**Vizsgaremek készítése**

**Az alkalmazás:**

Adminisztrációs rendszer legyen (sablon használható).

**Követelmények:**

* MongoDB alapú, NoSQL - legalább 5 féle entitás legyen (pl.: termék, rendelés, vásárló, kiszállítás, számla).
* NodeJS API, saját maga által leprogramozott API szolgálja ki a frontendet - legalább 10 különböző végpont legyen (pl.: api/user, api/product, api/order …).
* Frontend Angular/React alapú - legalább 5 külön oldal legyen, model-service-component architektúra, Bootstrap/Material vagy egyéb sablon használata engedélyezett, reszponzív legyen. A felület bizonyos oldalai csak belépés után legyenek elérhetőek (JWT-autentikáció).
* Clean kód elveit kövesse az alkalmazás összes eleme.
* Minden API útvonalhoz legalább egy tesztet kell írni. Legalább 1-1 Unit/Integrációs teszt.
* Swagger alapú dokumentáció az API-hoz.
* Markdown dokumentáció a repository-ban, tartalmazza az alkalmazás telepítését, konfigurálását, célját.
* Dockerizálva legyen a kész alkalmazás, konténerből legyen futtatható.

**Ötletek, választható témák:**

1. oltópont,
2. webshop,
3. idősek otthona (gyógyszerek, ételek, időpontok),
4. vállalatirányítási rendszer (számlák, rendelések),
5. munkaidő-nyilvántartó rendszer,
6. munkaerő-nyilvántartó rendszer,
7. kutatások nyilvántartása,
8. flottakezelő rendszer,
9. autókereskedés,
10. filmes adatbázis,
11. állatkereskedés,
12. menhelyi nyilvántartó,
13. eszközleltár,
14. biztosítási rendszer (szerződések, káresemény, biztosítási összeg, biztosítás fajtája),
15. foglalási rendszer,
16. utazási iroda,
17. építőipari projekt rendszere,
18. könyvtári nyilvántartó,
19. egyéni költségeket nyilvántartó rendszer,
20. valutaváltó,
21. vonatjegy applikáció,
22. ételrendelő applikáció,
23. szolgáltatásrendelő (fodrászat, időpontkezelés, szolgáltatások),
24. mesterember app (kit, mikor lehet hívni),
25. tanulónyilvántartó rendszer (NEPTUN),
26. kedvezményes kuponokat gyűjtő app (milyen termék, milyen kedvezmény),
27. ticketing (hibabejelentő rendszer),
28. online kurzusok (e-learning menedzsment rendszer, LMS),
29. telefonflotta nyilvántartása,
30. elektromos művek ügyfélnyilvántartója,
31. nyelvtanár-nyilvántartó,
32. ételkiszállítás/ételfutár app (Teletál, Netpincér),
33. konferencia-nyilvántartó,
34. receptgyűjtemény.

**Haladási terv**

1. hét (május 24-29.): témaválasztás, user story
2. hét (május 30-június 5.): Angular – frontend/adatbázis-séma, végpontok
3. hét (június 6-12.): backend
4. hét (június 13-19.): backend
5. hét (június 20-26.): Swagger, Docker

Végső leadási határidő: június 27. 12:00