

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	Alapképzés
1.6 Szak / Képesítés	Informatika

2. A tantárgy adatai

2.1 A tantárgy neve (hu)	Tervezői és Fejlesztői Környezetek						
2.2 Az előadásért felelős tanár neve	Jakab Hunor						
2.3 A szemináriumért felelős tanár neve	Jakab Hunor						
2.4 Tanulmányi év	3	2.5 Félév	6	2.6. Értékelés módja	vizsga	2.7 Tantárgy típusa	Kötelező - alaptárgy
2.8 A tantárgy kódja	MLM5013						

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

3.1 Heti óraszám	4	melyből: 3.2 előadás	2	3.3 szeminárium/labor	2
3.4 Tantervben szereplő össz-óraszám	48	melyből: 3.5 előadás	24	3.6 szeminárium/labor	24
A tanulmányi idő elosztása:					óra
A tankönyv, a jegyzet, a szakirodalom vagy saját jegyzetek tanulmányozása					20
Könyvtárban, elektronikus adatbázisokban vagy terepen való további tájékozódás					20
Szemináriumok / laborok, házi feladatok, portofóliók, referátumok, esszék kidolgozása					34
Egyéni készségfejlesztés (tutorálás)					
Vizsgák					3
Más tevékenységek:					
3.7 Egyéni munka össz-óraszama	77				
3.8 A félév össz-óraszama	125				
3.9 Kreditszám	5				

4. Előfeltételek (ha vannak)

4.1 Tantervi	<ul style="list-style-type: none"> Nincs
4.2 Kompetenciabeli	<ul style="list-style-type: none"> Haladó szintű java programozás, software engineering, adatbázisok

5. Feltételek (ha vannak)

5.1 Az előadás lebonyolításának feltételei	<ul style="list-style-type: none"> Videoprojektorral és táblával felszerelt előadóterem, megbízható hálózati csatlakozási lehetőséggel
--	---

5.2 A szeminárium / labor lebonyolításának feltételei	<ul style="list-style-type: none"> • Számítógépekkel és megbízható vezeték nélküli hálózattal felszerelt laborterem, hordozható számítógépek áramellátási lehetősége legyen biztosítva. • Különböző szerverekhez való csatlakozási lehetőség: Sonatype Nexus, Mysql, MsSql, Gitlab és Redmine
---	---

6. Elsajátítandó jellemző kompetenciák

Szakmai kompetenciák	<ul style="list-style-type: none"> • Haladó programozási módszerek ismerete • A különböző alkalmazás-fejlesztő eszközök felhasználása és kritikai értékelése • Adatbázisokat, multimédiás eszközöket, kliens-szerver technológiát és webszolgáltatásokat felhasználó számítógépes alkalmazások fejlesztése • Komponens-alapú, szolgáltatás orientált, illetve dinamikus komponens modelleken alapuló rendszerek fejlesztésének elsajátítása • Vállalati rendszerek fejlesztéséhez alkalmazott platformokkal kapcsolatos alapismeretek elsajátítása • Az információs/informatikai rendszerek fejlesztésére jellemző elemzési és tervezési módszerek felhasználása és meghatározott szempontok szerinti értékelése
Transzverzális kompetenciák	<ul style="list-style-type: none"> • A diák elemző és szintetizáló képességének fejlesztése. • Az új paradigmák megismerése által újfajta szemléletmód kialakítása, rálátás összetett rendszerek működésére

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

7.1 A tantárgy általános célkitűzése	<ul style="list-style-type: none"> • a szoftverrendszerek tervezésével és megvalósításával kapcsolatos ismeretek elmélyítése, rendszerezése • modern build rendszerek funkcionalitásainak megismerése • bonyolultabb szoftverrendszerek fejlesztésénél alkalmazható tervezési minták elsajátítása • osztott rendszerek fejlesztésével kapcsolatos ismeretek elsajátítása, távoli metódushívásokon, valamint üzenetküldéseken alapuló kliens-szerver alkalmazások fejlesztése • különböző programozási nyelvek által biztosított csomagok és függvénykönyvtárak egy részének áttanulmányozása • különböző tervezési és programozási környezetek megismerése (a biztosított funkcionalitásokon kívül beleértve azok felépítését is) • komponensorientált szoftverarchitektúrák megismerése, plug-in alapú
--------------------------------------	--

	<p>rendszerek fejlesztése, keretrendszerek alkalmazása</p> <ul style="list-style-type: none"> • többretegű alkalmazások fejlesztése, szolgáltatás orientált architektúrák tervezése • néhány, a modern szoftverfejlesztésben alkalmazott paradigma, modell és technológia bemutatása, alkalmazása (ORM, Rich client alkalmazások, web alkalmazások fejlesztésénél alkalmazott keretrendszerek, stb.)
7.2 A tantárgy sajátos célkitűzései	<ul style="list-style-type: none"> • Modern build rendszerek alapvető funkcionalitásainak megismerése (függőség management, sajátos műveletek definiálás, több modulós projektek kezelése). Az Ant, Maven és Gradle build rendszerek ismertetése • Haladó programozási módszerekkel kapcsolatos ismeretek kiegészítése, rendszerezése: szinkronizáció, naplózás, adatbázisok elérése, biztonság, nemzetköziesítés, reflection, XML feldolgozás stb. (slf4j, JDBC, Java Reflection API, ResourceBundles stb.) • Perzisztenciával kapcsolatos tervezési minták és magasabb szintű technológiák megismerése: többretegű szoftverarchitektúrák, kapcsolódó tervezési minták, ORM (Object Relational Mapping), JPA (Java Persistence API) specifikáció, JPA-t implementáló keretrendszerek (EclipseLink, Hibernate stb.) • Távoli metódushíváson alapuló kliens-szerver alkalmazások (CORBA, RMI) • Plug-in alapú rendszerek, dinamikus komponens modellek, vonatkozó standardok, keretrendszerek megismerése (OSGi, Equinox, Apache Karaf) • RCP (Rich Client Platform) alkalmazások fejlesztésének elsajátítása (SWT, JFace, Eclipse RCP) • A Java EE platform felépítésének megismerése

8. A tantárgy tartalma

8.1 Előadás	Didaktikai módszerek	Megjegyzések
<p>1. Modern build rendszerek ismertetése</p> <ul style="list-style-type: none"> - Általános tulajdonságok - Fontosabb funkcionalitások - Az Ant, Maven és Gradle bemutatása - Különbségek kielemezése 	magyarázat, vetítés, szemléltetés példák segítségével	
<p>2. Integrált fejlesztői környezetek</p> <ul style="list-style-type: none"> - általános bevezető, történeti áttekintés, fontosabb irányvonalak - általános funkcionalitások 	magyarázat, vetítés, szemléltetés példák	

<ul style="list-style-type: none"> - különböző környezetek rövid összehasonlítása - az Eclipse bemutatása (bevezető) - közösség, projektek, történeti áttekintő - az Eclipse IDE bemutatása 	segítségével	
<p>3. Programozási módszerekkel kapcsolatos ismétlés, ismeretek kiegészítése</p> <ul style="list-style-type: none"> - szálak szinkronizálása, kapcsolódó problémák - egyszerű hálózati alkalmazások - erőforrások betöltése, a java classpath felépítése - nemzetköziesítés és lokalizáció <ul style="list-style-type: none"> o általános tudnivalók (i18n és L10n) o nemzetköziesítés és lokalizáció megvalósítása Java-ban 	magyarázat, vetítés, szemléltetés példák segítségével, párbeszéd	
<p>4. Alkalmazások viselkedésének elemzése és befolyásolása futási időben</p> <ul style="list-style-type: none"> - Reflection <ul style="list-style-type: none"> a. általános tudnivalók b. megvalósítás Java-ban c. Java Reflection API - Alkalmazási lehetőségek, biztonsággal kapcsolatos kérdések 	magyarázat, vetítés, szemléltetés példák segítségével	
<p>5. Programozási módszerekkel, technológiákkal kapcsolatos ismétlés, ismeretek kiegészítése</p> <ul style="list-style-type: none"> - Serialization <ul style="list-style-type: none"> o általános tudnivalók o szerializáció Java-ban o protokollok, kapcsolódó fogalmak o alkalmazás, példák - XML feldolgozás <ul style="list-style-type: none"> o DTD, XSD o JAXB 	magyarázat, vetítés, szemléltetés példák segítségével	
<p>6. Adatbázisok elérése, kapcsolódó tervezési minták</p> <ul style="list-style-type: none"> - a JDBC API - többretegű szoftverarchitektúrák - az Abstract DAO Factory tervezési minta - kapcsolódó tervezési/fejlesztési eszközök 	magyarázat, vetítés, szemléltetés példák segítségével, párbeszéd	
<p>7. Objektumorientált programozás és relációs adatbázisok</p> <ul style="list-style-type: none"> - ORM (Object Relational Mapping): motiváció, általános tudnivalók - A JPA specifikáció - EclipseLink és Hibernate keretrendszerek 	magyarázat, vetítés, szemléltetés példák segítségével	
<p>8. Osztott rendszerek és távolsági metódushívásokon alapuló kliens-szerver alkalmazások</p> <ul style="list-style-type: none"> - erőforrások beazonosítása (JNDI) - az általános modell (egy Distributed Object Application általános modellje, távoli interfészek, objektumok, metódusok) - CORBA (Common Object Request Broker Architecture) <ul style="list-style-type: none"> o általános bevezető, OMG o CORBA összetevők (a kliens és a szerver oldal) 	magyarázat, vetítés, szemléltetés példák segítségével, párbeszéd	

<ul style="list-style-type: none"> ○ az IDL (Interface Definition Language), IDL-Java leképezés ○ CORBA alapú alkalmazások fejlesztése (példa) - RMI (Remote Method Invocation) 		
<p>9. Plug-in alapú, dinamikus komponens modelleken alapuló rendszerek fejlesztése</p> <ul style="list-style-type: none"> - plug-in alapú fejlesztés: motiváció, általános jellemzők, keretrendszerek - az OSGi specifikáció <ul style="list-style-type: none"> ○ OSGi bundle ○ OSGi szolgáltatások ○ OSGi alapú fejlesztések - az Equinox OSGI implementáció - OSGI modulok készítése Eclipse-ben, Maven valamint Gradle build rendszerek segítségével - az Apache Karaf osgi container ismertetése 	<p>magyarázat, vetítés, szemléltetés példák segítségével, párbeszéd</p>	
<p>10. Plug-in alapú fejlesztés Eclipse-ben</p> <ul style="list-style-type: none"> - az Eclipse felépítése, az Eclipse Workbench felépítése - Eclipse plug-in - Eclipse plug-in készítése 	<p>magyarázat, vetítés, szemléltetés példák segítségével, párbeszéd</p>	
<p>11. RCP (Rich Client Platform) alkalmazások fejlesztése</p> <ul style="list-style-type: none"> - Thin Client és Rich Client alkalmazások - az SWT grafikus csomag - a JFace grafikus eszköztár - Eclipse RCP alkalmazások fejlesztése (a fejlesztés lépései, példa) 	<p>magyarázat, vetítés, szemléltetés példák segítségével, párbeszéd</p>	
<p>12. Vállalati rendszerek fejlesztésénél használt platformok, technológiák</p> <ul style="list-style-type: none"> - vállalati rendszerek - alkalmazásszerverek - a Java EE platform - Java web programozással kapcsolatos ismeretek felfrissítése (Java Servlet API, JSP, JSTL) 	<p>magyarázat, vetítés, szemléltetés példák segítségével, párbeszéd</p>	
<p>Könyvészet</p> <ol style="list-style-type: none"> 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. Fowler, M., Patterns of Enterprise Application Architecture, Addison-Wesley, 2002. 4. Aniszczyk, J. M. J.-M. L. C., Eclipse Rich Client Platform Second Edition, Addison-Wesley, 2010 5. Raymond Gallardo Scott Hommel, S. K. J. G. S. B. Z. The Java Tutorial: A Short Course on the Basics 6th edition Addison-Wesley, 2014 6. Eclipse Documentation, Eclipse Foundation, 2015. http://www.eclipse.org/documentation/ 7. CORBA/IIOP Specifications, OMG, 		

<p>http://www.omg.org/spec/CORBA/3.3/</p> <p>8. The java RMI Tutorial, Oracle, 2015. http://docs.oracle.com/javase/tutorial/rmi/index.html</p> <p>9. EclipseLink Documentation, Eclipse foundation 2015, http://www.eclipse.org/eclipselink/documentation/</p> <p>10. OSGi Alliance, OSGi Specification Release 6 2014 https://osgi.org/download/r6/osgi.core-6.0.0.pdf</p>		
8.2 Szeminárium / Labor	Didaktikai módszerek	Megjegyzések
1. A Java programozási nyelv alapjainak ismétlését és az Eclipse környezet megismerését célzó feladatok (L1, L2)		
2. A JDBC API alkalmazása: egy többretegű alkalmazás megírása relációs adatbázisban tárolt adatok feldolgozására, a megfelelő tervezési minták alkalmazása, grafikus felszín készítése (L3, L4)		
3. Az előzőekben (2.-es pont) megírt alkalmazás átírása, ORM alkalmazása (javasolt keretrendszerek: MyBatis, Hibernate) (L5, L6)		
4. OSGi alapú szolgáltatásorientált alkalmazás létrehozása (L7)		
5. Eclipse RCP alkalmazás kifejlesztése (2-es laborfeladat felhasználói felületének lecserélése) (L9,L10)		
<p>Könyvészet</p> <p>1. Raymond Gallardo Scott Hommel, S. K. J. G. S. B. Z. The Java Tutorial: A Short Course on the Basics 6th edition <i>Addison-Wesley</i>, 2014</p> <p>2. Eclipse Documentation, Eclipse Foundation, 2015. http://www.eclipse.org/documentation/</p> <p>3. CORBA/IIOP Specifications, OMG, http://www.omg.org/spec/CORBA/3.3/</p> <p>4. The java RMI Tutorial, Oracle, 2015. http://docs.oracle.com/javase/tutorial/rmi/index.html</p> <p>5. EclipseLink Documentation, Eclipse foundation 2015, http://www.eclipse.org/eclipselink/documentation/</p> <p>6. OSGi Alliance, OSGi Specification Release 6 2014 https://osgi.org/download/r6/osgi.core-6.0.0.pdf</p> <p>7. Demers, T. O. S. M. B. The Maven Cookbook A Sonatype Open Book Mountain View, CA, 2010</p> <p>8. Tim Berglund, M. M. Building and Testing with Gradle Understanding Next-Generation Builds O'Reilly Media, 2011</p>		

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

- A tantárgy tematikája nagy átfedést mutat az egyetemi oktatásban a fontosabb egyetemeken oktatott hasonló tematikájú tantárgyak tartalmával.
- A tananyagok kidolgozása a nemzetközileg legelismertebb szerzők munkái alapján történt, az ajánlott könyvészet szintén a terület legrelevánsabb 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 technológiák ismerete	Írásbeli vizsga	40%
10.5 Szeminárium / Labor	Laborfeladatok	A megoldás pontozása	40%
	Gyakorlati vizsga	A megoldás pontozása	20%
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

13.04.2016

Előadás felelőse

Jakab Hunor

Szeminárium felelőse

Jakab Hunor

Az intézeti jóváhagyás dátuma

.....

Intézetigazgató

Dr. András Szilárd, egyetemi docens