

CLOUD ENGINEERING

4. forduló



A kategória támogatója: IBM

RENDELKEZÉSRE ÁLLÓ IDŐ:

10:00

Ismertető a feladathoz

Fontos információk

A forduló után a megoldások publikálásával együtt iránymutatásként elérhetőek lesznek a **helyezéssel kapcsolatos információk**, látni fogod, hogy a kategóriában a játékosok 20%, 40% vagy 60%-a közé tartozol-e épp.

Felhívjuk figyelmedet, hogy a következő, **5. fordulótól az egyes kategóriák csak a kijelölt napokon lesznek megoldhatóak 7-22 óra között**, érdemes letöltened a naptárat a [Kategóriáim](#) menüpontban.

Negyedik forduló

A negyedik lépés a teljesítmény optimalizálás megtervezése

Felhasznált idő: 02:03/10:00

Elért pontszám: 0/95

1. feladat 0/10 pont

Melyik alkalmazás nem skálázható horizontálisan?

Válasz

- ☒ Relációs adatbázis szerver
- ☐ NoSQL adatbázis szerver
- ☐ BigData szerver

Magyarázat

A tradicionális adatbázis szerverek kizárólag vertikálisan skálázhatók, egyre nagyobb kapacitású és teljesítményű szerver üzembe állításával.

2. feladat 0/10 pont

Az alkalmazások hogyan kezeljék a futás idejű logokat?

Válasz

- ☐ Ne készítsen logot
- ☒ Töltse be egy központosított log kezelő alkalmazásba
- ☐ Írja ki helyi lemezre
- ☐ Tartsa memóriában az utolsó 24 óra logját

Magyarázat

Az optimális megoldás egy központosított log kezelő alkalmazása. Ekkor egy közös helyen érhetjük el az összes alkalmazás logjait és jobb rálátásunk lesz a teljes informatikai rendszer állapotára.

3. feladat 0/10 pont

Melyik felhő alapú szolgáltatás jön garantált, beépített terhelés elosztóval (Workload balancer)?

Válasz

- ☐ SaaS (Software as a Service)
- ☐ PaaS (Platform as a Service)
- ☐ IaaS (Infrastructure as a Service)
- ☒ FaaS (Function as a Service)

Magyarázat

FaaS esetében az szolgáltató gondoskodik a kód lefordításáról, annak konténerizációjáról és elérhetővé tételéről. Ez magában foglalja a terhelés elosztását is.

4. feladat 0/15 pont

Ahhoz, hogy egy a felhasználóval interakcióban lévő web front-end alkalmazásunk virtuálisan végtelen terhelést is képes legyen lekezelni, milyen karakterisztikákkal kell rendelkeznie

Válaszok

- ☒ Terhelés elosztó használata (Workload Balancer)
- ☒ Alkalmazástól független adat csatornák (Message Queue) használata, hogy az alkalmazás és az adat független legyen egymástól
- ☐ Mesterséges intelligencia használata
- ☒ Micro services architektúra használata
- ☐ CI/CD folyamatok használata

Magyarázat

Egy kiugró terheléssel rendelkező, a felhasználóval interakcióban lévő honlap (aliexpress, amazon.com, etc) esetében horizontális skálázással (workload balancer) kell dolgoznunk. Az alkalmazásnak függetlenül kell kezelnie az adatot minden mástól és minden funkciót egyénileg skálázhatóvá kell tennie.

Frissítés: NoSQL adatbázis használata válaszlehetőség a versenyzői visszajelzések alapján szintén elfogadható válasz, de a kérdés értelmezésétől függően a helytelen válaszlehetőségként is magyarázható. A feladatíró visszajelzése alapján töröltük a választ.

5. feladat 0/10 pont

Mi a preferált mód sorrendje 2 alkalmazás közti adatcserének

Válasz

- ☐ Direkt adatbázis olvasás > API > File
- ☐ File > API > Direkt adatbázis olvasás
- ☐ File > Message Queue > API
- ☐ API > Direkt adatbázis olvasás > File

Magyarázat

Az iparági standard jelenleg az API hívások alkalmazása. Ezt követi a direkt adatbázis kapcsolat létrehozása majd legvégül a file alapú adatcsere.

6. feladat 0/15 pont

Új végfelhasználói igényként az egyik noSQL adatbázist használó alkalmazásra riportolási funkciót kell készítenünk az üzleti felhasználóknak. Milyen megoldás(ok) a preferáltak?

Válaszok

- ☐ Adjunk közvetlen hozzáférést az adatbázishoz a felhasználóknak, akik megírhatják a saját lekérdezéseiket
- ☐ Hozzunk létre egy csapatot akik a riportokat elkészítik és elküldik a végfelhasználóknak
- ☒ Adjunk egy riportolási / BI felületet, amin keresztül az adatbázis elérhető
- ☒ Előre definiáljunk lekérdezéseket, amiket a felhasználók futtathatnak

Magyarázat

A noSQL adatbázis riportolása nehézkes, mivel a lekérdezési felülete és típusa változó, nem egységesített, mint a hagyományos adatbázisok esetében (SQL). A jó megoldás ebben az esetben egy kezelőfelület biztosítása, amin keresztül előre megírt és optimalizált lekérdezéseket futtathatnak. Így biztosítható, hogy az adatbázist nem terheli le a riportolás olyannyira, hogy az az általános válaszdő rovására menjen.

7. feladat 0/15 pont

Tegyük fel, hogy egy felhőalkalmazásban több, koordinált módon működő feladat található. Ezek a feladatok lehetnek ugyanazt a kódot futtató és ugyanahhoz az erőforráshoz hozzáférést igénylő példányok, vagy előfordulhat, hogy egy összetett számítás egyes részeit végzik párhuzamosan. Például a feladatok egy összetett számítás egyes elemeit párhuzamosan végzik, majd a feladatok befejezése után összesíti az eredményeket.

Milyen tervezési pattern-el lehet biztosítani a koordinált futást?

Válaszok

- ☒ Vezetőválasztási minta (Leader Election)
- ☐ Priority Queue
- ☒ Feladatütemező (Scheduler Agent Supervisor)
- ☐ Erőforrás-konzolidálás (Compute Resource Consolidation)

Magyarázat

Hasznos lehet a vezetőválasztási mintával koordinálni a párhuzamosan futó alkalmazásokat majd összesíteni a folyamat végeredményét. Ugyan ezt az eredményt lehet elérni a Feladatütemező tervezési struktúra használatával is.

8. feladat 0/10 pont

Melyik terhelés elosztást azonosítják az alábbi jellemzők:

Terhelés elosztót használ (Load Balancer)

Virtuálisan végtelen terhelést is képes kezelni

Több szerver együttműködve szolgálja ki a kérést

Válasz

☒ Horizontális

☐ Vertikális

Magyarázat

A horizontális elosztás újabb és újabb szerverek hozzáadásával virtuálisan végtelen terhelést is képes kiszolgálni.