

## 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						
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, alaptárgy

### 3. Teljes becsült idő (az oktatási tevékenység féléves óraszámja)

3.1 Heti óraszám	4	melyből: 3.2 előadás	2	3.3 szeminárium/labor	1/1
3.4 Tantervben szereplő össz-óraszám	56	melyből: 3.5 előadás	28	3.6 szeminárium/labor	12/12
A tanulmányi idő elosztása:					Óra
A tankönyv, a jegyzet, a szakirodalom vagy saját jegyzetek tanulmányozása					24
Könyvtárban, elektronikus adatbázisokban vagy terepen való további tájékozódás					24
Szemináriumok / laborok, házi feladatok, portofóliók, referátumok, esszék kidolgozása					48
Egyéni készségfejlesztés (tutorálás)					0
Vizsgák					4
Más tevékenységek: .....					0
3.7 Egyéni munka össz-óraszámja	96				
3.8 A félév össz-óraszámja	152				
3.9 Kreditszám	8				

### 4. Előfeltételek (ha vannak)

4.1 Tantervi	<ul style="list-style-type: none"> <li>Nincs</li> </ul>
4.2 Kompetenciabeli	<ul style="list-style-type: none"> <li>Fejlett programozási módszerek és környezetek, Java SE és .NET ismeretek, Java technológiák és keretrendszerek ismerete, web programozási és osztott rendszerekkel kapcsolatos ismeretek, tervezési minták, software engineering</li> </ul>

### 5. Feltételek (ha vannak)

5.1 Az előadás lebonyolításának feltételei	<ul style="list-style-type: none"> <li>Videoprojektorral és táblával felszerelt előadóterem</li> </ul>
5.2 A szeminárium / labor lebonyolításának feltételei	<ul style="list-style-type: none"> <li>Számítógépekkel felszerelt laborterem, a gépeken a megfelelő szoftverekkel, hozzáférés a megfelelő szerverekhez (Java EE, NetBeans/Eclipse for Java EE Developers, Glassfish alkalmazáserver, MySQL adatbázisszerver, .NET keretrendszer,</li> </ul>

	<p>Visual Studio, MSSQL adatbázisszerver, IIS szerver)</p> <ul style="list-style-type: none"> <li>Függőségek, csomagok publikus tárolókból történő letöltésére alkalmas internet kapcsolat</li> </ul>
--	---

## 6. Elsajátítandó jellemző kompetenciák

<b>Szakmai kompetenciák</b>	<ul style="list-style-type: none"> <li>Informatikai alkalmazások fejlesztése és karbantartása (C2) <ul style="list-style-type: none"> <li>Megfelelő eszközök azonosítása szoftverrendszerek fejlesztéséhez (C2.1)</li> <li>Megfelelő módszerek azonosítása szoftverrendszerek specifikálásához (C2.2)</li> <li>Megfelelő módszerek és környezetek használata számítástechnikai rendszerek specifikálására és fejlesztésére (C2.3)</li> <li>Megfelelő kritériumok és módszerek alkalmazás számítástechnikai rendszerek ellenőrzésére és értékelésére (C2.4)</li> <li>Dedikált számítástechnikai projektek megvalósítása (C2.5)</li> </ul> </li> </ul>
<b>Transzverzális kompetenciák</b>	<ul style="list-style-type: none"> <li>Szervezett és operatív tevékenységek által igényelt szabályok alkalmazása, felelősségteljes hozzáállás oktatási, tudományos és fejlesztéssel kapcsolatos témákhoz, a saját tudás és tehetség kreatív módon történő értékesítése, a szakma etikai normáinak betartása (CT1)</li> <li>A megfelelő módszerek használata a tanulásra, kutatásra és az ismeretek értékesítésére, dinamikus hozzáigazodás a társadalom elvárásaihoz, a magyar, angol és román szaknyelv ismerete, hatékony kommunikáció az említett nyelveken (CT2)</li> </ul>

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

7.1 A tantárgy általános célkitűzése	<ul style="list-style-type: none"> <li>Komponens alapú fejlesztésekre és szolgáltatásorientált architektúrák kialakítására alkalmas programozási nyelvek, eszközök, módszerek és technológiák/keretrendszerek megismerése</li> <li>Összetett szolgáltatásorientált rendszerek komponens alapú fejlesztésével kapcsolatos tervezési minták, architektúra típusok és receptek elsajátítása</li> <li>Alapvető gyakorlati ismeretek megszerzése összetett rendszerek fejlesztésével kapcsolatban</li> </ul>
7.2 A tantárgy sajátos célkitűzései	<ul style="list-style-type: none"> <li>A Java és C# programozási nyelvek haladó szintű ismerete</li> <li>Komponens alapú rendszerek architektúrájának, dinamikus komponensmodelleknek (pl. OSGi), keretrendszereknek megismerése</li> <li>A Java EE szabványcsalád, a Spring keretrendszer, valamint a .NET keretrendszer ismerete</li> <li>Alkalmazásszerverek használata</li> <li>A Java EE és Spring komponensmodelljének elsajátítása</li> <li>Az Inversion of Control és Dependency Injection minták ismerete, IoC szabványok és konténerek alkalmazása</li> <li>Perzisztenciával kapcsolatos szabványok, ORM keretrendszerek (JPA/Eclipse Link, Entity Framework) használata</li> <li>Webszolgáltatások (klasszikus XML+SOAP+WSDL alapú webszolgáltatások és RESTful szolgáltatások, Java és .Net szabványok és keretrendszerek megismerése</li> <li>Tranzakciómenedzsment, biztonság, aspektus orientált fejlesztési lehetőségek, időzítő szolgáltatások, további összetett rendszerekkel kapcsolatos haladó témakörök ismerete</li> </ul>

## 8. A tantárgy tartalma

8.1 Előadás	Didaktikai módszerek	Megjegyzések
1. Általános bevezető, adminisztratív feladatok <ul style="list-style-type: none"><li>• Syllabus ismertetése, követelmények megbeszélése</li><li>• Csapatok beosztása</li><li>• Szemináriumi bemutatók témáinak és időpontjainak rögzítése</li><li>• Tanulmánymenedzsment rendszer konfigurációja, egyéb adminisztratív problémák megoldása</li><li>• Első féléves tárgyak ismétlésével kapcsolatos gyakorlat</li></ul>	magyarázat, vetítés, konverzáció, példák	
2. C# ismétlés, áttekintő <ul style="list-style-type: none"><li>• Alaptulajdonságok, szintaktikai elemek</li><li>• Típusok</li><li>• OOP fogalmak</li><li>• Haladóbb témakörök</li><li>• Összehasonlítás a Java-val</li></ul>	magyarázat, vetítés, konverzáció, példák	
3. A .NET keretrendszer ismertetése <ul style="list-style-type: none"><li>• Általános bemutatás</li><li>• Nyelvek</li><li>• Fontosabb keretrendszerek és eszközök</li></ul>	magyarázat, vetítés, konverzáció, példák	
4. A .NET Entity Framework <ul style="list-style-type: none"><li>• Modell kialakítása</li><li>• Kapcsolatok reprezentálása</li><li>• Megszorítások leírása</li><li>• Migráció</li></ul>	magyarázat, vetítés, konverzáció, példák	
5. .NET IoC keretrendszerek <ul style="list-style-type: none"><li>• Az IoC elv</li><li>• .NET IoC keretrendszerek összehasonlítása</li><li>• A StructureMap keretrendszer (konfiguráció, dependency injection, komponensek életciklusa és menedzsmentje)</li></ul>	magyarázat, vetítés, konverzáció, példák	
6. .NET biztonság és tranzakciókezelés <ul style="list-style-type: none"><li>• Biztonsággal kapcsolatos alapfogalmak és megközelítések</li><li>• Biztonsági szintek</li><li>• Biztonsági modellek</li><li>• Tranzakciókkal kapcsolatos alapfogalmak</li><li>• Tranzakció hatókörök</li><li>• Tranzakciókezelés</li></ul>	magyarázat, vetítés, konverzáció, példák	
7. Windows Communication Foundation (WCF) <ul style="list-style-type: none"><li>• Áttekintés</li><li>• Programozási modell</li><li>• Protokollok</li><li>• Konfiguráció</li><li>• Biztonság</li></ul>	magyarázat, vetítés, konverzáció, példák	
8. WCF és webszolgáltatások <ul style="list-style-type: none"><li>• Klasszikus webszolgáltatások</li></ul>	magyarázat, vetítés, konverzáció, példák	

<ul style="list-style-type: none"> <li>• Web API RESTful szolgáltatásokhoz</li> </ul>		
<p>9. Java ismétlés</p> <ul style="list-style-type: none"> <li>• A Java programozási nyelv és a Java SE platform</li> <li>• További ismert Java technológiák (ORM keretrendszerek, webes keretrendszerek, osztott rendszerekkel kapcsolatos keretrendszerek)</li> <li>• Ismert komponensmodellek és dinamikus komponensmodellek (OSGi és implementációk)</li> <li>• Összetett alkalmazások fejlesztésével kapcsolatos ismert módszerek és minták</li> </ul>	magyarázat, vetítés, konverzáció, példák	
<p>10. Java EE és EJB alapok</p> <ul style="list-style-type: none"> <li>• A Java EE szabványcsalád</li> <li>• Java EE alkalmazáserverek</li> <li>• Szerver oldali Java komponensek (EJB alapok)</li> <li>• Alkalmazáserverek szolgáltatásai és erőforrásmenedzsment</li> </ul>	magyarázat, vetítés, konverzáció, példák	
<p>11. A Spring keretrendszer alapjai</p> <ul style="list-style-type: none"> <li>• Általános áttekintés</li> <li>• A Spring IoC konténer</li> <li>• Konfiguráció</li> <li>• Spring komponensek és hatókörök</li> <li>• Spring AOP – alapok</li> <li>• További Spring keretrendszerek (Spring DATA, Spring Security stb.)</li> </ul>		
<p>12. IoC és DI összefoglaló</p> <ul style="list-style-type: none"> <li>• Dependency lookup vs. dependency injection</li> <li>• DI általánosan</li> <li>• DI az EJB szabvány alapján</li> <li>• DI a Spring modelljének megfelelően</li> <li>• A CDI (Context and Dependency Injection) szabvány</li> </ul>		
<p>13. A Java Persistence Framework (JPA)</p> <ul style="list-style-type: none"> <li>• A JPA szabvány és implementációk (EclipseLink)</li> <li>• JPA alapok (konfiguráció, entity menedzserek, JPA entitások, perzisztencia környezet és perzisztencia egység)</li> <li>• Entitások közötti kapcsolatok reprezentálása</li> <li>• JPQL és Criteria Query API</li> <li>• Kapcsolódó témakörök (entity figyelők, callback mechanizmus stb.)</li> </ul>	magyarázat, vetítés, konverzáció, példák	
<p>14. Java EE alkalmazáserverek szolgáltatásai</p> <ul style="list-style-type: none"> <li>• Biztonság (JAAS)</li> <li>• Tranzakció menedzsment (JTA)</li> <li>• Üzenetküldés alapuló fejlesztés (JMS)</li> <li>• Interceptorok és AOP fogalmak</li> <li>• Időzítő szolgáltatás</li> </ul>	magyarázat, vetítés, konverzáció, példák	
<p>Könyvészet</p> <ol style="list-style-type: none"> <li>1. Simon K., Kenyerünk Java, Egyetemi Kiadó, Kolozsvár, 2010.</li> <li>2. Rubinger A.L., Burke B., Enterprise Java Beans 3.1, 6th edition, O'Reilly, 2010.</li> <li>3. Arun Gupta, Java EE 7 Essentials, O'Reilly Media, 2013.</li> <li>4. Eric Jendrock, Ricardo Cervera-Navarro, Ian Evans, Kim Haase, William Markito, The Java EE 7 Tutorial, Oracle, <a href="http://docs.oracle.com/javaee/7/tutorial/doc/home.htm">http://docs.oracle.com/javaee/7/tutorial/doc/home.htm</a></li> <li>5. Clarence Ho, Rob Harrop, Pro Spring 3, New York: Springer Science+Business Media, Apress Media LLC, 2012.</li> <li>6. Enterprise JavaBeans hivatalos dokumentáció, <a href="http://www.oracle.com/technetwork/">http://www.oracle.com/technetwork/</a>-</li> </ol>		

java/javaee/ejb/index.html 7. ***, Spring keretrendszer hivatalos weboldal, <a href="http://spring.io">http://spring.io</a> 8. ***, CDI specifikáció, <a href="http://www.cdi-spec.org/">http://www.cdi-spec.org/</a> 9. ***, Microsoft Development Network Library, Microsoft <a href="http://msdn.microsoft.com">http://msdn.microsoft.com</a> 10. ***, The Java Tutorial, Oracle, 2004-2014. <a href="http://docs.oracle.com/javase/tutorial/">http://docs.oracle.com/javase/tutorial/</a>
---

8.2 Szeminárium / Labor	Didaktikai módszerek	Megjegyzések
Szeminárium/gyakorlat: kettes csoportokban adott tantárgyhoz kapcsolódó témakörök kidolgozása, bemutatók készítése és megtartása, mintaalkalmazások készítése az illető témával kapcsolatban, ezek bemutatása és megbeszélése a csoportban	Feladatmegoldás és bemutatók készítése csapatban	
Gyakorlati feladatok		
1. .NET évközi projekt (első 7 alkalom) <ul style="list-style-type: none"> <li>Minta projekt témájának megbeszélése, specifikáció készítése, becslés, feladatok beosztása, környezeti elemzés, architektúra terve</li> <li>Szerver alkalmazás elkészítése, adathozzáférési és szolgáltatási réteg biztosítása, WCF szolgáltatások megírása</li> <li>Kliensalkalmazás elkészítése, RESTful API biztosítása</li> </ul> 2. Java EE évközi projekt (7-14 hét) <ul style="list-style-type: none"> <li>Minta projekt témájának megbeszélése, specifikáció készítése, becslés, feladatok beosztása, környezeti elemzés, architektúra terve</li> <li>Szerver alkalmazás készítése: adathozzáférési réteg (JPA+EJB) és szolgáltatási réteg implementációja (EJB)</li> <li>RESTful API biztosítása (JAX-RS)</li> <li>Kliensalkalmazások készítése: egy RMI-n keresztül kommunikáló és egy RESTful szolgáltatásokat használó Java konzolalkalmazás</li> </ul>	Feladatmegoldás önállóan és csapatban	

<b>Könyvészet</b> 1. Eric Jendrock, Ricardo Cervera-Navarro, Ian Evans, Kim Haase, William Markito, The Java EE 7 Tutorial, Oracle, <a href="http://docs.oracle.com/javaee/7/tutorial/doc/home.htm">http://docs.oracle.com/javaee/7/tutorial/doc/home.htm</a> 2. ***, Microsoft Development Network Library, Microsoft, <a href="http://msdn.microsoft.com">http://msdn.microsoft.com</a>
---

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

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

**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	40%

10.5 Szeminárium / Labor	Évközi feladatok	A megoldások pontozása	40%
	Gyakorlati vizsga	A megoldások pontozása	20%
10.6 A teljesítmény minimumkövetelményei			
Az átmenő jegy feltételei:			
<ul style="list-style-type: none"> <li>• A végső jegyet meghatározó minden komponens esetén kötelező az átmenő jegy (min. 5-ös).</li> <li>• A végső jegy minimálisan 5-ös.</li> </ul>			

Kitöltés dátuma

Előadás felelőse

Szeminárium felelőse

2014.04.25

dr. Simon Károly

dr. Simon Károly

Az intézeti jóváhagyás dátuma

Intézetigazgató

dr. Szenkovits Ferenc