


# SZAKÉRTŐI VÉLEMÉNY

Képzés (képzési program) megnevezése	Junior Fullstack API fejlesztő
Felnőttképző megnevezése és engedélyszáma:	QTC Informatikai és Szolgáltató Korlátolt Felelősségű Társaság E/2020/000075
Szakértői megállapítások	
<ol style="list-style-type: none"><li>1. A képzési program tartalma megfelel a felnőttképzésről szóló 2013. évi LXXVII. törvénynek és szakmai oktatás vagy szakmai képzés esetén a szakképzésről szóló törvénynek és a szakképzésről szóló törvény végrehajtásáról szóló kormányrendeletnek.</li><li>2. A képzési programban meghatározott tartalommal, feltételekkel és módon, valamint a képzéssel érintett célcsoport számára megszerezhető a képzési programban megjelölt kompetenciák.</li><li>3. A képzési program minden oldala folyamatos oldalszámozással van ellátva, és az összefűzésre úgy került sor, hogy annak szétválasztására sérülésmentesen nincs lehetőség.</li></ol>	
Szakértői vélemény kelte	Budapest, 2021.06.04.
Felnőttképzési szakértő neve, nyilvántartási száma	Pádár Tivadar FSZ/2020/000057
Felnőttképzési szakértő aláírása	

# KÉPZÉSI PROGRAM

## SZAKMAI KÉPZÉS

**JUNIOR FULLSTACK API FEJLESZTŐ**

**(PROGRAMKÖVETELMÉNY AZONOSÍTÓ: 06135010)**

**Vonatkozó jogszabályok:**

**2019. évi LXXX. törvény a szakképzésről**  
**12/2020. (II. 7.) Korm. rendelet**  
**2013. évi LXXVII. törvény**  
**11/2020. (II. 7.) Korm. rendelet**

## 1. Alapadatok

<b>A programkövetelmény alapján szervezhető szakmai képzés:</b>	
1.1.	Megnevezése: <b>Junior Fullstack API fejlesztő</b>
1.2.	Ágazat megnevezése: <b>Informatika és távközlés ágazat</b>
1.3.	Besorolása a képzési területek egységes osztályozási rendszere (KEOR) szerinti kód alapján: <b>0613 Szoftverek és alkalmazások fejlesztése és elemzése</b>
<b>A programkövetelmény alapján szervezhető szakmai képzéssel megszerezhető szakképesítés:</b>	
1.4.	Megnevezése: <b>Junior Fullstack API fejlesztő</b>
1.5.	Az Európai Képesítési Keretrendszer (EKKR) szerinti szint: <b>5</b>
1.6.	A Magyar Képesítési Keretrendszer (MKKR) szerint szint: <b>5</b>
1.7.	A Digitális Kompetencia Keretrendszer szerinti szint: <b>6</b>
<b>A programkövetelmény alapján szervezhető szakmai képzéssel megszerezhető szakképesítés és az azzal betölthető munkakör vagy végezhető tevékenység kapcsolata, összefüggése:</b>	
<p>A szakmai képzéshez kapcsolódóan megszerezhető szakképesítéshez szükséges kompetenciákkal szakmajegyzékben szereplő szakma körébe vonható munkaterület, tevékenység vagy munkakör magasabb szinten gyakorolható, vagy a szakmai képzés szakmajegyzékben szereplő szakma képzési és kimeneti követelményeiben meg nem határozott speciális szakmai ismeretek és szakmai készségek megszerzésére irányul.</p> <p>1.8. A Junior Fullstack API fejlesztő azon túl, hogy képes létező API-k kliens-oldali használtára, saját fejlesztésű API-k tervezésére és megírására is képes. Az alkalmazásokhoz tesztek készítését, használja a RESTful alapú API szabványokat, valamint az azon alapuló API-tervezés és dokumentáció-generálás folyamatát és eszközeit. Képes a szolgáltatásokhoz és erőforrásokhoz való hozzáférést jogosultságokhoz kötötten szabályozni. Tisztában van a NoSQL megoldások alapvető fajtáit és használatát, tisztában van azok előnyeivel és hátrányaival. Érti a legfontosabb szektorokban általánosan használt API-k jelentőségét és felhasználásuk módjait. Ismeri a konténerizációs technológia előnyeit, és képes annak segítségével az API-t kiszolgáló és felhasználó alkalmazásokat publikálni.</p>	
<b>A képzés célja:</b>	
1.9.	<p>A képzés célja, hogy a képzésben résztvevő sajátítsa el a Junior Fullstack API fejlesztő szakképesítés megszerzéséhez szükséges elméleti és gyakorlati ismereteket, készségeket és kompetenciákat.</p> <p>A képzés célja, hogy a résztvevők megismerkedjenek különböző webfejlesztési módszerekkel, gyakorlatot szerezzenek ezek használatában. A képzés során a résztvevők elsajátítják az algoritmikus gondolkodás alapjait, és mentorálás mellett ismereteik segítségével össze tudnak állítani frontend alkalmazásokat.</p>
<b>A képzés célcsoportja:</b>	
1.10.	<p>A képzés célcsoportját jelenti minden olyan személy, aki a belépési feltételeknek megfelel és a képzési programmal elérhető ismeretek, készségek és kompetenciák megszerzését tűzte ki célként maga elé. A képzés célcsoportját azok a személyek alkotják, akik érdeklődést mutatnak az informatika világa iránt, nagy bennük a tudásvágy és igényesek a munkájukra.</p>
<b>Megszerezhető kompetenciák:</b>	
1.11.	<ul style="list-style-type: none"> <li>● Ismeri a REST webszolgáltatás-architektúrát, és annak kliensoldali használatát</li> <li>● Ismer legalább egy backend-oldali technológiát, aminek segítségével képes API-végpontokat fejleszteni</li> <li>● Ismeri a unit és integrációs teszt eszközöket és technikákat, azok korlátait, és a problémák megoldásait</li> <li>● Ismeri és használja a RESTful alapú API szabványokat (OpenAPI), valamint az azon alapuló API-tervezés és dokumentáció-generálás folyamatát és eszközeit (Swagger Editor, Swagger UI)</li> <li>● Ismeri és használja az autentikáció és a hozzáférés-hitelesítés alapvető megoldásait (API credentials)</li> <li>● Ismer egy konténerizációs technológiát és hogy hogyan kell konténereket létrehozni és kezelni</li> </ul>

## 2. A képzésbe való bekapcsolódás és részvétel feltételei

2.1.	Iskolai végzettség	<b>Középfokú végzettség</b>
2.2.	Szakmai végzettség	-
2.3.	Szakmai gyakorlat	-
2.4.	Egészségügyi alkalmasság	-
2.5.	Előzetesen elvárt ismeretek	06134004 Junior frontend fejlesztő szakképesítésben meghatározott szakmai kompetenciák megléte a végzettségről szóló tanúsítvánnyal, vagy ennek hiányában, a képző által szervezett előzetes szintfelmérő vizsga sikeres teljesítésével bizonyítottan.
2.6.	Egyéb feltételek	Minden résztvevővel az Fktv. (a felnőttképzésről szóló 2013. évi LXXVII. törvény és végrehajtási rendelete (a 11/2020. (II. 7.) Korm. rendelet) által meghatározott tartalmú felnőttképzési szerződést kell kötni. A felnőttképzési szerződésben feltüntetésre kerül az esetleges előzetes tudásmérés eredménye, továbbá az esetleges előzetesen megszerzett tudás beszámítása.

## 3. Tervezett képzési idő

3.1.	A képzés óraszám:	<b>320 óra</b>
3.2.	Megengedett hiányzás mértéke:	<b>A konzultációs óra 20 százaléka (kontaktóra esetén)</b> <i>(A felnőttképzési szerződést azonnali hatállyal felmondható, ha a képzésben részt vevő személy a kontaktórákról a képzési programban meghatározott időnél többet mulasztott.)</i>

## 4. Tananyagegységek

A képzés tananyag egységeinek megnevezése:	Óraszám:
<b>Szoftver Architektúrák</b>	<b>40</b>
Node JS alapok	40
Tesztelés	40
<b>NoSQL adatbázisok – MongoDB</b>	<b>40</b>
Node JS API fejlesztés	80
Angular haladó	40
<b>Konténerizáció - Docker</b>	<b>40</b>

### 4.1. Tananyagegység

4.1.1.	Megnevezése:	<b>Szoftver Architektúrák</b>
4.1.2.	Célja:	A tananyag célja, hogy a résztvevő átfogó képet kapjon a többretegű alkalmazások architektúrájáról, a webes alkalmazások és a REST API alkalmazások felépítéséről. A résztvevő a modul keretében megismerkedik a NoSQL adatbázisok elméleti alapjaival is.
4.1.3.	Megvalósítása során alkalmazott munkaformák:	Frontális oktatás, mentor által felügyelt egyéni- és csoportmunka, távoktatás
4.1.4.	Megvalósítása során alkalmazott képzési módszerek:	Előadás, magyarázat, szemléltetés, gyakorlat, megbeszélés, együttes és önálló tananyag feldolgozás, önálló tanulás
4.1.5.	Óraszám:	40 óra

4.1.6.	Beszámítható óraszám <sup>1</sup> :	40 óra		
4.1.7.	A programkövetelményben meghatározott, a tananyagegység keretében elsajátítandó szakmai követelmények			
S.sz.:	Készségek, képességek:	Ismeretek:	Elvárt viselkedésmódok, attitűdök:	Önállóság és felelősség mértéke:
1.	API kiszolgáló architektúrát tervez és dokumentál.	Ismeri és használja a RESTful alapú API szabványokat (OpenAPI), valamint az azon alapuló API-tervezés és dokumentáció-generálás folyamatát és eszközeit (Swagger Editor, Swagger UI).	Törekszik a szabványos API architektúrában gondolkodásra és az API dokumentálására.	Önállóan tervez meg és dokumentál egyszerűbb API-t.
Megtanítandó és elsajátítandó témakör(ök), kapcsolódó foglalkozás(ok) megnevezése, (óra)száma és tartalmi elemei: (az egyes foglalkozások számát óraszámban jelöljük)				
<b>Témakör:</b> Szoftver Architektúrák ismerete (40 óra)				
1. Önálló tanulás mentori támogatással és konzultáció szoftver architektúrákról (40 óra)				
<ul style="list-style-type: none"> <li>• 1.1 Architektúra fogalma</li> <li>• 1.2 Webes alkalmazás</li> <li>• 1.3 Többretegű alkalmazások</li> <li>• 1.4 Swagger Editor</li> <li>• 1.5 Swagger UI</li> <li>• 1.6 REST szolgáltatások</li> </ul>				

## 4.2. Tananyagegység

4.2.1.	Megnevezése:	<b>Node JS alapok</b>		
4.2.2.	Célja:	A modul célja, hogy a résztvevő elsajátítsa a Node JS alapjait, melynek segítségével a későbbiekben önállóan fejleszthet API kiszolgáló programot.		
4.2.3.	Megvalósítása során alkalmazott munkaformák:	Frontális oktatás, mentor által felügyelt egyéni- és csoportmunka, távoktatás		
4.2.4.	Megvalósítása során alkalmazott képzési módszerek:	Előadás, magyarázat, szemléltetés, gyakorlat, megbeszélés, együttes és önálló tananyag feldolgozás, önálló tanulás		
4.2.5.	Óraszám	40 óra		
4.2.6.	Beszámítható óraszám <sup>2</sup> :	40 óra		
4.2.7.	A programkövetelményben meghatározott, a tananyagegység keretében elsajátítandó szakmai követelmények			
S.sz.:	Készségek, képességek:	Ismeretek:	Elvárt viselkedésmódok, attitűdök:	Önállóság és felelősség mértéke:
1.	API kiszolgáló programot fejleszt.	Ismer legalább egy backend-oldali technológiát, aminek segítségével képes	Motivált egy API-vegpontokat kiszolgáló hatékony alkalmazás fejlesztésére.	Önállóan ír API kiszolgáló programot.

<sup>1</sup> Kontaktórától eltérő munkaforma alkalmazása esetén, ha az a képzés óraszámába beszámítható

<sup>2</sup> Kontaktórától eltérő munkaforma alkalmazása esetén, ha az a képzés óraszámába beszámítható

	API-végpontokat fejleszteni.		
Megtanítandó és elsajátítandó témakör(ök), kapcsolódó foglalkozás(ok) megnevezése, (óra)száma és tartalmi elemei: (az egyes foglalkozások számát óraszámokban jelöljük)			
<b>Témakör: Node JS alapok (40 óra)</b>			
1. Önálló tanulás mentori támogatással és konzultáció a Node JS alapokról. (40 óra)			
<ul style="list-style-type: none"> <li>• 1.1 Az npm csomagkezelő, a package.json file felépítése</li> <li>• 1.2 A NodeJS modulrendszere - commonjs</li> <li>• 1.3 Beépített modulok</li> <li>• 1.4 Szinkron vs aszinkron file műveletek</li> <li>• 1.5 Callback vs promise vs async await szintaxis</li> <li>• 1.6 Streamek – adatfolyamok, streamek típusai</li> <li>• 1.7 Eseménykezelés</li> <li>• 1.8 Clean Code ismeretek</li> </ul>			

### 4.3. Tananyagegység

4.3.1.	Megnevezése:	<b>Tesztelés</b>		
4.3.2.	Célja:	A modul célja, hogy megismertesse a résztvevőt a fejlesztett alkalmazás tesztelési lehetőségeiről, mint a unit és az integrációs tesztek. Megismeri ezen technikák korlátait és az esetlegesen felmerülő problémák megoldásait. A résztvevő a modul elvégzését követően képes önállóan választani a tesztelési technikák között, teszteseteket ír és futtat.		
4.3.3.	Megvalósítása során alkalmazott munkaformák:	Frontális oktatás, mentor által felügyelt egyéni- és csoportmunka, távoktatás		
4.3.4.	Megvalósítása során alkalmazott képzési módszerek:	Előadás, magyarázat, szemléltetés, gyakorlat, megbeszélés, együttes és önálló tananyag feldolgozás, önálló tanulás		
4.3.5.	Összes óraszám:	40 óra		
4.3.6.	Beszámítható órászáma <sup>3</sup> :	40 óra		
4.3.7.	A programkövetelményben meghatározott, a tananyagegység keretében elsajátítandó szakmai követelmények			
S.sz.:	Készségek, képességek:	Ismeretek:	Elvárt viselkedésmódok, attitűdök:	Önállóság és felelősség mértéke:
1.	Az alkalmazáshoz, melyet fejleszt, unit és integrációs tesztek írását és futtatását.	Ismeri a unit és integrációs teszt eszközöket és technikákat, azok korlátait és a problémák megoldásait.	Törekszik a hibamentes kód írására, ehhez tesztekkel lefedni a teljes általa készített kódot.	Önállóan választ teszt eszközöket, teszteseteket ír és futtat melyekkel a funkcionális helyesség biztosítható, és automatizáltan újra tesztelhető.
Megtanítandó és elsajátítandó témakör(ök), kapcsolódó foglalkozás(ok) megnevezése, (óra)száma és tartalmi elemei: (az egyes foglalkozások számát óraszámokban jelöljük)				
<b>Témakörök: Tesztelés (40 óra)</b>				

<sup>3</sup> Kontaktórától eltérő munkaforma alkalmazása esetén, ha az a képzés órázámba beszámítható

	<p>1. Önálló tanulás mentori támogatással és konzultáció a tesztelési lehetőségek használatáról és alkalmazási lehetőségeiről. (40 óra)</p> <ul style="list-style-type: none"> <li>• 1.1 Unit tesztek írása</li> <li>• 1.2 integrált tesztek fogalma, és írása</li> <li>• 1.3 end to end tesztek készítése</li> <li>• 1.4 Tesztkörnyezet telepítése és futtatása (JEST)</li> <li>• 1.5 Tesztek kiértékelése, coverage adatok generálása és report készítés</li> </ul>
--	---

#### 4.4. Tananyagegység

4.4.1.	Megnevezése:	<b>NoSQL adatbázisok – MongoDB</b>		
4.4.2.	Célja:	A tananyag célja, hogy a résztvevő megismerje az adatbázishoz tartozó alapelveket, CRUD fogalmakat. Képes legyen alap adatbázis lekérdezéseket készíteni. Egyszerű lekérdezéseket tudjon megfogalmazni, táblákat létrehozni, adatokat beszúrni, módosítani, törölni.		
4.4.3.	Megvalósítása során alkalmazott munkaformák:	Frontális oktatás, mentor által felügyelt egyéni- és csoportmunka, távoktatás		
4.4.4.	Megvalósítása során alkalmazott képzési módszerek:	Előadás, magyarázat, szemléltetés, gyakorlat, megbeszélés, együttes és önálló tananyag feldolgozás, önálló tanulás		
4.5.5.	Összes óraszám:	40 óra		
4.4.6.	Beszámítható órászáma <sup>4</sup> :	40 óra		
4.4.7.	<b>A programkövetelményben meghatározott, a tananyagegység keretében elsajátítandó szakmai követelmények</b>			
S.sz.:	Készségek, képességek:	Ismeretek:	Elvárt viselkedésmódok, attitűdök:	Önállóság és felelősség mértéke:
	NoSQL adatbázist tervez, létrehoz és dokumentál.	Ismer egy NoSQL adatbázist, és egy ehhez kapcsolódó ODM-et, melyet az API fejlesztés során adatok tárolására felhasznál.	Törekszik az adatbázisok biztonságának, konzisztenciájának megőrzésére.	Önállóan készít NoSQL adatbázist használó API-t, mely során egy ODM adta funkcionalitást is felhasznál.
1.	Megtanítandó és elsajátítandó témakör(ök), kapcsolódó foglalkozás(ok) megnevezése, (óra)száma és tartalmi elemei: (az egyes foglalkozások számát órászámban jelöljük)			
	<b>Témakör:</b> NoSql adatbázisok – MongoDB (40 óra)			
	<p>1. Önálló tanulás mentori támogatással, konzultáció és gyakorlat NoSQL adatbázisokról (40 óra):</p> <ul style="list-style-type: none"> <li>• 1.1 NoSql alapfogalmai, főbb parancsai: use, select, show, insert, drop, createCollection, insert, update, find, remove, sort, limit</li> <li>• 1.2 Adatok tárolása és szerkezetük a NoSql adatbázisokban</li> <li>• 1.3 Lekérdezések írása NoSql-ben</li> </ul>			

<sup>4</sup> Kontaktórától eltérő munkaforma alkalmazása esetén, ha az a képzés órászámába beszámítható

#### 4.5. Tananyagegység

4.5.1.	Megnevezése:	<b>Node JS API fejlesztés</b>		
4.5.2.	Célja:	A tananyag célja, hogy a résztvevő megismerjen egy backend oldali technológiát, és a modul végére képes legyen önállóan megírni egy egyszerű API kiszolgáló programot.		
4.5.3.	Megvalósítása során alkalmazott munkaformák:	Frontális oktatás, mentor által felügyelt egyéni- és csoportmunka, távoktatás		
4.5.4.	Megvalósítása során alkalmazott képzési módszerek:	Előadás, magyarázat, szemléltetés, gyakorlat, megbeszélés, együttes és önálló tananyag feldolgozás, önálló tanulás		
4.5.5.	Összes óraszám:	80 óra		
4.5.5.	Beszámítható órászáma <sup>5</sup> :	80 óra		
4.5.6.	A programkövetelményben meghatározott, a tananyagegység keretében elsajátítandó szakmai követelmények			
S.sz.:	Készségek, képességek:	Ismeretek:	Elvárt viselkedésmódok, attitűdök:	Önállóság és felelősség mértéke:
	API kiszolgáló programot fejleszt.	Ismer legalább egy backend-oldali technológiát, aminek segítségével képes API-végpontokat fejleszteni.	Motivált egy API-végpontokat kiszolgáló hatékony alkalmazás fejlesztésére.	Önállóan ír API kiszolgáló programot.
	Kapcsolódó foglalkozás megnevezése, megtanítandó és elsajátítandó témaköre és tartalmi elemei:			
	<b>Témakör:</b> Node JS API fejlesztés (80 óra)			
1.	1. Önálló tanulás mentori támogatással, konzultáció és gyakorlat <ul style="list-style-type: none"> <li>• 1.1 Az Express http keretrendszer</li> <li>• 1.2 Routing</li> <li>• 1.3 Hibakezelés</li> <li>• 1.4 Logolás</li> <li>• 1.5 Static fileok kiszolgálása</li> <li>• 1.6 Autentikáció</li> <li>• 1.7 JWT autentikáció és autorizáció</li> <li>• 1.8 NOSQL adatbázisok</li> <li>• 1.9 Komplet API készítése</li> <li>• 1.10 Clean code elvek használata a gyakorlatban</li> </ul>			

#### 4.6. Tananyagegység

4.6.1.	Megnevezése:	<b>Angular haladó</b>
4.6.2.	Célja:	A tananyagegység célja, hogy a résztvevő megismerkedjen az alkalmazás különböző biztonsági szintjeivel. Képes legyen használni az autentikáció és a hozzáférés-hitelesítés alapvető megoldásaival. A modul elvégzése után a résztvevő önállóan hozzon létre hozzáférés-hitelesített API kiszolgáló programot. A résztvevő önállóan elkészít egy frontend-oldali alkalmazást, amely egy meglévő REST API-t használ, a legjobb gyakorlatoknak és a tiszta kód elveinek megfelelően.
4.6.3.	Megvalósítása során alkalmazott munkaformák:	Frontális oktatás, mentor által felügyelt egyéni- és csoportmunka, távoktatás

<sup>5</sup> Kontaktóráról eltérő munkaforma alkalmazása esetén, ha az a képzés órászámába beszámítható



4.6.4.	Megvalósítása során alkalmazott képzési módszerek:	Előadás, magyarázat, szemléltetés, gyakorlat, megbeszélés, együttes és önálló tananyag feldolgozás, önálló tanulás		
4.6.5.	Összes óraszám:	40 óra		
4.6.6.	Beszámítható órászáma <sup>6</sup> :	40 óra		
4.6.7.	A programkövetelményben meghatározott, a tananyagegység keretében elsajátítandó szakmai követelmények			
S.sz.:	Készségek, képességek:	Ismeretek:	Elvárt viselkedésmódok, attitűdök:	Önállóság és felelősség mértéke:
1.	Felméri a szükséges biztonsági szinteket, és használja az autentikáció és a hozzáférés-hitelesítés alapvető megoldásait (API credentials)	Ismeri az autentikáció és a hozzáférés-hitelesítés alapvető megoldásait (API credentials).	Törekszik védett, biztonságos API kiszolgáló alkalmazást készíteni.	Önállóan hoz létre hozzáférés-hitelesített API kiszolgáló programot.
	Frontend-oldali alkalmazásokat készít, amelyek létező és jól dokumentált REST API-kat használnak.	Ismeri a REST webszolgáltatás-architektúrát, és annak kliensoldali használatát.	Törekszik a legjobb gyakorlatoknak és a tiszta kód elveinek megfelelő kliensoldali kód írására	Önállóan el tud készíteni egy REST API-t használó frontend-oldali alkalmazást.
Megtanítandó és elsajátítandó témakör(ök), kapcsolódó foglalkozás(ok) megnevezése, (óra)száma és tartalmi elemei: (az egyes foglalkozások számát óraszámban jelöljük)				
<b>Témakör: Angular haladó technikák ( 40 óra)</b>				
<p>1. Konzultáció és gyakorlat az Angular haladó technikákról, azok működéséről és gyakorlati használatáról (40 óra)</p> <ul style="list-style-type: none"> <li>● 1.1 Cors policy</li> <li>● 1.2 Basic Autentikáció Angularban</li> <li>● 1.3 Autorizáció angularban</li> <li>● 1.4 Json webtoken</li> <li>● 1.5 Hibakezelés, hibakódok</li> <li>● 1.6 REST API integrációja az alkalmazásba</li> </ul>				

#### 4.7. Tananyagegység

4.7.1.	Megnevezése:	<b>Konténerizáció - Docker</b>
4.7.2.	Célja:	A tananyagegység célja, hogy a résztvevő elsajátítsa a egy konténerizációs technológia alapjait, képes legyen konténereket létrehozni és kezelni. Megismeri a CI/CD folyamatát.
4.7.3.	Megvalósítása során alkalmazott munkaformák:	Frontális oktatás, mentor által felügyelt egyéni- és csoportmunka, távoktatás
4.7.4.	Megvalósítása során alkalmazott képzési módszerek:	Előadás, magyarázat, szemléltetés, gyakorlat, megbeszélés, együttes és önálló tananyag feldolgozás, önálló tanulás

<sup>6</sup> Kontaktórától eltérő munkaforma alkalmazása esetén, ha az a képzés órászámába beszámítható

4.7.5.	Összes óraszám	40 óra		
4.7.6.	Beszámítható óraszáma <sup>7</sup> :	40 óra		
4.7.7.	A programkövetelményben meghatározott, a tananyagegység keretében elsajátítandó szakmai követelmények			
S.sz.:	Készségek, képességek:	Ismeretek:	Elvart viselkedésmódok, attitűdök:	Önállóság és felelősség mértéke:
1.	Alkalmazásokat és az adatbázist konténerizált környezetben futtat.	Ismer egy konténerizációs technológiát és hogy hogyan kell konténereket létrehozni és kezelni	Törekszik arra, hogy az alkalmazásait úgy írja meg, hogy azok konténerizált környezetben is futtathatóak legyenek.	Önállóan hoz tervezési döntéseket az alkalmazás konténerizált környezetben történő futtatásához.
Megtanítandó és elsajátítandó témakör(ök), kapcsolódó foglalkozás(ok) megnevezése, (óra)száma és tartalmi elemei: (az egyes foglalkozások számát óraszámban jelöljük)				
- <b>Témakör:</b> Konténerizáció (40 óra)				
1. Önálló tanulás mentori támogatással és konzultáció (40 óra)				
<ul style="list-style-type: none"> <li>• 1.1 A NodeJS alkalmazás konténerizálása</li> <li>• 1.2 CI/CD, deploy folyamata</li> <li>• 1.3 Docker Compose</li> </ul>				

## 5. Csoportlétszám

5.1.	Maximális csoportlétszám:	40 fő (zárt rendszerű távoktatási rendszer esetén nem releváns)
------	---------------------------	---

## 6. A képzésben részt vevő teljesítményét értékelő rendszer leírása

Szakképzés esetén: (Szkt. végrehajtásáról szóló 12/2020. (II. 7.) Korm. rendelet 20. § (1) bekezdés b) pont.) A szakmai képzés képzési programja tartalmazza a képzésben részt vevő személy tanulmányi munkájának írásban, szóban vagy gyakorlatban történő ellenőrzési és értékelési módjait, diagnosztikus, szummatív, fejlesztő formáit is.

6.1.	Előzetes tudásmérés (diagnosztikus értékelés): Nincs, résztvevő kérésére biztosított
6.2.	<p>Képzés közbeni (fejlesztő) értékelés:</p> <p>A fejlesztő értékelés szerepe, hogy a képzésben résztvevők fejlődését támogassa, a tanulási igényeket pontosítsa, az oktatók tanulásszervezési feladatait segítse.</p> <p>A képzés közbeni fejlesztő értékelés módjai lehetnek: Visszakérdezés, Gyakorlati feladatmegoldás, Képzésben résztvevő visszajelzései, Beszélgetés, Szintfelmérő teszt</p> <p>A fenti fejlesztő értékeléshez <b>nem tartozik minősítés</b>, a tanulási és tanítási folyamatokat szolgálja.</p> <p>A képzés során <b>egy részvizsgán (félidős)</b> kell megfelelnie a résztvevőnek.</p> <p>A részvizsga (az addig leadott 160 óra anyagából) két részből áll:</p> <ul style="list-style-type: none"> <li>- A teszt 40 db feleletválasztásos tesztkérdést tartalmaz. A feleletválasztásos tesztkérdések egyetlen helyes válaszlehetőséget tartalmaznak. A vizsgatevékenység végrehajtására rendelkezésre álló időtartam 60 perc. A helyes válasz 0,5 pontot ér, a helytelen válasz 0 pontot, így maximálisan 20 pontot lehet elérni. A sikeres vizsgához (megfelelt minősítéshez) 10,5 pont megszerzése szükséges.</li> </ul>

<sup>7</sup> Kontaktórától eltérő munkaforma alkalmazása esetén, ha az a képzés óraszámába beszámítható

	<ul style="list-style-type: none"><li>- Gyakorlati vizsga (összesen 80 pont), 5 részfeladatból áll:<ul style="list-style-type: none"><li>- 0 - 5 pont: az adott projekt nem fordul le, többségében nem futnak le a tesztesetek, az elvárt funkcionalitást nem valósítja meg.</li><li>- 6 - 11 pont: a projekt lefordul, a tesztesetek legtöbbje lefut, ezek nagyrészt funkcionálisan is helyesek, és a clean code elvek nagyrészt betartásra kerültek.</li><li>- 12 - 16 pont: ha a projekt lefordul, a tesztesetek lefutnak, funkcionálisan helyesek, és csak apróbb funkcionális vagy clean code hibák szerepelnek a megoldásban.</li></ul></li></ul> <p>A vizsgatevékenység akkor eredményes, ha a vizsgázó a megszerezhető összes pontszám legalább 51 %-át elérte.</p> <p>Sikertelen részvizsga esetén a vizsga (2 alkalommal) megismételhető.</p>
6.3.	<p><b>Részvevő záró (szummatív) értékelése:</b></p> <p><b>A képzés záróvizsgával zárul.</b> A záróvizsga a képzés végén kerül megtartásra, és két részből áll. Tartalmaz egy tesztkérdéssort és egy projektfeladatot.</p> <p><b>Tesztkérdéssor</b></p> <p>A tesztkérdéssor 40 kérdésből áll, mellyel összesítve 20 pontot lehet elérni. A feleletválasztásos tesztkérdéseket úgy kell kialakítani, hogy egyetlen helyes válaszlehetőség legyen lehetséges. A vizsgatevékenység végrehajtására rendelkezésre álló időtartam 60 perc. A vizsgatevékenység aránya a teljes záróvizsgán belül: 20 %. A vizsgatevékenység akkor eredményes, ha a vizsgázó a megszerezhető összpontszám legalább 51%-át elérte.</p> <p><b>Gyakorlati vizsga</b></p> <p>A vizsgatevékenység végrehajtására rendelkezésre álló időtartam 210 perc. Tartalma: A tananyagegységek tartalma szerinti ismeretek, készségek, képességek elsajátításának gyakorlati feladatokba ágyazott bemutatása, alkalmazása, programozási feladat(ok) végrehajtása. A vizsgázó a feladatokat szöveges formátumban kapja meg. A feladatléírás tartalmazza a feladat szöveges leírását. A vizsga tevékenység akkor eredményes, ha a vizsgázó a megszerezhető összpontszám legalább 51 %-át elérte.</p> <p>A záróvizsgán megszerezhető minősítések és a megszerezhető minősítésekhez tartozó követelményszintek:</p> <ul style="list-style-type: none"><li>• Megfelelt: a záróvizsga mindkét vizsgatevékenységének eredményes (legalább 51%-os) teljesítése.</li><li>• Nem felelt meg: a záróvizsga egyik vagy mindkét vizsgatevékenységének eredménytelen (nem felelt meg) teljesítése.</li></ul>

## 7. A képzés, a képzés egyes tananyagegységeinek elvégzéséről szóló igazolás kiadásának feltételei

7.1.	A képzés elvégzéséről szóló igazolás megnevezése:	<b>TANÚSÍTVÁNY</b> 2013. évi LXXVII. törvény 13/B. § 11/2020. (II. 7.) Korm. rendelet 22. § (1)
7.2.	A képzés elvégzéséről szóló igazolás kiadásának feltételei:	A képzés elvégzéséről szóló tanúsítvány kiadásának feltétele a záróvizsgán „Megfelelt” minősítés megszerzése.
7.3.	A képzés egyes tananyagegységeinek elvégzéséről szóló igazolás kiadásának feltételei:	A tananyagegységek elvégzéséről nem kerül kiadásra külön igazolás.

## 8. A képzési program végrehajtásához szükséges feltételek

8.1.	Személyi feltételek:	Felsőfokú iskolai végzettség és legalább 2 év a képzési program tananyagához illeszkedő szakirányú szakmai tapasztalat, vagy középfokú iskolai végzettség és legalább 3 év a képzési program tananyagához illeszkedő szakirányú szakmai tapasztalat.
8.2.	Személyi feltételek biztosításának módja:	Az oktatót képző intézmény foglalkoztatja munkaszerződéssel, megbízási szerződéssel vagy az oktató alkalmazását bizonyító más szerződéssel.
8.3.	Tárgyi feltételek:	Kontaktórás képzés esetén résztvevői létszámnak megfelelő: <ul style="list-style-type: none"> <li>- asztal, szék</li> <li>- számítógép (javasolt: Core i5, 8GB RAM, 40GB háttértár), internetkapcsolat</li> </ul> Távoktatás esetén: <ul style="list-style-type: none"> <li>- személyi számítógép vagy laptop, lokális rendszergazdai (adminisztrátori) jogosultsággal (javasolt: minimum Core i5 vagy ennek megfelelő AMD processzor, minimum 8 GB RAM és legalább 40 GB szabad tárhely),</li> <li>- Microsoft Windows 10 vagy Linux operációs rendszer,</li> <li>- mikrofon (beépített, vagy külső)</li> <li>- javasolt webkamera,</li> <li>- valamint szélessávú (legalább 5 Mbit/sec szabad sáv szélességű) internetelérés.</li> </ul>
8.4.	Tárgyi feltételek biztosításának módja:	Kontaktórás képzés esetén: a képzési helyszínek, oktatóterem meglétét felnőttképző tulajdonjog, használati jog vagy bérleti jogviszony alapján biztosítja; a tárgyi eszközök meglétét képzésben résztvevő saját eszközeként, vagy a felnőttképző tulajdonjog, használati jog vagy bérleti jogviszony alapján biztosítja.  Távoktatás esetén a tárgyi eszközök meglétét képzésben résztvevő saját eszközeként biztosítja.
8.5.	A képzéshez kapcsolódó egyéb speciális feltételek:	-
8.6.	A képzéshez kapcsolódó egyéb speciális feltételek biztosításának módja:	-

## 9. Képesítő vizsga

A képesítő vizsgát nem a képző intézmény szervezi és bonyolítja.

A szakmai képzéshez kapcsolódóan megszerezhető szakképesítés megszerzésére irányuló képesítő vizsgát a nemzeti akkreditálásról szóló törvény szerinti akkreditáló szerv által személytanúsító szervezetként **akkreditált vizsgaközpont szervezhet.**

A szakmai képzéshez kapcsolódóan megszerzett képesítő bizonyítvány államilag elismert, önálló végzettségi szintet nem biztosító szakképesítést tanúsít.


A képesítő vizsga megszervezéséhez szükséges feltételek és a képesítő vizsga vizsgatevékenységeinek részletes leírása a <https://szakkesites.ikk.hu/> weblapon érhető el a programkövetelmények menüpont alatt.

### **A képesítő vizsgára bocsátás feltétele:**

**A szakmai képzés követelményeinek teljesítéséről (7.1. pont) a képző intézmény által a felnőttképzési adatszolgáltatási rendszerben kiállított tanúsítvány.**

Egyéb feltételek: -----

### 10. Az előzetes minősítés ténye

Szakértő nyilatkozata:	A képzési program előzetes minősítése megtörtént.
Az előzetes minősítés helye:	Budapest
Az előzetes minősítés időpontja:	2021.06.04.
Az előzetes minősítést végző felnőttképzési szakértő neve:	Pádár Tivadar
Az előzetes minősítést végző felnőttképzési szakértő nyilvántartási száma:	FSZ/2020/000057
Felnőttképzési szakértő aláírása: (papíralapú képzési program esetében)	
Felnőttképző képviselőjére jogosult személy aláírása:	