

## 2. feladat

Összesen: 40 pont

## Weblapkészítés

A szolgáltatásorientált architektúra (SOA)<sup>1</sup>

A következő feladatban egy magyar nyelvű weblapot fog készíteni, amely a szolgáltatásorientált architektúrát mutatja be röviden. A feladat megoldása során a következő állományokat kell felhasználnia: `index.html`, `styles.css`, `SOA_Back.jpg`, `SOA_metamodell.png`, `forras.txt`. Nagy felbontású, színes mintát a kész weboldalról a  `minta.jpg` állományban talál, melyet tilos a megoldásában felhasználni!

1. Nyissa meg az `index.html` állományt! Helyezzen el HTML5-ös dokumentumtípus definíciót az első sorba!
2. Állítsa be az oldal kódolását UTF-8-ra, a nyelvet magyarra!
3. Az `index.html` állományba helyezzen el hivatkozást a `styles.css` stíluslapra!
4. A böngésző címsorában megjelenő cím „SOA” legyen!
5. A tartalom azonosítójú keretben lévő szöveget sorolja bekezdésekbe a minta szerint, és alakítsa ki a h1 és h2 címsorszintű címeket! A „Legfontosabb szabályok” és a „Minták” címsorokat lássa el megfelelő méretű felső margóval úgy, hogy a mintához hasonló térközt kapjon! Az utóbbi feladatot az `index.html` állományban Bootstrap osztály felhasználásával oldja meg!
6. Alakítsa ki a számozott és számozatlan (felsorolt) listákat a minta szerint!
7. Helyezze el a `SOA_metamodell.png` képet az `imageright` azonosítójú keret első elemeként! A kép azonosítója SOA legyen! Ha a kép fölé visszük az egeret, vagy a kép valamiért nem jeleníthető meg, akkor mindkét esetben „SOA metamodell” felirat jelenjen meg!
8. Készítsen képaláírást a minta szerint! Ehhez a kép alá hozzon létre egy bekezdést, melyet soroljon a `kepszoveg` osztályba! A képaláírás szövege „SOA metamodell a The Linthicum Group szerint, 2007” legyen!
9. A szövegben a SOA rövidítést emelje ki #F05033 kódú betűszínnel és félkövér betűstílussal! A megoldáshoz használjon `<span>` taget és megfelelő típusú szelektort! Oldja meg, hogy ha a kurzort a kiemelt név fölé visszük, akkor színe váltszon kékre (blue)! A CSS beállításokat a `styles.css` állományban helyezze el új szelektor létrehozásával!
10. Alakítsa ki a táblázatot a minta szerint! A táblázathoz rendelje a `table` Bootstrap osztályt! Az első oszlopban lévő cellák fejléccella-típusúak legyenek! A táblázat elhelyezésére és formázására később talál információt.
11. Készítsen Bootstrap 4 rácsot a „Legfontosabb szabványokat” bemutató rész és a mellette látható táblázat igazításához! Extra nagy (xl) méretű kijelzőknél egymás mellett, annál kisebb méretű kijelzők esetén egymás alatt jelenjenek meg!
12. Alakítsa ki „Szolgáltatásorientált architektúra (SOA)” címre a forrásra hivatkozó linket! Az URL címet a `forras.txt` állományban találja meg. Oldja meg, hogy a hivatkozás minden esetben új oldalon nyíljon meg!

<sup>1</sup> Forrás: [https://hu.wikipedia.org/wiki/Szolgáltatásorientált\\_architektúra](https://hu.wikipedia.org/wiki/Szolgáltatásorientált_architektúra)



A következő beállításokat a `styles.css` stíluslapon végezze új szelektorok létrehozásával vagy a meglévők bővítésével!

13. Állítsa be az oldal háttérének a `SOA_Back.jpg` állományt, ami ismétlés nélkül kerüljön az oldal közepére függőlegesen és vízszintesen is!
14. Az `imageright` azonosítójú elem lebegjen jobbra, bal és alsó margója 20 képpont legyen, és rendelkezzen 2 képpont széles, folyamatos, `#F05033` színkódú kerettel!
15. A `kepszoveg` osztályú bekezdések legyenek középre igazítva, állítson be félkövér és dőlt betűstílust, és oldja meg, hogy az ebbe az osztályba sorolt bekezdéseknek ne legyen elsősori behúzása!
16. A táblázat fejléccelláinak előtérshíne fehér, a háttérshíne `#F05033` színkódú legyen! Oldja meg, hogy az adatcellákban lévő tartalom sorkizárt legyen!
17. A linkek (hivatkozások) színe fekete legyen, ha az egérkurzor föléjük kerül, akkor színe ebben az állapotban `#F05033` színkódúra váltsa!

Minta:

### Szolgáltatásorientált architektúra (SOA)

A szolgáltatásorientált architektúra (angolul Service-Oriented Architecture, röviden **SOA**) egy programozási módszer, aminek lényege, hogy az egyes szolgáltatások egy hálózatban helyezkednek el, és egy protokoll által meghatározott módon kommunikálnak. Alapvető elve a független gyártóktól, termékektől, terjesztőktől és technológiáktól. A szolgáltatások különböző egységek, amik távolról is elérhetők egymástól függetlenül elérhetők, használhatók, frissíthetők, újra kombinálhatók a folyamatok folyamatos változásának megújulásának megfelelően.

Egy szolgáltatásnak a következőket kell teljesítenie:

1. Logikailag reprezentál egy üzleti aktivitást előre meghatározott kimenettel.
2. Önálló, független egység (self-contained).
3. Feketedoboz a kliens számára.
4. Tartalmazhat több részsolgáltatást.

Az 1970-es években a modularitásra megadott definíció szerint különböző szolgáltatások használhatók együtt, hogy egy nagy alkalmazás funkcionálisát nyújtsák. Azonban a szolgáltatásorientált architektúra célja nem az egyes alkalmazások tagolása, és az így létrejövő, részfunkcionalitásokat nyújtó modulok közötti kapcsolat biztosítása, hanem ellenkezőleg: minden szempontból önálló, saját funkcionálisával bíró, és eleve együttműködésre tervezett szolgáltatások összekapcsolásán keresztül megvalósítani az üzleti folyamatokat. A **SOA** a vállalat üzleti folyamataira jellemző - valós üzleti tevékenységeket tükröző - szolgáltatások tervezésén alapul és nem az alkalmazások technológiai célú szétbontásán.

SOA metamodel a The Linthicum Group szerint, 2007

### Legfontosabb szabványok

A **SOA**-t egy szabványos felszínen, a webszolgáltatás platformon valósítják meg. Ezt a Web szolgáltatás szabványt egyformán támogatja a .NET és a Java. A **SOA** által használt szolgáltatásorientált környezet legfontosabb szabványai a következők:

- SOAP, egy XML (WSDL) alapú kiterjesztett üzenet formátum (boríték).
- WSDL webszolgáltatás leíró nyelv.
- UDDI általános kereső és integrációs leírásokat tároló eszköz.
- Üzenetküldés, mint ActiveMQ, JMS, RabbitMQ.
- RESTful HTTP, a REST saját megkötés alapú architektúrális stílusával
- OPC-UA
- WCF a Microsofttól
- Apache Thrift
- SORCER

### Minták

<b>Szolgáltató</b>	Létrehozza a webszolgáltatást, és információt közvetít a névszolgáltatónak. Azt is leírja, hogy milyen szolgáltatást nyújt, mi a fontosabb: a könnyű elérhetőség vagy a biztonság, mennyire lehet igénybe venni és további információkat is adhat, amik bekerülnek a névszolgáltatóhoz.
<b>Névszolgáltató</b>	Egy névszolgáltató feladata az, hogy információt nyújtson az elérhető szolgáltatásokról. Megvalósítjuk döntésről függően lehetnek nyilvánosak vagy privátok, ez utóbbiakat nem mindenki veheti igénybe. Az UDDI egy korai kísérlet volt a webszolgáltatások felderítésére.
<b>Szolgáltatásigénylő</b>	Az általa igényelt szolgáltatás elérhetőségét névszolgáltatótól keresi, majd a kellő információk birtokában hozzákapszólódik a szolgáltatóhoz, és meghívja szolgáltatásait. Egy résztvevő lehet egyszer szolgáltató, másszor szolgáltatásigénylő. A szolgáltatók is névszolgáltatótól keresik a számukra fontos adatokat.