
A tanulmányi idő elosztása:					Óra
A tankönyv, a jegyzet, a szakirodalom vagy saját jegyzetek tanulmányozása					50
Könyvtárban, elektronikus adatbázisokban vagy terepen való további tájékozódás					30
Szemináriumok / laborok, házi feladatok, portofóliók, referátumok, esszék kidolgozása					75
Egyéni készségfejlesztés (tutorálás)					0
Vizsgák					3
Más tevékenységek:					0
3.7 Egyéni munka össz-óraszama					158
3.8 A félév össz-óraszama					200
3.9 Kreditszám					8

4. Előfeltételek (ha vannak)

4.1 Tantervi	<ul style="list-style-type: none"> Nincs
4.2 Kompetenciabeli	<ul style="list-style-type: none"> Programozási paradigmák (elsősorban objektumorientált) és haladó programozási módszerek, tervezési minták, fejlesztői környezetek ismerete

5. Feltételek (ha vannak)

5.1 Az előadás lebonyolításának feltételei	<ul style="list-style-type: none"> Videoprojektorral és táblával felszerelt előadóterem
5.2 A szeminárium / labor lebonyolításának feltételei	<ul style="list-style-type: none"> Videoprojektorral és táblával felszerelt szeminárium terem

6. Elsajátítandó jellemző kompetenciák

Szakmai kompetenciák	<ul style="list-style-type: none"> • A komponens-, illetve szolgáltatásorientált szoftverfejlesztés alapelveinek ismerete, a témával kapcsolatos legújabb irányelvek ismerete • Az elméleti alapokon túlmenően, a jelenleg használatos elterjedtebb gyakorlati implementációk terén való jártasság
Transzverzális kompetenciák	<ul style="list-style-type: none"> • Hatékony információgyűjtés, összegzés képessége • Problémamegoldó készség, kreativitás fejlesztése

7. A tantárgy célkitűzései (az elsajátítandó jellemző kompetenciák alapján)

7.1 A tantárgy általános célkitűzése	<ul style="list-style-type: none"> • Komponens alapú és szolgáltatásorientált architektúrák kiépítésére alkalmas módszerek és technológiák bemutatása
7.2 A tantárgy sajátos célkitűzései	<ul style="list-style-type: none"> • Különböző komponensmodellek és ezekre vonatkozó standardok bemutatása és összehasonlítása • A témához kapcsolódó tervezési minták megismerése • Komponens alapú fejlesztésre alkalmazható Java specifikációk és keretrendszerek (EJB, CDI, Spring IoC) bemutatása • Szolgáltatásorientált architektúrák bemutatása, példák • Webszolgáltatások fejlesztésére vonatkozó Java specifikációk és keretrendszerek (JAX-WS, JAX-RS és implementációk) bemutatása • Különböző technológiák áttekintése és összehasonlítása (pl. Java - .Net)

8. A tantárgy tartalma

8.1 Előadás	Didaktikai módszerek	Megjegyzések
1. Ismétlés, a további témák megismeréséhez szükséges alapismeretek összefoglalása <ul style="list-style-type: none"> - Programozási paradigmák áttekintése - Ismert komponens modellek és komponens-alapú architektúrák ismétlése 	magyarázat, vetítés, konverzáció, példák	
2. Bevezető: az objektumorientáltságtól a komponens-alapú fejlesztésekig és szolgáltatásorientált rendszerekig <ul style="list-style-type: none"> - Motiváció és alapelvek - Az interfészek szerepe 	magyarázat, vetítés, konverzáció, példák	
3. Komponens modellek, vonatkozó standardok <ul style="list-style-type: none"> - Osztott rendszerek (CORBA) - Dinamikus komponensmodellek (OSGi) 	magyarázat, vetítés, konverzáció, példák	
4. Szerver oldali komponensmodellek <ul style="list-style-type: none"> - Az életciklus menedzsmentje és az IoC 	magyarázat, vetítés, konverzáció, példák	

tervezési minta - Komponensek közötti függőségek, a Dependency Lookup és a Dependency Injection minták - Szerver oldali keretrendszerek és platformok általános áttekintése		
5. A Java EE szerver oldali komponensmodellje - A Java EE alkalmazások architektúrája - Az EJB specifikáció - EJB típusok, és a komponensek közötti kommunikáció - A komponensek életciklusának menedzsmentje	magyarázat, vetítés, konverzáció, példák	
6. A CDI specifikáció - CDI komponensek - Komponensek hatóköre és menedzsmentje	magyarázat, vetítés, konverzáció, példák	
7. A Spring komponensmodellje - Spring bean-ek és IoC konténer - Komponensek hatóköre és életciklusuk menedzsmentje	magyarázat, vetítés, konverzáció, példák	
8. Webszolgáltatások - Klasszikus SOAP alapú webszolgáltatások - WSDL - A REST modell	magyarázat, vetítés, konverzáció, példák	
9. Java technológiák SOAP alapú webszolgáltatások fejlesztésére - Vonatkozó specifikációk áttekintése - A JAX-WS specifikáció és implementációi, ezek felhasználása webszolgáltatások létrehozására	magyarázat, vetítés, konverzáció, példák	
10. Java technológiák RESTful webszolgáltatások fejlesztésére - A JAX-RS specifikáció és implementációi, illetve ezek felhasználása webszolgáltatások létrehozására - Adatátvitel: a DTO tervezési minta, a JAXB specifikáció és implementációi	magyarázat, vetítés, konverzáció, példák	
11. .Net komponensmodell - a .NET keretrendszer architektúrája - .NET komponens-orientált programozás - Előzmények: Component Object Model technológiák (COM/DCOM/COM+) és ezek .NET-el való együttműködőképessége - .NET-specifikus technológiák	magyarázat, vetítés, konverzáció, példák	
12. .Net technológiák szolgáltatásorientált rendszerek fejlesztésére - A keretrendszerek általános áttekintése - Webszolgáltatások	magyarázat, vetítés, konverzáció, példák	
13. További technológiák áttekintése és összehasonlító elemzése	magyarázat, vetítés, konverzáció, példák	
14. Összefoglaló ismétlés	magyarázat, vetítés, konverzáció, példák	
Könyvészet 1. Simon K., Kenyerünk Java, Egyetemi Kiadó, Kolozsvár, 2010.		

2. Rubinger A.L., Burke B., Enterprise Java Beans 3.1, 6th edition, O'Reilly, 2010. 3. ***, Java EE 6 Tutorial, Sun Microsystems, http://download.oracle.com/javaee/6/tutorial/doc/ 4. Fowler, M., Patterns of Enterprise Application Architecture, Addison-Wesley, 2002. 5. Craig Walls, Spring in Action, 3rd Edition, Manning Publications, 2011.		
8.2 Szeminárium / Labor	Didaktikai módszerek	Megjegyzések
Labor: nincs		
Szeminárium		
1. hét: projekttel és szemináriumi bemutatókkal kapcsolatos megbeszélés 2. hét: projekt specifikációjának elkészítése, fejlesztői csoportok, feladatkörök meghatározása 3. hét: a projekt fejlesztése során felhasznált, a fejlesztési folyamatot és csapatmunkát támogató eszközök bemutatása 4. hét: a projekt fejlesztéséhez használt eszközök konfigurálása, a fejlesztői környezet előkészítése 5. hét: a projekt tervének elkészítése 6. hét – 13. hét: a projekt fejlesztése, ezzel kapcsolatos megbeszélések 14. hét: projekt bemutatása	Példák, ismétlés, feladatmegoldás, egyéni és csoportos munka, párbeszéd	
Könyvészet 2. ***, The Java Tutorial, Sun Microsystems, Inc, 2004. http://java.sun.com/docs/books/tutorial/index.html 3. ***, Java EE 6 Tutorial, Sun Microsystems, http://download.oracle.com/javaee/6/tutorial/doc/		

9. Az epiztemikus közösségek képviselői, a szakmai egyesületek és a szakterület reprezentatív munkáltatói elvárásainak összhangba hozása a tantárgy tartalmával.

<ul style="list-style-type: none"> • A tantárgy tematikája nagy átfedést mutat az egyetemi oktatásban a fontosabb egyetemeken oktatott hasonló tematikájú tantárgyak tartalmával. • A tananyagok kidolgozása nemzetközileg elismert szerzők munkái alapján történt, az ajánlott könyvészet szintén a terület releváns munkái alapján van összeállítva. • A tantárgy keretein belül oktatott témák szükségesek a szoftverfejlesztői iparban történő elhelyezkedéshez, a cégek elvárják az ilyen jellegű ismereteket.
--

10. Értékelés

Tevékenység típusa	10.1 Értékelési kritériumok	10.2 Értékelési módszerek	10.3 Aránya a végső jegyben
10.4 Előadás	Vizsgafeladatok, bemutatott fogalmak és módszerek ismerete	Írásbeli vizsga	50%
10.5 Szeminárium / Labor	Évközi tevékenység és projekt értékelése	Évközi tevékenység pontozása, projektbemutató gyakorlati vizsga	50%
10.6 A teljesítmény minimumkövetelményei			
Az átmenő jegy feltételei:			
<ul style="list-style-type: none"> • A végső jegyet meghatározó minden komponens esetén kötelező az átmenő jegy (min. 5-ös). • A végső jegy minimálisan 5-ös. 			

Kitöltés dátuma

Előadás felelőse

Szeminárium felelőse

2017.10.10

dr. Simon Károly

dr. Simon Károly

Az intézeti jóváhagyás dátuma

.....

Intézetigazgató

dr. András Szilárd