Tesztautomatizálás vizsga

# Architektúra fogalma

Adott egy háromrétegű vékonyklienses alkalmazás, adatbázissal (pl. locations alkalmazás). Írj ennek az architektúrájáról! Éles üzemben hány számítógépre érdemes feltelepíteni? A futtatásához minimum hány számítógép kell? Milyen komponensekből áll össze? Minek mi a feladata? Ezek hogyan kapcsolódnak egymáshoz? Sorolj fel legalább két teszteszközt, és írd le, hova kapcsolódott, és mit tesztelt!

Csak a fontos részeket emeld ki, amik elengedhetetlenek a működéshez! Tegyük fel, hogy számítógép feltelepített operációs rendszerrel már rendelkezésre áll, azt nem kell részletezni.

# Batch fájlok

Írj egy b.bat fájlt, mely ha futtatáskor kap egy verbose paramétert, kiírja, hogy A MY\_DIR értéke:, és utána a MY\_DIR környezeti változó értékét.

Azaz futtatva:

b.bat verbose

A MY\_DIR értéke: C:\TMP

Ha nem verbose paraméterrel hívjuk, akkor ne írjon ki semmit!

Írj egy a.bat fájlt, mely beállítja a MY\_DIR környezeti változót, legyen az értéke C:\TMP és írja is ki annak értékét! Majd hívja meg a b.bat fájlt verbose paraméterrel. Ezt futtatva ezt kell kapnunk:

a.bat

C:\TMP

A MY\_DIR értéke: C:\TMP

Egyik bat fájl se írja ki a végrehajtandó parancsokat!

Töltsd fel a két bat fájlt egy zip állományba tömörítve!

# Selenium IDE

Adott egy alkalmazás, mely három egész számot vár, melyek a háromszög oldalainak hossza. Az alkalmazás eldönti a háromszög típusát. Az alkalmazás megtalálható a következő címen:

<http://tadev.training360.com/epbva/>

Tervezd meg fejben a teszteseteket, majd implementáld őket Selenium IDE segítségével!

Mentsd el a projektet, és töltd fel!

A projektet és a teszteseteket értelmesen nevezd el! A tesztesetekben CSAK a szükséges lépések legyenek, egyéb más klikk sem szerepelhet benne!

# SoapUI

Az előző alkalmazás SOAP webszolgáltatást is biztosít. Ennek WSDL állománya elérhető a következő címen:

<http://tadev.training360.com/epbva/ws/triangle.wsdl>

Térképezd fel a webszolgáltatás működését! Milyen input esetén milyen értékeket ad vissza?

Implementáld az előbb megtervezett teszteseteket SoapUI-ban is. Beszédesen nevezd el a teszteseteket!

A SoapUI projekt fájlt töltsd fel!

# Naplóelemzés

Jelentkezz be SSH-val a tadev.training360.com gépre, ahol a felhasználónév developer, a jelszó developer!

Az előző alkalmazás napló állományát a /opt/training/epbva/logs könyvtárban találod.

Találd ki, hogy milyen hibaüzenetet kerül a napló állományba, ha nem well-formed (azaz rossz) XML-t küldesz be SoapUI-jal!

Listázd ki, hogy az adott órában megjelenő ilyen jellegű hibaüzenetek számát!

Erről tölts fel egy screenshotot, ahogy először kiadsz egy parancsot, mely a log állomány utolsó 5 sorát írja ki, majd kiadod a parancsot, mely megjeleníti a hibaüzenetek számát, és az kiírja a végeredményt!