

Blockchain technológiák és alkalmazások

3 . Exam slot token - DAML

Tervezői döntések

A feladatleírásnak megfelelően létrehoztunk egy ExamCoin template-t, amely biztosítja, hogy a vizsgák cserélhetőek legyenek egymás között ezen template egységeiért. Csak olyan ExamCoin-nal lehet vizsgát venni, amelyet az egyetem állít ki, illetve az utalás hitelesítéshez mind 2 fél szükséges, ezáltal kikényszerítve, hogy a pénz „nyomonkövethető legyen” (látható legyen, hogy kitől származik közvetlenül a bevétele, ezáltal esetlegesen az üzérkedő hallgatókat lehet követni). A fizetőeszköz fentiekén túl „osztható”, azért, hogy a megfelelő mennyiségű eszköz (se több, se kevesebb) álljon rendelkezésre az ExamSlot-ok vásárlása esetén.

A ERC777 szabvány fontos elemeit megvalósítottuk a házi feladatban:

- Ennek megfelelően az ExamSlot-ot az azt létrehozó Egyetem el tudja „égetni” (Destroy). Ez nem történik meg automatikusan a vizsga megtörténte után, az archiválási időszakok az intézményre vannak bízva.
- A vizsgahely továbbadhatósága beállítható, ezt egy boolean érték állításával dönthetjük el.

ExamSlot jellemzői: Egy vizsgahely a vizsga nevével és dátumával egyértelműen azonosítható, és létrehozója kezdetben minden esetben az egyetem, illetve meghatározott körű diák számára elérhető. A feladatleírásnak megfelelően kezdetben a vizsgahelyek számának megfelelően mindenki számára „ingyenesen” hozzáférhetőek a vizsgahelyek. A hallgatók saját maguk állíthatják be később, hogy el akarják-e adni, illetve, hogy mennyiért. Az első „megvétel” után alapértelmezés szerint eladhatatlan állapotba kerülnek, illetve 0 ExamCoin lesz az áruk.

A vizsgahelyekre az alábbi megkötések vonatkoznak:

- Amennyiben az eladható állapotban van, és egy másik vizsgázónak van elég egyetem által kiállított pénze rá, akkor az adásvétel meg fog történni (kivéve, ha adott nevű ExamSlot-tal már rendelkezik a venni szándékozó hallgató)
- A létrehozó határozza meg, hogy hány darab hely van vizsgaidőpontként és vizsgánként
- A létrehozónak nem szükséges felügyelnie a vizsgahelyekre való jelentkezést, azért a diákok „küzdenek meg”
- Egy hallgató csak saját magának választhat ki vizsgahelyet

Implementáció

ExamCoin

A feladatleírásban szereplő „unit of value”, melynek értéke vizsgákban mérhető.

`ContractId, ExamCoinTransfer` **Transfer**(`newOwner` : `Party`): Segítségével a pénz tulajdonosa átutalást tud kezdeményezni a paraméterként átadott `newOwner` számára. Egy ExamCoinTransfer kerül létrehozásra.

ContractId ExamCoin, ContractId ExamCoin Split(at : Int): Szétosztja a hallgató ExamCoin-ját két részre: a paraméternek megfelelő mennyiségű részre és az eredeti mennyiség paraméternek megfelelő számmal csökkentett részre. A felhasználó csak pozitív paraméterrel hívhatja meg, illetve a paraméternek kisebbnek kell lennie, mint az ExamCoin-ban lévő mennyiség.

ExamCoinTransfer

Az egyetem által létrehozott pénz hallgatóhoz utalását lehetővé tevő osztály.

ContractId ExamCoin Accept(): Minden diák a neki szóló utalást elfogadhatja, ezáltal új ExamCoin-hoz jutva.

ExamSlot

Minden vizsgához meghatározott számban tartozik, diákok kereskedhetnek vele szabadon.

„void” **Destroy()**: A létrehozó törölheti bármelyik vizsgahelyet. (de csak ő)

ContractId ExamSlot SetPrice(newPrice : Int): A tulajdonos beállíthatja, hogy mennyibe kerül az adott vizsgahely, ezáltal automatikusan eladhatóvá is téve azt.

ContractId ExamSlot, ContractId ExamCoin Buy(coinCid : ContractId ExamCoin , newOwner : Party): Egy megvételre kínált vizsgahely vehető meg a függvény segítségével abban az esetben, ha nincs erre a vizsgára más alkalma, illetve, ha a felkínált ExamCoin összege pontosan megegyezik a kikiáltási árral. Ekkor a vizsgahely alapértelmezett állapotban (nem eladható, melynek ára 0 egység) kerül az új tulajdonoshoz, a kikiáltási árnak megfelelő ExamCoin pedig a vizsgahely eredeti tulajdonosához kerül.

Exam

Vizsgaosztály, melynek létrehozásáról az egyetem gondoskodik a megfelelő vizsgahelyszám és a jogosult hallgatók beállításával.

ContractId ExamSlot, ContractId Exam TakeSlot(student: Party): A hallgató saját magának elfoglalhat egy vizsgahelyet, amennyiben nem rendelkezik más ugyanezen nevű (tehát ugyanabból a tárgyból való) vizsgahellyel. A vizsgahely csak akkor vihető el, ha a létrehozó által megadott helyek száma még nem fogyott el, ugyanis minden sikeres foglalás csökkenti eggyel a vizsgánként és időpontonként rendelkezésre álló helyek számát.

Teszt esetek

1. Ellenőrizzük, hogy egy frissen vásárolt vizsgahely megvehető az azt megvehető hallgatótól, annak beleegyezése nélkül.

Lépések:

- a. Létrehozza az egyetem a vizsgát
- b. Az egyetem létrehoz ExamCoin-t
- c. Az egyetem odaadja az ExamCoin-t az egyes számú hallgatónak
- d. Egyes számú hallgató elfogadja az ExamCoin-t
- e. Egyes számú hallgató elfoglal egy vizsgahelyet

- f. Kettes számú hallgató **sikertelenül** (submitMustFail) próbálja meg elvenni tőle ezt a vizsgahelyet
- 2. Teszteljük, hogy sikeresen megvehető egy eladásra szánt vizsgahely.
Lépések:
 - a. Létrehozza az egyetem a vizsgát
 - b. Az egyetem létrehoz ExamCoin-t
 - c. Az egyetem odaadja az ExamCoin-t az egyes számú hallgatónak
 - d. Egyes számú hallgató elfogadja az ExamCoin-t
 - e. Egyes számú hallgató elfoglal egy vizsgahelyet
 - f. Egyes számú hallgató beállítja a vizsgahely árát (ezáltal eladandóvá állítva a státuszát)
 - g. Kettes számú hallgató megveszi tőle a vizsgahelyet -> **sikerül**
- 3. Teszteljük, hogy sikeresen adható-e ExamCoin hallgatónak.
Lépések:
 - a. Az egyetem létrehoz ExamCoin-t
 - b. Az egyetem odaadja az ExamCoin-t az egyes számú hallgatónak
 - c. Egyes számú hallgató elfogadja az ExamCoin-t -> **sikerül**
- 4. Teszteljük, hogy valaki, aki nem tulajdonosa az ExamCoin-nak szét tudja-e darabolni azt.
Lépések:
 - a. Az egyetem létrehoz ExamCoin-t
 - b. Egyes számú hallgató megpróbálja elutalni magának a pénzt -> **nem fog sikerülni**
 - c. Egyes számú hallgató megpróbálja feldarabolni az eredeti összeget -> **ez sem fog sikerülni**
- 5. Többszörösen teszteljük, hogy a pénz darabolható-e a tulajdonosa által
 - a. Az egyetem létrehoz ExamCoin-t
 - b. Az egyetem felosztja 60:40 arányban -> **sikerül**
 - c. Az egyetem utalást kezdeményez az egyes számú hallgató felé
 - d. Egyes számú hallgató ezt az összeget (60 ExamCoin) elfogadja
 - e. Egyes számú hallgató elfelezi ezt az összeget -> **sikerül**