

# SZAKKÉPZÉSI KERETTANTERV

a(z)

55 213 04

## MOBILALKALMAZÁS FEJLESZTŐ SZAKKÉPESÍTÉS-RÁÉPÜLÉSHEZ

### I. A szakképzés jogi háttere

A szakképzési kerettanterv

- a nemzeti köznevelésről szóló 2011. évi CXC. törvény,
- a szakképzésről szóló 2011. évi CLXXXVII. törvény,

valamint

- az Országos Képzési Jegyzékről és az Országos Képzési Jegyzék módosításának eljárásrendjéről szóló 150/2012. (VII. 6.) Kormányrendelet,
- az állam által elismert szakképesítések szakmai követelménymoduljairól szóló 217/2012. (VIII. 9.) Kormányrendelet,
- a(z) 55 213 04 számú, Mobilalkalmazás fejlesztő megnevezésű szakképesítés szakmai és vizsgakövetelményeit tartalmazó rendelet

alapján készült.

### II. A szakképesítés-ráépülés alapadatai

A szakképesítés-ráépülés azonosító száma: 55 213 04.

Szakképesítés-ráépülés megnevezése: Mobilalkalmazás fejlesztő

A szakmacsoport száma és megnevezése: 7. Informatika

Ágazati besorolás száma és megnevezése: XIII. Informatika

Iskolai rendszerű szakképzésben a szakképzési évfolyamok száma: 0,5 év

Elméleti képzési idő aránya: 40%

Gyakorlati képzési idő aránya: 60%

### III. A szakképzésbe történő belépés feltételei

Iskolai előképzettség: érettségi vizsga

                  vagy iskolai előképzettség hiányában: .....

Bemeneti kompetenciák: .....

Szakmai előképzettség: 54 213 05 Szoftverfejlesztő

Előírt gyakorlat: —

Egészségügyi alkalmassági követelmények: nincsenek

Pályaalkalmassági követelmények: nem szükségesek

#### **IV. A szakképzés szervezésének feltételei**

##### **Személyi feltételek**

A szakmai elméleti és gyakorlati képzésben a nemzeti köznevelésről szóló 2011. évi CXCV. törvény és a szakképzésről szóló 2011. évi CLXXXVII. törvény előírásainak megfelelő végzettséggel rendelkező pedagógus és egyéb szakember vehet részt.

Ezen túl az alábbi tantárgyak oktatására az alábbi végzettséggel rendelkező szakember alkalmazható:

<b>Tantárgy</b>	<b>Szakképesítés/Szakképzettség</b>
-	-

##### **Tárgyi feltételek**

A szakmai képzés lebonyolításához szükséges eszközök és felszerelések felsorolását a szakképesítés szakmai és vizsgakövetelménye (szvk) tartalmazza, melynek további részletei az alábbiak: Nincs.

Ajánlás a szakmai képzés lebonyolításához szükséges további eszközökre és felszerelésekre: Nincs.

#### **V. A szakképesítés-ráépülés óraterve nappali rendszerű oktatásra**

A szakképző iskolai képzés összes szakmai óraszámja 0,5 évfolyamos képzés esetén: 560 óra (16 hét x 35 óra)

A szakképző iskolai képzés összes szakmai óraszámja szabadsáv nélkül 0,5 évfolyamos képzés esetén: 504 óra (16 hét x 31,5 óra)

1. számú táblázat  
A szakmai követelménymodulokhoz rendelt tantárgyak heti óraszám

Szakmai követelménymodulok	Tantárgyak	Elméleti heti óraszám	Gyakorlati heti óraszám
	Összesen	12	19,5
	Összesen	31,5	
10837-16 Mobil alkalmazások fejlesztése	<b>Mobilalkalmazás fejlesztés</b>	12	
	<b>Mobilalkalmazás fejlesztés gyakorlata</b>		19,5

A 2. számú táblázat „A szakmai követelménymodulokhoz rendelt tantárgyak és témakörök óraszám” megadja a fent meghatározott heti óraszámok alapján a teljes képzési időre vonatkozó óraszámokat az egyes tantárgyak témaköreire vonatkozóan is (szabadsáv nélküli szakmai óraszámok).

2. számú táblázat  
A szakmai követelménymodulokhoz rendelt tantárgyak és témakörök óraszám

Szakmai követelménymodulok	Tantárgyak/témakörök	Elméleti órák száma	Gyakorlati órák száma
	Összesen	192	312
	Összesen	504	
10837-16 Mobil alkalmazások fejlesztése	<b>Mobilalkalmazás fejlesztés</b>	192	0
	Reszponzív oldalak tervezése	32	
	MVC, MVVM rendszerű alkalmazások	32	
	Webszolgáltatások (REST, SOAP)	32	
	Mobilalkalmazások tesztelése	32	
	Java programozási nyelv	32	
	Android natív alkalmazások fejlesztése	32	
	<b>Mobilalkalmazás fejlesztés gyakorlata</b>	0	312
	Reszponzív oldalak készítése		48
	MVC, MVVM rendszerű alkalmazások készítése		48
	Webszolgáltatások használata		48
	Mobilalkalmazások tesztelése		48
	Java programozási nyelv		48
	Android natív alkalmazások fejlesztése		72

A szakképzésről szóló 2011. évi CLXXXVII. törvény 8.§ (5) bekezdésének megfelelően a táblázatban a nappali rendszerű oktatásra meghatározott tanulói kötelező szakmai elméleti és gyakorlati óraszám legalább 90%-a felosztásra került.

A szakmai és vizsgakövetelményben a szakképesítésre meghatározott elmélet/gyakorlat arányának a teljes képzési idő során kell teljesülnie.

A tantárgyakra meghatározott időkeret kötelező érvényű, a témakörökre kialakított óraszám pedig ajánlás.

**A**

**10837-16 azonosító számú**

**Mobil alkalmazások fejlesztése  
megnevezésű**

**szakmai követelménymodul**

**tantárgyai, témakörei**

A 10837-16 azonosító számú Mobil alkalmazások fejlesztése megnevezésű szakmai követelménymodulhoz tartozó tantárgyak és témakörök oktatása során fejlesztendő kompetenciák

	Mobilalkalmazás fejlesztés	Mobilalkalmazás fejlesztés gyakorlata
<b>FELADATOK</b>		
Megtervezi az alkalmazás felhasználói felületét és interakcióit	x	
Megtervezi az alkalmazás kommunikációját a háttérrendszerekkel	x	
Megtervezi az eszköz-specifikus funkciókat (pl.: touchscreen, szenzorok)	x	
Megvalósítja a felhasználói felületet és az interakciókat	x	x
Integrálja a mobil alkalmazást a háttérrendszerekkel	x	x
Használja a főbb mobil operációs rendszerek (iOS, Android) fejlesztői környezetét	x	x
Képes létező webalkalmazásokhoz mobil felületet készíteni.	x	x
Megvalósítja az eszköz-specifikus funkciókat	x	x
Teszteli a rendszer felhasználói felületét	x	x
Teszteli a háttérrendszerekkel való integrációt, az alkalmazás funkcionális és minőségi követelményeit	x	x
Teszteli az eszköz-specifikus funkciókat	x	x
Fejlesztői és felhasználói dokumentációt készít	x	x
Telepítőcsomagot készít	x	x
Betartja a munka-, baleset-, tűz- és környezetvédelmi előírásokat	x	x
Angol nyelvű szakmai szöveget értelmez	x	x
<b>SZAKMAI ISMERETEK</b>		
Mobil alkalmazások lebontása együttműködő komponensekre (szoftver architektúra kialakítása)	x	
Rendszertervezési módszerek, programfejlesztési módszertanok	x	
Felhasználói felület, navigáció és interakciók tervezése	x	
Felhasználói felület prototípusok készítése	x	x
Mobil alkalmazások fejlesztői környezetének használata	x	x
Alapvető mobil alkalmazástervezési és megvalósítási minták (MVP, MVVM)	x	x
Alapvető webes integrációs eljárások és technikák	x	x
Adatbiztonsági alapok, autentikációs alapok	x	x

Web services, REST és SOAP alapok	x	x
Hibakeresés, hibajavítás, teljesítmény-optimalizáció	x	x
Alapvető felhasználói felület és interakciós technikák (pl. kiválasztás, lapozás stb.)	x	x
Mobil eszköz emulátorok használata	x	x
Online adatbázisok létrehozása, adatbázisszerver-rendszerek (MySQL, MS SQL Szerver)	x	x
Mobilalkalmazások tervezésének ergonómiai és esztétikai szempontjai	x	x
Objektum Orientált programnyelvek (Objective C, Java/JavaME) és mobil fejlesztési SDK-k ismerete	x	x
Felhasználói felületek tesztelésének alapvető módszerei	x	x
Mobil alkalmazások működésének nyomonkövetése	x	x
Mobil alkalmazások diagnosztikai és hibaelhárítási módszerei	x	x
Mobil telepítőcsomagok készítése	x	x
Munka-, baleset-, tűz- és környezetvédelmi előírások	x	x
Angol nyelvű szakmai kifejezések	x	x
<b>SZAKMAI KÉSZSÉGEK</b>		
Információk gyűjtése és elemzése	x	x
Szoftverfejlesztő környezet használata	x	x
Grafikus vázlatkészítési készség	x	x
Integrált fejlesztői keretrendszert használata	x	x
Angol nyelvű, olvasott szakmai szöveg megértése		
<b>SZEMÉLYES KOMPETENCIÁK</b>		
Pontosság	x	x
Precizitás	x	x
Fejlődőképesség, önfejlesztés	x	x
<b>TÁRSAS KOMPETENCIÁK</b>		
Együttműködés	x	x
Kapcsolatteremtő készség	x	x
Kezdeményezőkézség	x	x
<b>MÓDSZERKOMPETENCIÁK</b>		
Logikus gondolkodás	x	x
Rendszerben való gondolkodás	x	x
Problémaelemzés, -feltárás	x	x

**1.1. A tantárgy tanításának célja**

A tantárgy oktatásának alapvető célja azoknak az ismereteknek, kompetenciáknak a fejlesztése, amelyek képessé teszik a tanulót egy komplex mobilalkalmazás megtervezésére, a megvalósításhoz szükséges adatszerkezetek kiválasztására, a fejlesztői dokumentáció elkészítésére és a tesztelési tervének elkészítésére.

**1.2. Kapcsolódó szakmai tartalmak**

11625-16 Programozás és adatbázis-kezelés

12011-16 Szoftverfejlesztés

12012-16 Webfejlesztés I.

**1.3. Témakörök****1.3.1. Reszponzív oldalak tervezése****32 óra**

A témakör oktatásának a célja, hogy megismertesse a tanulókat a rezponzív, mobil eszközökre optimalizált weboldalak készítésére egy elterjedt keretrendszer bemutatásával.

Mobilalkalmazások tervezésének ergonómiai és esztétikai szempontjai.

Felhasználói felület, navigáció és interakciók tervezése.

Felhasználói felület prototípusok készítése.

Reszponzív weboldaltervezés fogalma, alapjai, mobil-first filozófia lényege.

Reszponzív tervezési minták.

Bootstrap keretrendszer jellemzői, telepítése, file-struktúra.

CSS előfeldolgozó (LESS), változók, műveletek, egymásba ágyazás (nesting).

Bootstrap használatának lépései.

Reszponzív elrendezés kialakítása Bootstrap CSS-el.

Reszponzív osztályok, jelölők (xs, sm, md, lg) szerepe.

Képek megjelenítése.

A Grid rendszer szerepe, jellemzői.

Adatbeviteli formok készítése.

Hasznos segédosztályok jellemzői (.sr-only, .show, .hide, .pull-left, .pull-right, .text-\*, .bg-\*, .table\*, .table-responsive).

Bootstrap komponensek: Ikonok (Glyphicons), navigációs menü, jelvények, figyelmeztetések, parancsgomb csoportok, panelek, Jumbotron-ok, navigációs elemek.

Bootstrap JavaScript kiegészítők.

Bootstrap testreszabása, a rendszer fordításának eszközei, lépései.

A Bootstrap kiegészítése (pl.: Bootswatch, tree-view vezérlő, WYSIWYG editor).

Felhasználói felületek tesztelésének alapvető módszerei.

**1.3.2. Java MVC rendszerű alkalmazások****32 óra**

A témakör oktatásának a célja, hogy megismertesse a tanulókat Java alapú (J2EE) Spring MVC webfejlesztés alapjaival.

A Java Enterprise Edition alapjai, Java platformok.

A többretegű web alkalmazások fejlesztésének alapjai, követelményei, többretegű alkalmazások komponensei, az egyes rétegek szerepe.

Mobil alkalmazások lebontása együttműködő komponensekre (szoftver architektúra kialakítása)

Az MVC tervezési minta.  
 Spring MVC keretrendszer bemutatása, a rendszer felépítése.  
 Spring Core kontroller (Core, Beans, Context, Expression Language).  
 Aspect-Oriented Programming (AOP) modul.  
 Adatelérési és integrációs modul (JDBC, ORM, OXM, JMS).  
 Web modul (servlet, struts, portlet).  
 Test modul (JUnit, TestNG).  
 Fejlesztői környezet elemeinek bemutatása, telepítésük (Eclipse for JEE, Java SE 8, Maven 3, Tomcat 8, Spring, Spring MVC, SpringSource Tool Suite), az első egyszerű Spring alkalmazás.  
 IoC (Inversion of Control) koncepció.  
 Spring Core konténer. XML bázisú bean konfiguráció, BeanFactory és ApplicationContext interfészek implementációja.  
 Dependency Injection (DI) tervezési minta alkalmazása a Spring keretrendszerben: Setter Injection, Constructor Injection.  
 Spring bean-ek és életciklusuk.  
 Data Access Object (DAO) tervezési minta és réteg, CRUD műveletek.  
 Spring JDBC keretrendszer.  
 DataSource és JdbcTemplate osztályok, CRUD metódusok.  
 ORM technika lényege, a Hibernate architektúra, a Hibernate Query Language (HQL) Spring integrációja.  
 HQL és HCQL lekérdezések.  
 Spring Web MVC keretrendszer: Front Controller tervezési minta, egyszerű Spring MVC alkalmazások fejlesztése (DispatcherServlet, kontrollerek, ViewResolver, kivételkezelés).  
 Spring security alapok, konfiguráció. BASE és FORM alapú autentikáció, saját authentication provider használata.  
 Fontosabb konfigurációs beállítások, controller- és szervizréteg védelme.  
 Spring alkalmazások tesztje: JUnit4, TestNG, agilis egység- és integrációs tesztek.

### **1.3.3. Webszolgáltatások**

**32 óra**

A témakör oktatásának célja, hogy a tanuló megismerje a webszolgáltatások fajtáit, szerepüket Java fejlesztői környezet használatával.

Webszolgáltatások áttekintése

SOA architektúra jellemzői, tervezése, alkalmazása webszolgáltatásokkal

Java szabványos API-k webszolgáltatásokhoz (SOAP JAX-WS, RESTful JAX-RS)

SOAP üzenetek szerepe, felépítése, műveletek, üzenetek és hibák.

WSDL dokumentum.

SOAP kliensek Java implementációja.

SOAP szolgáltatások paraméterezése.

Contract-first és Code-first szolgáltatások.

RESTful webszolgáltatások.

REST (Representational State Transfer) jellemző.

SOAP és RESTful szolgáltatások összehasonlítása.

RESTful szolgáltatások implementálása, terjesztése JAX-RS alkalmazásával.

Szerver oldali támogatás AJAX kliensek számára.

JavaScript Object Notation (JSON) alkalmazása.

RESTful kliensek fejlesztése.

REST üzenetek (kérések és válaszok kezelése JAX-RS-el).

XML adatkötések (JAXB).

Aszinkron üzenetek kezelése.  
Webszolgáltatások védelme, autentikáció és titkosítás (SSL/TSL).  
Kliensek hitelesítése és hozzáférés vezérlése.

#### **1.3.4. Mobilalkalmazások tesztelése 32 óra**

A témakör oktatásának célja, hogy a tanuló megismerjen egy MVC rendszerű mobilalkalmazás tesztelésében elterjedt rendszert, technológiát.  
Alapvető RESTful és SOAP webszolgáltatások fejlesztése és tesztelése SoapUI és JUnit használatával.  
Adatvezérelt tesztek készítése külső adatforrások használatával.  
Teszt suit-ok, tesztesetek összeállítása.  
A MockService modell, a szolgáltatás indítása.  
Dinamikus REST és SOAP Mock szolgáltatások fejlesztése és terjesztése.  
Tesztautomatizálás szkriptek (pl.: Groovy) használatával.  
Log állományok vizsgálata.  
Jelentések készítése.  
Biztonságos webszolgáltatások tesztelésének lehetőségei.  
Terheléses tesztek.

#### **1.3.5. Java programozási nyelv 32 óra**

A témakör oktatásának célja, hogy a tanulók megismerjék a Java programozási nyelv alapvető elemeit, az objektum orientált programozás alapjait.  
Java programozási nyelv jellemzői, verzió, SDK telepítése.  
IDE fajtái, kiválasztása, telepítése.  
Programkészítés lépései.  
Adattípusok fogalma, jellemző, kiválasztásának szempontjai.  
Egyszerű és referenciatípusok a Java programozási nyelvben.  
Változó fogalma, deklarációjának szintaxisa, azonosítók, literálok, típuskényszerítés.  
Decimális számok adatvesztései float/double adattípusok alkalmazása esetén.  
Felsorolt típus fogalma, definiálása, alkalmazása.  
Operátorok (deklaráció, inicializáció, értékadás, aritmetikai operátorok, sztringek összefűzése, relációs operátorok, logikai operátorok, feltételes operátor, bitenkénti logikai operátorok).  
Kifejezések fogalma, készítése, a kiértékelés sorrendje, precedenciák.  
Vezérlési szerkezetek (szekvencia, szelekciók, iterációk, break és continue utasítások, üres utasítás).  
Osztályok és objektumok: deklaráció, tagtípusok, metódusok definiálása és hívása, aktuális és formális paraméterek, paraméterátadás módszerei, a main() metódus szerepe, példányosítás, hozzáférés vezérlés, import deklarációk, konstruktorok, inicializációk, final foglalt szó, Varargs metódus, generikus osztályok bemutatása egyszerű példákon keresztül.  
Az Object osztály, hash kód fogalma, szerepe, generálása, objektumok egyenlősége, klónozása, finalize() metódus, az Object osztály metódusai, csomagoló (wrapper) osztályok, be- és kicsomagolás.  
Kivételek fogalma, kezelése, az Exception osztály hierarchiája.  
Karakterláncok jellemzőik, speciális karakterek, sztringek kezelése.  
Dátumok és időpontok kezelése, formázása, konvertálása.  
Adatok formázása, konverziós karakterek.  
Reguláris kifejezések fogalma, készítése, a Pattern és Matcher osztályok

Tömb fogalma, fajtái, jellemzőik, kezelésük, változó hosszúságú tömbök, tömbparaméterek.

Haladó OOP ismeretek: öröklés fogalma, alkalmazása, az objektumok életciklusa, metódusok felüldefiniálása, osztálytagok elrejtése, absztrakt osztályok és metódusok.

Interfész fogalma, szerepe, definiálása, implementálása.

Garbage collection alapelvei, memory leak.

### **1.3.6. *Android natív alkalmazások fejlesztése* 32 óra**

A témakör oktatásának a célja, hogy a tanuló megismerje az Android natív alkalmazások készítésének eszközeit és módszereit.

Fejlesztéshez szükséges eszközök jellemző, telepítésük lépései.

JDK (Java Development Kit) telepítése, konfigurálása.

Android Studio és komponenseinek (Android SDK, Android Virtual Device) telepítése, beállítása, egyszerű projekt készítése és futtatása virtuális és fizikai mobileszközön.

Android Studio felépítése, alapvető kezelése, konfigurálása.

Programok készítése Android Studioval. Osztályok, gettlerek, settlerek definiálása, metódusok felüldefiniálása, fontosabb metódusok használata.

Élő sablonok (live templates) alkalmazása.

Forráskód készítését, szerkesztését támogató módszerek, kód refactoring.

Komplex projekt (pl.: ToDo alkalmazás) készítése:

Felhasználói felület készítése a Visual Designer eszközzel.

Felhasználói felület tervének módosítása a leíró XML állomány szerkesztésével.

Vezérlők fajtái, alkalmazásuk.

Adatmodell készítése.

Adatok tárolása az SQLite API osztály használatával, CRUD műveletek.

Felhasználó beavatkozások fajtái, kezelésük.

Események és eseménykezelő metódusok.

Párbeszédablakok tervezése, implementálása.

Verziókezelő rendszer (GIT) telepítése, szolgáltatásai, alapvető funkcióinak használata az Android Studioval (pull, push, branching, commits, reverting, merging, rebasing, remotes).

Az Activity osztály fontosabb metódusai, használata.

Az elrendezés (layout) osztályai, módszerei.

MVC alkalmazás készítése Android Studioval.

Hibakeresés, hibajavítás és a teljesítmény-optimalizáció eszközei és módszerei.

Alkalmazások tesztelése, MonkeyRunner eszköz használata.

Lint használata a forrásállományok ellenőrzésére.

Speciális eszközök használata: Android Device Monitor, Thread Monitor, Heap Monitor, Network Statistics, stb.

Mobil alkalmazások terjesztése, telepítőcsomagok.

## **1.4. A képzés javasolt helyszíne (ajánlás)**

*Számítógép terem*

## **1.5. A tantárgy elsajátítása során alkalmazható sajátos módszerek, tanulói tevékenységformák (ajánlás)**

### 1.5.1. A tantárgy elsajátítása során alkalmazható sajátos módszerek (ajánlás)

Sorszám	Alkalmazott oktatási módszer neve	A tanulói tevékenység szervezeti kerete			Alkalmazandó eszközök és felszerelések
		egyéni	csoporth	osztály	
1.	magyarázat			x	
2.	kiselőadás	x	x		
3.	megbeszélés		x		
4.	vita		x	x	
5.	szemléltetés			x	
6.	projekt		x		
7.	kooperatív tanulás		x		
8.	szimuláció	x			
9.	feladatmegoldás	x	x	x	
10.	házi feladat	x			

### 1.5.2. A tantárgy elsajátítása során alkalmazható tanulói tevékenységformák (ajánlás)

Sorszám	Tanulói tevékenységforma	Tanulói tevékenység szervezési kerete (differenciálási módok)			Alkalmazandó eszközök és felszerelések
		egyéni	csoporth-bontás	osztály-keret	
1.	Információ feldolgozó tevékenységek				
1.1.	Olvasott szöveg önálló feldolgozása	x			
1.2.	Olvasott szöveg feladattal vezetett feldolgozása	x			
1.3.	Olvasott szöveg feldolgozása jegyzeteléssel	x			
1.4.	Hallott szöveg feldolgozása jegyzeteléssel			x	
1.5.	Hallott szöveg feladattal vezetett feldolgozása			x	
1.6.	Információk önálló rendszerezése	x			
1.7.	Információk feladattal vezetett rendszerezése	x			
2.	Ismeretalkalmazási gyakorló tevékenységek, feladatok				
2.1.	Írásos elemzések készítése	x	x		
2.2.	Leírás készítése	x	x		
2.3.	Válaszolás írásban mondatszintű kérdésekre	x			
2.4.	Tesztfeladat megoldása	x			
2.5.	Szöveges előadás egyéni felkészüléssel	x			
2.6.	Tapasztalatok utólagos ismertetése szóban		x	x	
2.7.	Tapasztalatok helyszíni ismertetése szóban		x		
3.	Képi információk körében				
3.1.	rajz értelmezése	x		x	

3.2.	rajz készítése leírásból	x			
3.3.	rajz kiegészítés	x			
3.4.	rajz elemzés, hibakeresés	x		x	
4.	Komplex információk körében				
4.1.	Elemzés készítése tapasztalatokról		x		
5.	Csoportos munkaformák körében				
5.1.	Feladattal vezetett kiscsoportos szövegfeldolgozás		x		
5.2.	Információk rendszerezése mozaikfeladattal		x		
5.3.	Kiscsoportos szakmai munkavégzés irányítással		x		
5.4.	Csoportos helyzetgyakorlat		x		
5.5.	Csoportos versenyjáték		x		

### 1.6. A tantárgy értékelésének módja

A nemzeti köznevelésről szóló 2011. évi CXCV. törvény. 54. § (2) a) pontja szerinti értékeléssel.

## 2. Mobilalkalmazás fejlesztés gyakorlata tantárgy

312 óra

### 2.1. A tantárgy tanításának célja

A tantárgy oktatásának alapvető célja azoknak az ismereteknek, képességeknek a fejlesztése, amelyek képessé teszik a tanulót egy komplex mobilalkalmazás elkészítésére, kódolására és tesztelésére.

### 2.2. Kapcsolódó szakmai tartalmak

11625-16 Programozás és adatbázis-kezelés

12011-16 Szoftverfejlesztés

12012-16 Webfejlesztés I.

### 2.3. Témakörök

#### 2.3.1. *Reszponzív oldalak készítése*

48 óra

A témakör oktatásának a célja, hogy megismertesse a tanulókat a rezponzív weboldalak készítésével egy elterjedt keretrendszer használatán keresztül.

Bootstrap keretrendszer, telepítése, file-struktúra kialakítása.

Bootstrap használatának lépései.

Reszponzív elrendezés kialakítása Bootstrap CSS-el.

Reszponzív osztályok, jelölők (xs, sm, md, lg) alkalmazása.

Képek megjelenítése.

A Grid rendszer használata.

Adatbeviteli formok készítése.

Hasznos segédosztályok (.sr-only, .show, .hide, .pull-left, .pull-right, .text-\*, .bg-\*, .table\*, .table-responsive) alkalmazása.

Bootstrap komponensek: Ikonok (Glyphicons), navigációs menü, jelvények, figyelmeztetések, parancsgomb csoportok, panelek, Jumbotron-ok, navigációs elemek használata.

Bootstrap JavaScript kiegészítők alkalmazása.

Bootstrap testreszabása, a rendszer fordításának lépései.

A Bootstrap kiegészítése (pl.: Bootswatch, tree-view vezérlő, WYSIWYG editor).  
A felhasználói felület tesztelése.

### **2.3.2. Java MVC rendszerű alkalmazások készítése 48 óra**

A témakör oktatásának a célja, hogy megismertesse a tanulókat Java alapú (J2EE) Spring MVC webfejlesztés alapjaival.

Fejlesztői környezet elemeinek telepítése, beállítása (Eclipse for JEE, Java SE 8, Maven 3, Tomcat 8, Spring, Spring MVC, SpringSource Tool Suite), az első egyszerű Spring alkalmazás készítése és futtatása.

IoC (Inversion of Control) koncepció.

Spring Core konténer. XML bázisú bean konfiguráció, BeanFactory és ApplicationContext interfészek implementációja.

Dependency Injection (DI) tervezési minta alkalmazása a Spring keretrendszerben: Setter Injection, Constructor Injection.

Data Access Object (DAO) tervezési minta alkalmazása, CRUD műveletek.

Spring JDBC keretrendszer használata.

DataSource és JdbcTemplate osztályok, CRUD metódusok alkalmazása.

HQL és HCQL lekérdezések készítése.

Spring Web MVC keretrendszer: Front Controller tervezési minta alkalmazása, egyszerű Spring MVC alkalmazások fejlesztése (DispatcherServlet, kontrollerek, ViewResolver, kivételkezelés).

Spring security konfiguráció. BASE és FORM alapú autentikáció alkalmazása, saját authentication provider használata. Fontosabb konfigurációs beállítások, controller- és szervizréteg védelme.

Spring alkalmazások tesztje: JUnit4, TestNG, agilis egység és integrációs tesztek.

### **2.3.3. Webszolgáltatások használat 48 óra**

A témakör oktatásának célja, hogy a tanuló megtanulja a webszolgáltatások alkalmazását Java fejlesztői környezet használatával.

Java szabványos API-k alkalmazása webszolgáltatásokhoz (SOAP JAX-WS, RESTful JAX-RS).

SOAP üzenetek, műveletek, üzenetek és hibák alkalmazása.

SOAP kliensek Java implementációja.

SOAP szolgáltatások paraméterezése.

RESTful szolgáltatások implementálása, terjesztése JAX-RS alkalmazásával.

JavaScript Object Notation (JSON) alkalmazása.

RESTful kliensek fejlesztése.

REST üzenetek (kéresek és válaszok kezelése JAX-RS-el).

XML adatkötések (JAXB) alkalmazása.

Aszinkron üzenetek kezelése.

Webszolgáltatások védelme, autentikáció és titkosítás (SSL/TSL).

Kliensek hitelesítése és hozzáférés vezérlése.

### **2.3.4. Mobilalkalmazások tesztelése 48 óra**

A témakör oktatásának célja, hogy a tanuló megismerjen egy MVC rendszerű mobilalkalmazás tesztelésében elterjedt rendszert, technológiát.

Alapvető RESTful és SOAP webszolgáltatások fejlesztése és tesztelése SoapUI és JUnit használatával.

Adatvezérelt tesztek készítése külső adatforrások használatával.

Teszt suit-ok, tesztesetek összeállítása.

A MockService modell, a szolgáltatás indítása.  
Dinamikus REST és SOAP Mock szolgáltatások fejlesztése és terjesztése.  
Tesztautomatizálás szkriptek (pl.: Groovy) használatával.  
Log állományok vizsgálata.  
Jelentések készítése.  
Biztonságos webszolgáltatások tesztelésének lehetőségei.  
Terheléses tesztek készítése.

### **2.3.5. Java programozási nyelv 48 óra**

A témakör oktatásának célja, hogy a tanuló megtanulja a Java programozási nyelv alkalmazását és az objektum orientált programozás módszereit Java környezetben.  
Java programozási nyelv SDK telepítése.  
IDE kiválasztása, telepítése.  
Programkészítés lépései.  
Adattípusok alkalmazása.  
Egyszerű és referenciatípusok használata a Java programozási nyelvben.  
Változó deklaráció, azonosítók, literálok, típuskényszerítés alkalmazása.  
Felsorolt típus definiálása, alkalmazása.  
Operátorok (deklaráció, inicializáció, értékadás, aritmetikai operátorok, sztringek összefűzése, relációs operátorok, logikai operátorok, feltételes operátor, bitenkénti logikai operátorok) használata.  
Kifejezések készítése.  
Vezérlési szerkezetek (szekvencia, szelekciók, iterációk, break és continue utasítások, üres utasítás) alkalmazása.  
Osztályok és objektumok: deklaráció, tagtípusok, metódusok definiálása és hívása, aktuális és formális paraméterek, paraméterátadás módszerei, a main() metódus szerepe, példányosítás, hozzáférés vezérlés, import deklarációk, konstruktorok, inicializációk, final foglalt szó, Varargs metódus, generikus osztályok alapszintű használata.  
Az Object osztály, hash kód generálása, objektumok egyenlőségének vizsgálata, klónozása, finalize() metódus, az Object osztály metódusai, csomagoló (wrapper) osztályok, be- és kicsomagolás.  
Kivételek kezelése, az Exception osztály használata.  
Karakterláncok kezelése.  
Dátumok és időpontok kezelése, formázása, konvertálása.  
Adatok formázása, konverziós karakterek használata.  
Reguláris kifejezések készítése, a Pattern és Matcher osztályok alkalmazása.  
Tömbök kezelése, változó hosszúságú tömbök, tömbparaméterek.  
Haladó OOP ismeretek: öröklés alkalmazása, metódusok felüldefiniálása, osztálytagok elrejtése, absztrakt osztályok és metódusok alkalmazása.  
Interfészek definiálása, implementálása.

### **2.3.6. Android natív alkalmazások fejlesztése 72 óra**

A témakör oktatásának a célja, hogy a tanuló megismerje az Android natív alkalmazások készítésének eszközeit és módszereit.  
Fejlesztéshez szükséges eszközök jellemző, telepítésük lépései.  
JDK (Java Development Kit) telepítése, konfigurálása.  
Android Studio és komponenseinek (Android SDK, Android Virtual Device) telepítése, beállítása, egyszerű projekt készítése és futtatása virtuális és fizikai mobil eszközön.

Android Studio felépítése, alapvető kezelése, konfigurálása.  
 Programok készítése Android Studioval. Osztályok, gettlerek, settlerek definiálása, metódusok felüldefiniálása, fontosabb metódusok használata.  
 Élő sablonok (live templates) alkalmazása.  
 Forráskód készítését, szerkesztését támogató módszerek, kód refactoring.  
 Komplex projekt (pl.: ToDo alkalmazás) készítése:  
 Felhasználói felület készítése a Visual Designer eszközzel.  
 Felhasználói felület tervének módosítása a leíró XML állomány szerkesztésével.  
 Vezérlők fajtái, alkalmazásuk.  
 Adatmodell készítése.  
 Adatok tárolása az SQLite API osztály használatával, CRUD műveletek.  
 Felhasználó beavatkozások fajtái, kezelésük.  
 Események és eseménykezelő metódusok.  
 Párbeszédablakok tervezése, implementálása.  
 Verziókezelő rendszer (GIT) telepítése, szolgáltatásai, alapvető funkcióinak használata az Android Studioval. (pull, push, branching, commits, reverting, merging, rebasing, remotes).  
 Az Activity osztály fontosabb metódusai, használata.  
 Az elrendezés (layout) osztályai, módszerei.  
 MVC alkalmazás készítése Android Studioval.  
 Hibakeresés, hibajavítás és a teljesítmény-optimalizáció alkalmazása.  
 Alkalmazások tesztelése, MonkeyRunner eszköz használata.  
 A Lint használata a forrásállományok ellenőrzésére.  
 Speciális eszközök használata: Android Device Monitor, Thread Monitor, Heap Monitor, Network Statistics, stb.).  
 Mobil alkalmazások terjesztése, telepítőcsomagok összeállítása.

#### 2.4. A képzés javasolt helyszíne (ajánlás)

*Számítógép terem*

#### 2.5. A tantárgy elsajátítása során alkalmazható sajátos módszerek, tanulói tevékenységformák (ajánlás)

##### 2.5.1. A tantárgy elsajátítása során alkalmazható sajátos módszerek (ajánlás)

Sorszám	Alkalmazott oktatási módszer neve	A tanulói tevékenység szervezeti kerete			Alkalmazandó eszközök és felszerelések
		egyéni	csoporthoz	osztály	
1.	magyarázat			x	
2.	kiselőadás	x	x		
3.	megbeszélés		x		
4.	vita		x	x	
5.	szemléltetés			x	
6.	projekt		x		
7.	kooperatív tanulás		x		
8.	szimuláció	x			
9.	feladatmegoldás	x	x	x	
10.	házi feladat	x			

## 2.5.2. A tantárgy elsajátítása során alkalmazható tanulói tevékenységformák (ajánlás)

Sorszám	Tanulói tevékenységforma	Tanulói tevékenység szervezési kerete (differenciálási módok)			Alkalmazandó eszközök és felszerelések
		egyéni	csoporthatás	osztálykeret	
1.	Információ feldolgozó tevékenységek				
1.1.	Olvasott szöveg önálló feldolgozása	x			
1.2.	Olvasott szöveg feladattal vezetett feldolgozása	x			
1.3.	Olvasott szöveg feldolgozása jegyzeteléssel	x			
1.4.	Hallott szöveg feldolgozása jegyzeteléssel			x	
1.5.	Hallott szöveg feladattal vezetett feldolgozása			x	
1.6.	Információk önálló rendszerezése	x			
1.7.	Információk feladattal vezetett rendszerezése	x			
2.	Ismeretalkalmazási gyakorló tevékenységek, feladatok				
2.1.	Írásos elemzések készítése	x	x		
2.2.	Leírás készítése	x	x		
2.3.	Válaszolás írásban mondat szintű kérdésekre	x			
2.4.	Tesztfeladat megoldása	x			
2.5.	Szöveges előadás egyéni felkészüléssel	x			
2.6.	Tapasztalatok utólagos ismertetése szóban		x	x	
2.7.	Tapasztalatok helyszíni ismertetése szóban		x		
3.	Képi információk körében				
3.1.	rajz értelmezése	x		x	
3.2.	rajz készítése leírásból	x			
3.3.	rajz kiegészítés	x			
3.4.	rajz elemzés, hibakeresés	x		x	
4.	Komplex információk körében				
4.1.	Elemzés készítése tapasztalatokról		x		
5.	Csoportos munkaformák körében				
5.1.	Feladattal vezetett kiscsoportos szövegfeldolgozás		x		
5.2.	Információk rendszerezése mozaikfeladattal		x		
5.3.	Kiscsoportos szakmai munkavégzés irányítással		x		
5.4.	Csoportos helyzetgyakorlat		x		
5.5.	Csoportos versenyjáték		x		

6.	Gyakorlati munkavégzés körében				
6.1.	Műveletek gyakorlása	x	x		

### **2.6. A tantárgy értékelésének módja**

A nemzeti köznevelésről szóló 2011. évi CXC. törvény. 54. § (2) a) pontja szerinti értékeléssel.