DEVOPS

3. forduló

A kategória támogatója: E.ON Digital Technology

Ismertető a feladatlaphoz

Kérjük, hogy a feladatlap indítása előtt mindenképp olvasd el az alábbi útmutatót:

> Amennyiben olyan kategóriában játszol, ahol van csatolmány, de hibába ütközöl a letöltésnél, ott valószínűleg a vírusirtó korlátoz, annak ideiglenes kikapcsolása megoldhatja a problémát. (Körülbelül minden 3000. letöltésnél fordul ez elő.)

> Helyezéseket a 4. forduló után mutatunk, százalékos formában: adott kategóriában a TOP 20-40-60%-hoz tartozol.

> A feltűnően rövid idő alatt megoldott feladatlapok kizárást vonnak maguk után, bármilyen más gyanús esetben fenntartjuk a jogot a forduló érvénytelenítésére!

A hét témája: Continuous Integration, Continuous Delivery

Jó versenyzést kívánunk!

1. feladat 2 pont

Milyen előnyei vannak a Continuous Delivery (folyamatos szállítás) pipeline-ok használatának a manuális folyamatokhoz képest?

Válaszok

- [ ] A folyamatok felgyorsítása által lehetőséget teremt a gyakoribb szoftverkiadásokra.
- [ ] Emberi hibalehetőségek csökkentése.
- [ ] Konzisztencia és megismételhetőség.
- [ ] Csökkenti a fejlesztők feladatait, mivel a pipeline-okat kizárólag az üzemeltetői csapat kezeli.

2. feladat 2 pont

Milyen teszt típusokat és ellenőrzéseket használhatunk egy teljesen automatizált Continuous Delivery folyamatban?

Válaszok

- [ ] Linting
- [ ] Smoke testing
- [ ] Statikus kód analízis
- [ ] Unit teszt
- [ ] Manuális tesztelés
- [ ] Exploratory testing

3. feladat 1 pont

Mit értünk 80%-os kód lefedettség (code coverage) alatt?

Válasz

- ( ) A forráskód 80%-a buildelhető, a többi szintaktikai hibát tartalmaz.
- ( ) A forráskód megváltoztatott sorainak 80%-a került átnézésre a vezető fejlesztő által a pull request folyamat során.

- [ ] A forráskódban található eljárások 80%-a került dokumentálásra.
- [ ] A forráskód sorainak vagy utasításainak 80%-a került ellenőrzésre automatizált teszttel.

4. feladat 2 pont

Mi a Canary Deployment (kanári telepítési stratégia)?

Válasz

- [ ] A szoftver új verziójának csak egy kis felhasználói bázis számára elérhetővé tétele, hogy a szoftver stabilitását és működését ellenőrizzük a széleskörű telepítés előtt.
- [ ] A szoftver új verziójának tesztelés nélküli telepítése (“fail fast” elv).
- [ ] A szoftver különböző verzióinak párhuzamos futtatása teljesítményteszt céljából.
- [ ] A szoftver új verziójának telepítése az éles környezet másolatán, majd forgalom átirányítása az új környezetre.
- [ ] Canary.js használata a frontenden a felhasználók eltérő backendre irányításához.

5. feladat 3 pont

A csapat által fejlesztett és üzemeltetett alkalmazás forráskódja egy publikus Git repositoryban helyezkedik el, a build folyamat során egy Docker image készül, mely szintén publikusan elérhető.
Melyik megoldás NEM ajánlott az alkalmazás által használt jelszavak tárolására?

Válaszok

- [ ] Forráskódban, saját fejlesztésű titkosítást alkalmazva
- [ ] SOPS (Secrets OPerationS)
- [ ] Docker fordítás-idejű paraméterek (docker build --build-arg)
- [ ] Azure Key Vault
- [ ] AWS Secrets Manager
- [ ] Ansible Vault

Megoldások beküldése