Formai követelmények (kötelező):

* Kód github repository-ban az oktatónak elküldve
* Kubernetes telepítő fájl(ok) a repository-ban
* Kong API gateway konfigurációs scriptek
* Npm install és swagger project start segítségével az API hiba nélkül elindul lokális gépen
* Minikube-ban a Kubernetes telepítővel + Kong API gateway konfiguráció után Swagger UI-al vagy Postmannel megszólítható a projekt (+jelezni, hogy a hallgató melyikkel tesztelte)
* Dokumentáció: Az alábbi táblázat megjegyzés oszloppal min. 1 mondatos cellákkal, amiben dokumentálva van, hogy az adott követelmény megvalósítása melyik fájl(ok) mely soraiban található.

Maximális pontszám: 60 pont

|  |  |
| --- | --- |
| OpenAPI elemek megvalósítása | Pontszám |
| * **Szabványos swagger raw endpoint** | 1 ? |
| * **JSON támogatás** | 1 ✔ |
| * Swagger request és response validation használata | 1 ✔ |
| * **GET, PUT, POST, DELETE végpontok** | 4 ✔ |
| * Újrahasznosítható objektum definíciók használata minden végponthoz | 2 ✔ |
| * Required mezők használata az objektum definíciókban és paraméterekben | 2 ✔ |
| * Query string paraméter használat lekérdezésnél | 1 ✔ |
| * Hibakezelés: Közös hiba definíció használata, minden endpoint által használva, controllerekben megvalósítva | 3 ✔ |
| * Hibakezelés: HTTP hibakódok használata különböző hiba esetekre (pl. hiányzó objektum, authentikációs hiba, hiányzó jogosultság, egyéb szerver hiba), controllerekben megvalósítva | 3 ✔ |
| * Tag-ek használata végpontok csoportosítására | 1 ✔ |
| * XML támogatás minimum 1 végpontra | 2 |
| API authentikáció - Swagger Security |  |
| * **Globális session alapú authentikáció minden endpointra (API Key)** | 5 ✔ |
| * **Login és Signup végpontok, authentikációs kivételekkel** | 2 ✔ |
| API üzleti logika - Controllers |  |
| * Adattárolás (in-memory vagy perzisztens) használata | 2 ✔ |
| * **A választott téma szerinti logika megvalósítása (számítás, nem csak CRUD)** | 6 ✔ |
| API Gateway használat - Kong |  |
| * **API Gateway használata reverse proxy-ként** | 3 |
| * API Key használata kliens azonosításra (mobil, web) | 4 |
| * Rate limit használata (globális) | 3 |
| * Rate limit használata (klienstől függő - Kong consumer) | 4 |
| * Dinamikus terhelés (load balancing) | 4 |
| Docker Compose |  |
| * **Működő Docker-compose deployment Kong-gal és OpenAPI-val (verzio min 3)** | 5 |
| * Verziózott Docker image-ek használata | 3 |
| * Health check definiálása az API-hoz | 2 |

Választható témák:

* Mobil tárca API
  + Különféle böngészések és vásárlások megvalósítása (példa: Simple by OTP)
* Navigációs API
  + Cél és forrás (szabad szöveges cím) megjelöléssel útvonal számítás
  + A sessionben tárolja az aktuális helyszínt és forrás megjelölés nélkül azt használja
* Angry Birds API
  + Egymás elleni játékként