

ANALÍZIS MODELL KIDOLGOZÁSA II.

40 – ZETA

Konzulens:
KOVÁCS BOLDIZSÁR

Csapattagok

Alpek Dávid Zsolt
Csia Klaudia Kitty
Litavecz Marcell
Marton Judit

C31X0F
HA5YCV
IPHJNB
M0MYIM

alpek.david.zsolt@gmail.com
kitkat@sch.bme.hu
marcell.litavecz@gmail.com
judit.marton@edu.bme.hu

2022. 03. 21.

4. Analízis modell kidolgozása I.

4.1 Objektum katalógus

4.1.1 Játék

A **játék menedzseléséért** felelős objektum. Ez inicializálja a dolgokat a játék kezdetén, helyezi el a játékosokat a kiindulómezőre, illetve helyezi el a genetikai kódokat a laborokba. A virológus ennek az objektumnak üzen, ha megszerezte a kellő mennyiségű genetikai kódot, így ennek az objektumnak a felelőssége lezárni a játékot, és erről értesíteni a játékosokat is.

4.1.2 Időzítő

Az **időzítések** kezelésért felelős. Megadott időnként lépteti a *Steppable* interfészű objektumokat. Csak egy darab létezik belőle.

4.1.3 Mező

A **pálya egy egységnyi része**. A virológusok ezeken mozognak, továbbá ezeken találhatóak meg a gyűjthető elemek, mint a genetikai kódok, eszközök és anyagok. Egy mezőnek több másik szomszédja is lehet, akár különböző típusúak is.

4.1.4 Laboratórium

A mezővel megegyező tulajdonságú játékelem. A virológus itt tudja megtanulni a **genetikai kódokat**, mely később az ágens generálásához, illetve a játék megnyeréséhez szükséges.

4.1.5 Óvóhely

A mezővel megegyező tulajdonságú játékelem. A virológus itt tudja összegyűjteni és felvenni a **felszereléseket**, mint a kesztyű, a köpeny, és a zsák.

4.1.6 Raktár

A mezővel megegyező tulajdonságú játékelem. A virológus itt tudja összegyűjteni az **ágens generálásához szükséges anyagokat**, mint a nukleotid vagy az aminosav.

4.1.7 Virológus

Tárolja és rendelkezésre bocsátja a **leltárat**. Felveszi, illetve leveszi magáról az **eszközöket**, a levett eszközöket belerakja az leltárba, amennyiben van elegendő hely benne. Ha nincs elegendő hely, akkor eldobja (megsemmisíti) őket. Megtámad egy másik **virológust (ágenst)** ken rá), illetve védekezik az őt ért támadás során. A védekezésnél használhatja az eszközeit (max 3 eszköz, melyeket önmagán hord). Letapogatja a labor falán található **kódot**. Ágenseket generál. Szomszédos **mezőre lép**. Eszközöket és anyagokat **lop** egy másik, lebénult virológustól.

4.1.8 Leltár

Ez egy olyan eszköz, amely a játék kezdete óta a karakternél van. Ide tudja **begyűjteni** a játékos a **genetikai kódokat**, az **anyagokat** és az **eszközöket** is. Minden játékosnál csak egy darab ilyen található, nem ellopható.

4.1.9 Aminosav

Ez egy olyan anyagtípus, amely szükséges az **ágens** **előállításához**. A játékos a raktárakban vagy esetleg más játékosoktól jut hozzá. Más különleges képessége nincs.

4.1.10 Nukleotid

Ez egy olyan anyagtípus, amely szükséges az **ágens** **előállításához**. A játékos a raktárakban vagy esetleg más játékosoktól jut hozzá. Más különleges képessége nincs.

4.1.11 Vitustánc

Az előállítható **ágensek** egyik fajtája, melyek a virológus mind magára, mind másik virológusra is elhasználhat. Hatása, hogy a **játékos elveszíti az irányítást a karakter felett**, aki véletlenszerű mozgást kezd el végezni, azaz elkezdi ezt a „táncot”, ameddig a hatás le nem jár a karakterről.

4.1.12 Immunitás

Az előállítható **ágensek** egyik fajtája, melyek a virológus mind magára, mind másik virológusra is elhasználhat. Hatása, hogy a játékos **védettséget élvez** más ágensekkel szemben”, ameddig a hatás le nem jár.

4.1.13 Bénulás

Az előállítható **ágensek** egyik fajtája, melyek a virológus mind magára, mind másik virológusra is elhasználhat. Hatása, hogy a játékos **mozgás- és cselekvőképtelenné válik**, ameddig a hatás le nem jár.

4.1.14 Felejtés

Az előállítható **ágensek** egyik fajtája, melyek a virológus mind magára, mind másik virológusra is elhasználhat. Hatása, hogy a játékos **elfelejti az eddig megtanult összes genetikai kódot**. A hatás végleges és nem visszafordítható.

4.1.15 Kesztyű

Ez egy olyan **felszerelés** objektum, amelyet a virológus az óvóhelyen tud összegyűjteni és felvenni magára, de akár más játékostól is meg tudja szerezni. Képessége, hogy **vissza** tudja **dobni** a rákent ágenst.

4.1.16 Köpeny

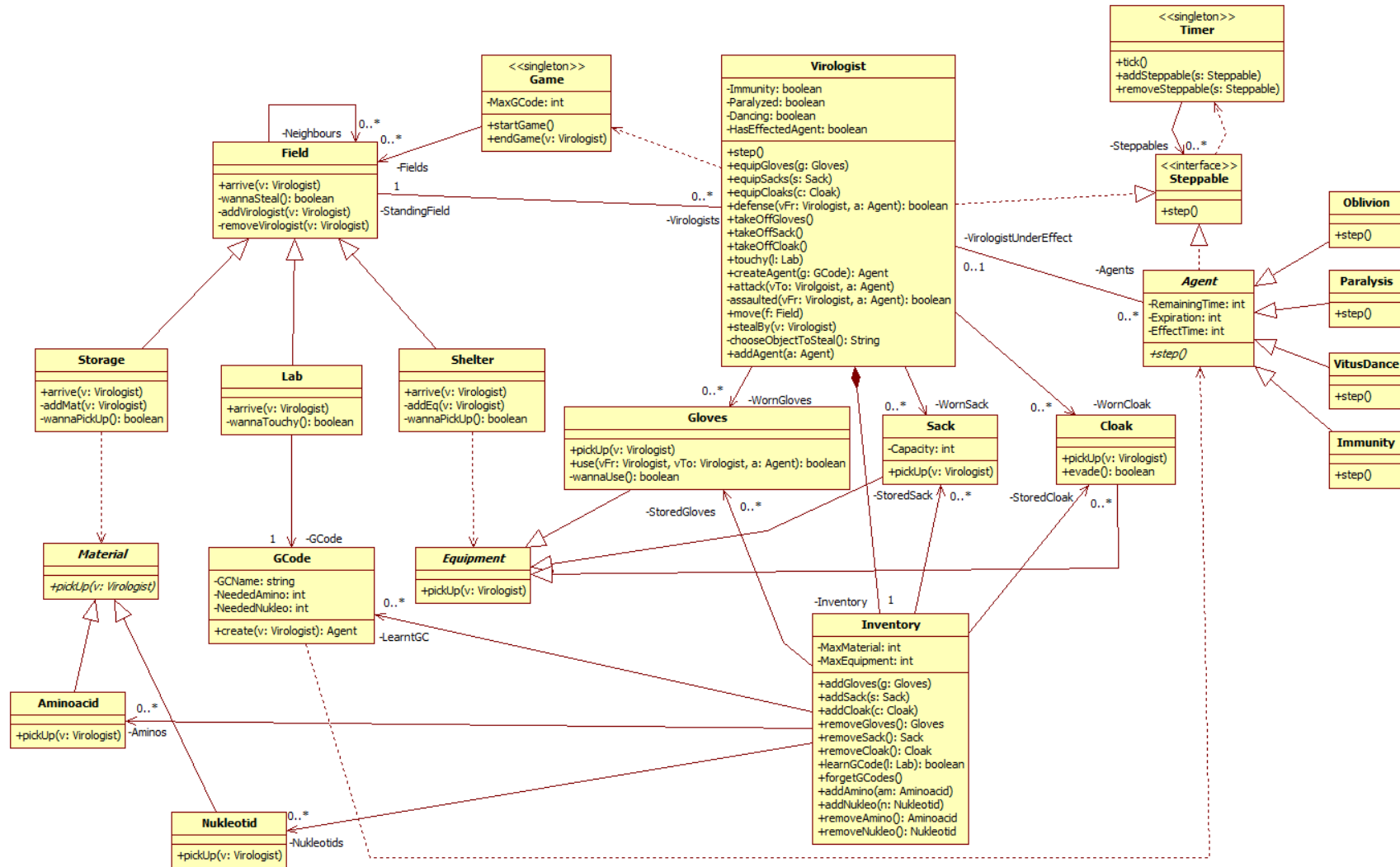
Ez egy olyan **felszerelés** objektum, amelyet a virológus az óvóhelyen tud összegyűjteni és felvenni magára, de akár más játékostól is meg tudja szerezni. Képessége, hogy 82,3%-os hatással **tartja távol** az ágenseket.

4.1.17 Zsák

Ez egy olyan **felszerelés** objektum, amelyet a virológus az óvóhelyen tud összegyűjteni és felvenni magára, de akár más játékostól is meg tudja szerezni. Képessége, hogy **meg** tudja **növelni a leltár anyaggyűjtő kapacitását**.

4.2 Statikus struktúra diagramok

A jobb áttekinthetőség érdekében pár metódust (pl. Getter-Setter metódusokat) nem jelenítettünk meg a diagramon.



4.3 Osztályok leírása

4.3.1 Immunity

- **Felelősség**
 - Speciális típusú ágens, amely megvédi a virológust attól, hogy mások által használt ágensek hatással legyenek rá. Hatása 4 körön keresztül tart.
- **Ősosztályok**
 - Agent
- **Interfészek**
 - Steppable
- **Attribútumok**
 - -
- **Metódusok**
 - **void step()**: Ha nem járt le az időzítő, akkor beállítja igazra azon virológus immunitás attribútumát, amelyekre hat az ágens és csökkenti az időzítőt. Ha lejárt, akkor a virológus immunitás attribútumát hamisra állítja és törli a virológusra éppen ható ágenst attribútumot.

4.3.2 Paralysis

- **Felelősség**
 - Olyan ágens, amely megbénít, így amíg az ágens hatása tart, az áldozat nem tud semmit csinálni. Hatása 5 körön keresztül tart.
- **Ősosztályok**
 - Agent
- **Interfészek**
 - Steppable
- **Attribútumok**
 - -
- **Metódusok**
 - **void step()**: Ha nem járt le az időzítő, akkor beállítja igazra azon virológus bénultság attribútumát, amelyekre hat az ágens és csökkenti az időzítőt. Ha lejárt, akkor a virológus bénultság attribútumát hamisra állítja és törli a virológusra éppen ható ágenst attribútumot.

4.3.3 Oblivion

- **Felelősség**
 - Ezen ágens hatására az áldozat elfelejti a már megismert genetikai kódokat. Hatása 1 körön keresztül tart.
- **Ősosztályok**
 - Agent
- **Interfészek**
 - Steppable
- **Attribútumok**
 - -

- **Metódusok**

- **void step():** Ha nem járt a le az időzítő, akkor elkéri azon virológus immunitás leltárját, amelyikre hat az ágens és a felejtés metódus segítségével törli az összes megismert genetikai kódot. Majd az időzítőt nullára állítja, illetve törli a virológusra éppen ható ágenst attribútumot.

4.3.4 VitusDance

- **Felelősség**

- Ennek hatására a virológus véletlenszerűen kezd el a haladni a pályán. Hatása 6 körön keresztül tart.

- **Össztályok**

- Agent

- **Interfészek**

- Steppable

- **Attribútumok**

- -

- **Metódusok**

- **void step():** Ha nem járt le az időzítő, akkor beállítja igazra azon virológus táncolás attribútumát, amelyikre hat az ágens, csökkenti az időzítőt és egy véletlenszerűen választott szomszédos mezőre lépteti. Ha lejárt, akkor a virológus táncolás attribútumát hamisra állítja és törli a virológusra éppen ható ágenst attribútumot.

4.3.5 Agent

- **Felelősség**

- Absztrakt osztály, mely keretet biztosít a különböző ágens típusok együttes kezelésére.

- **Össztályok**

- -

- **Interfészek**

- Steppable

- **Attribútumok**

- **RemaningTime:** A fennmaradt szavatossági időt vagy a fennmaradt hatás idejét tárolja.
- **Expiration:** A szavatossági idő, ami azt az időt jelenti ameddig az elkészített ágenst van lehetőség felhasználni.
- **EffectTime:** A hatás ideje, ami azt az időt jelenti ameddig egy ágens hat egy virológusra.
- **VirologistUnderEffect:** Az ágens hatása alatt álló virológust tárolja.

- **Metódusok**

- **void step():** Absztrakt léptető metódus.
- **void setRemainingTime(time: int):** beállítja a remainingTime attribútumot a paraméterben kapott értékre.

4.3.6 Steppable

- **Felelősség**

- Egy interfész, ami minden olyan dolgot reprezentál, amely időben lépni tud.

- **Össztályok**

- -
- **Interfészek**
 - -
- **Attribútumok**
 -
- **Metódusok**
 - **void step():** Az adott lépésben végrehajtandó művelet.

4.3.7 Timer

- **Felelősség**
 - Periodikus időzítőt reprezentál a játékban, a léptethető (Steppable) dolgokat lépteti.
- **Ósosztályok**
 - -
- **Interfészek**
 - Steppable
- **Attribútumok**
 - **Steppable:** A léptethető dolgok.
- **Metódusok**
 - **void tick():** Minden léptethető dolog léptetése.
 - **void AddSteppable(s:Steppable):** Új léptethető dolog hozzáadása.
 - **void RemoveSteppable(s:Steppable):** Léptethető dolog törlése.

4.3.8 Virologist

- **Felelősség**
 - Lehetővé teszi a felhasználó és a rendszer közötti kommunikációt. A játékos ezen az osztályon keresztül mozoghat a szomszédos mezők között, ágenszt generálhat, felvehet és levehet eszközöket (amelyekből maximum 3 lehet rajta egyszerre), ellenfeleire vagy önmagára ágenszt kenhet, és védekezhet egy általa indított támadás (ágens kenés) ellen. Emellett eszközöket, anyagokat lophat egy bénult virológustól.
- **Ósosztályok**
 - -
- **Interfészek**
 - Steppable
- **Attribútumok**
 - **Immunity:** Megadja, hogy a virológus immunis-e a rákent ágenssekkel szemben.
 - **Paralyzed:** Megadja, hogy a virológus le van-e bénulva.
 - **Dancing:** Megadja, hogy a virológus épp vitustánc hatása alatt áll-e.
 - **HasEffectAgent:** Megadja, hogy a virológus jelen pillanatban ágens hatása alatt áll-e.
 - **WornGloves:** A virológus által viselt kesztyűk listája, melyben minimum 0, maximum 3 elem lehet (attól függően, hogy a másik két eszköz fajta listája mennyi elemből áll).
 - **WornSack:** A virológus által viselt zsákok listája, melyben minimum 0, maximum 3 elem lehet (attól függően, hogy a másik két eszköz fajta listája mennyi elemből áll).
 - **WornCloak:** A virológus által viselt köpenyek listája, melyben minimum 0, maximum 3 elem lehet (attól függően, hogy a másik két eszköz fajta listája mennyi elemből áll).

- **StandingField:** Az a mező, amelyen a virológus áll.
- **Inventory:** A virológus leltára.
- **Metódusok**
 - **void setImmunity(b: boolean):** Beállítja az *Immunity* attribútum értékét a *b* értékére.
 - **void setVirologistUnderEffect(a: Agent):** Beállítja a virológusra éppen ható ágens attribútum értékét a bementi paraméterére.
 - **void setDancing(b: boolean):** Beállítja a *Dancing* attribútum értékét a bemeneti paraméterre.
 - **Field getStandingField():** Visszaadja azt a mezőt, amin a virológus éppen van.
 - **void equipGloves(g: Gloves):** Hozzáad egy kesztyűt a virológuson lévő kesztyűk listájához, a kesztyűt kitörli az *inventory*-ből.
 - **void equipSacks(g: Gloves):** Hozzáad egy zsákot a virológuson lévő zsákok listájához, a zsákot kitörli az *inventory*-ből, valamint megnöveli a maximálisan gyűjthető anyagok(nukleotid, aminosav) mennyiségét 5-tel.
 - **void equipCloak(g: Gloves):** Hozzáad egy köpenyt a virológuson lévő köpenyek listájához, a köpenyt kitörli az *inventory*-ből.
 - **boolean defense(vFr: Virologist, a: Agent):** Ha a virológuson lévő kesztyű listája nem üres, akkor meghívja az első kesztyű *use* metódusát. Amennyiben ennek visszatérési értéke *false* (vagyis a játékos nem használta a kesztyűt), akkor megnézzük a virológuson lévő köpenyek listáját. Ha a lista üres, akkor *false*-szal visszatérünk. Egyébként kivesz egy köpenyt a listából, és meghívja a köpeny *evade()* metódusát. Ha az *evade()* metódus *false*-szal tér vissza, akkor kiveszi a következő köpenyt, és megismétli rajta az előző műveletet, amíg a lista ki nem ürül. Amennyiben egy köpeny *evade()* metódusa *true*-val tér vissza, a folyamat megszakad. Végül visszaadja a legutoljára meghívott *evade* metódus értékét. Amennyiben a kesztyű *use()* metódusa *true*-val tér vissza (a játékos használta a kesztyűt), akkor törölünk egy kesztyűt a viselt eszközök közül, és a köpenyek listájának vizsgálata nélkül *true*-val visszatérünk.
 - **void takeOffGloves():** Ha a virológus nincs lebénulva, leveszi a rajta lévő kesztyűt, ha van elég hely, akkor berakja a leltárba, egyébként eldobja.
 - **void takeOffSack():** Ha a virológus nincs lebénulva, leveszi a rajta lévő zsákot, ha van elég hely, akkor berakja a leltárba, egyébként eldobja.
 - **void takeOffCloak():** Ha a virológus nincs lebénulva, leveszi a rajta lévő köpenyt, ha van elég hely, akkor berakja a leltárba, egyébként eldobja.
 - **Agent createAgent(g : GCode):** Létrehoz egy ágens (a paraméterben kapott *GCode* objektum *create* metódusának meghívásával), amennyiben ez sikeres volt (a *create* metódus visszatérési értéke nem null), akkor hozzáadja az ágens a *steppable* objektumok közé (*Timer addSteppable* metódusát hívva), és ezzel elindítja az ágens időzítőjét. Elhelyezi a létrehozott ágens az ágensek listájába.
 - **void attack(vTo: Virologist, a: Agent):** Megvizsgáljuk, hogy a virológus le van-e bénulva. Ha igen, nem történik semmi, egyébként meghívjuk a *vTo* paraméterben átadott virológus *assaulted()* metódusát, melynek visszatérési értéke jelzi, hogy sikeres volt-e a támadás. Amennyiben a támadás sikerrel járt, megvizsgáljuk, hogy a megtámadott virológus le van-e bénulva. Ha igen, és a támadó virológus tárhelye nincs tele, meghívjuk a virológus *stealBy()* metódusát.
 - **boolean assaulted(Virologist vFr, Agent a):** Megnézi, hogy a virológuson van-e bármilyen hatás. Ha van, akkor nem történik semmi, mivel a virológusra egy időben csak egyetlen ágens hathat. Amennyiben nincs, és a virológus nem önmagára indítja a támadást (kenést), akkor meghívja a *defense()* metódust. Ha a *defense()* metódus *false*-szal tér vissza, inicializáljuk az ágens *remainingTime* attribútumát az *effectTime*

attribútumának értékével, beállítjuk az ágens *virologistUnderEffect* attribútumát a megtámadott virológusra, majd elindítjuk az ágens működési folyamatát.

- **void move(f: Field):** A virológus átmozdul egy másik mezőre.
- **void stealBy(v: Virologist):** A felhasználó meglophat egy másik virológust, amennyiben az le van bénulva. A függvény elkéri a paraméterben kapott virológus inventory-ját, majd meghívja a *chooseObjectToSteal()* metódust. Ezután a visszatérési értéknek megfelelően eltávolít egy eszközt vagy material-t a saját inventory-jából, és ha ez sikeres volt (volt az inventory-jában olyan eszköz / material), akkor hozzáadja azt a paraméterként kapott virológus inventory-jához. A virológus által viselt eszköz nem ellopható, csak az, ami az inventory-ban található.
- **void addAgent(a: Agent):** Hozzáad egy ágenszt az ágensek listájába.
- **void step():** A virológus által már létrehozott, de fel nem használt ágensek lejáratí idejét csökkenti, amennyiben a lejáratí idő 0, akkor törli az adott ágenszt a listából.
- **void touchy(l: Lab):** meghívja a virológus inventory-jának learnGCode() metódusát, amennyiben ennek visszatérése true (tanult új kódot), akkor megvizsgálja, hogy az összes kódot ismeri-e már. Ha az összes kódot megtanulta, akkor meghívja a Game objektum endGame(v:Virologist) metódusát.
- **String chooseObjectToSteal():** A játékos kiválasztja, hogy mit szeretne ellopni, az ellopandó eszköz vagy material nevének kiválasztásával. A visszatérési érték a kiválasztott név.

4.3.9 Inventory

- **Felelősség**
 - Ezen osztály felelőssége a virológus által összegyűjtött eszközök, anyagok tárolása, felvétele.
- **Össztályok**
 - -
- **Interfészek**
 - -
- **Attribútumok**
 - **MaxMaterial:** Maximálisan eltárolható anyagok száma, 20 db
 - **MaxEquipment:** Maximálisan eltárolható eszközök száma, 5db
 - **StoredGloves:** Eltárolt kesztyűk listája.
 - **StoredSack:** Eltárolt zsákok listája.
 - **StoredCloak:** Eltárolt köpenyek listája.
 - **LearntGC:** Már megismert kódok listája.
 - **Aminos:** Megszerzett aminosavak listája.
 - **Nukleotids:** Megszerzett nukleotidok listája.
- **Metódusok**
 - **Inventory getInventory ():** Visszaadja a virológus által tárolt Inventory objektumot.
 - **void addGloves(g: Gloves):** Hozzáad egy kesztyűt az inventory kesztyű listájához, amennyiben van elég hely az inventory-ban. Ellenkező esetben eldobja azt. (Objektum megsemmisítése)
 - **void addSack(s: Sack):** Hozzáad egy zsákot az inventory zsák listájához, amennyiben van elég hely az inventory-ban. Ellenkező esetben eldobja azt. (Objektum megsemmisítése)
 - **void addCloak(c: Cloak):** Hozzáad egy köpenyt az inventory köpeny listájához, amennyiben van elég hely az inventory-ban. Ellenkező esetben eldobja azt. (Objektum megsemmisítése)

- **Gloves removeGloves()**: Eltávolít egy kesztyűt az inventoryból, és visszatér az eltávolított objektummal. (Ha a lista üres, akkor változatlan marad, és NULL értékkel tér vissza)
- **void removeSack()**: Eltávolít egy zsákot az inventoryból, és visszatér az eltávolított objektummal. (Ha a lista üres, akkor változatlan marad, és NULL értékkel tér vissza)
- **void removeCloak()**: Eltávolít egy köpenyt az inventoryból, és visszatér az eltávolított objektummal. (Ha a lista üres, akkor változatlan marad, és NULL értékkel tér vissza)
- **boolean learnGCode(l: Lab)**: Leellenőrzi, hogy ismeri-e a kódot, amit meg akar tanulni a virológus, ha nem, akkor hozzáadja a már megismert kódokhoz az újat. Ha már ismeri ezt a kódot, akkor nem adja hozzá. Visszatérése jelzi, hogy hozzáadódott-e új elem a listához.
- **void forgetGCodes()**: Törli a virológus GCode-kat tároló listájának minden elemét.
- **void addAmino(am: Aminoacid)**: Hozzáad egy aminosavat az inventoryhoz, amennyiben van elég hely az inventory-ban. Ellenkező esetben eldobja azt. (Objektum megsemmisítése)
- **void addNukleo(n: Nukleotid)**: Hozzáad egy nukleotidot az inventoryhoz, amennyiben van elég hely az inventory-ban. Ellenkező esetben eldobja azt. (Objektum megsemmisítése)
- **void removeAmino()**: Eltávolít egy aminosavat az inventoryból, és visszatér az eltávolított objektummal. (Ha a lista üres, akkor változatlan marad, és NULL értékkel tér vissza)
- **void removeNukleo()**: Eltávolít egy nukleotidot az inventoryból, és visszatér az eltávolított objektummal. (Ha a lista üres, akkor változatlan marad, és NULL értékkel tér vissza)

4.3.10 Field

- **Felelősség**
 - Üres mezőt reprezentáló osztály.
- **Össztályok**
 - -
- **Interfészek**
 - Steppable
- **Attribútumok**
 - Neighbours: A mező szomszédai (Egy mezőnek 4-8 szomszédja lehet).
 - Virologist: A mezőn álló virológusok listája.
- **Metódusok**
 - **Field getARandomNeighbour()**: A mező szomszédok listájából véletlenszerűen visszaad egy elemet. A szomszédok száma legalább 4 kell, hogy legyen, maximum meg az általunk megadott természetes szám, amely jelen játékunk esetében 8.
 - **void arrive(v: Virologist)**: A virológust törli annak a field-nek a listájából, amelyen előzőleg állt (virológus field változójából kiolvasva), majd hozzáadja a saját listájához (ez az új mező, ahová a virológus lépett). Ez után beállítja a virologist *field* változóját az új mezőre (vagyis önmagára). Végül végigiterál a mezőn aktuálisan álló virológusokon, és ha bénultat talál, meghívja a wannaSteal() metódust. Amennyiben ez true-val tér vissza, meghívja a bénult virológus stealBy() metódusát.
 - **boolean wannaSteal()**: Megkérdezi a játékost, hogy szeretne-e lopni az azonos mezőn álló bénult virológustól.
 - **void addVirologist(v: Virologist)**: Hozzáadja a virológust a mező listájához.
 - **void removeVirologist(v: Virologist)**: Eltávolítja a virológust a mező listájából.

4.3.11 Lab

- **Felelősség**
 - A laboratóriumot reprezentáló osztály. Itt találhatóak a genetikai kódok, melyekből ágens készíthető.
- **Ősosztályok**
 - Field
- **Interfészek**
 - Steppable
- **Attribútumok**
 - -
- **Metódusok**
 - **void arrive(v: Virologist):** Meghívja az őosztály arrive(v: Virologist) metódusát, majd a *wannaTouchy*-t is. Ha a *wannaTouchy* metódus true-val tér vissza, akkor meghívja a virológus *touchy()* metódusát is.
 - **boolean wannaTouchy():** Ez a metódus kérdezi meg a felhasználót, hogy akar-e kódot letapogatni.

4.3.12 Storage

- **Felelősség**
 - A raktára reprezentáló osztály. Itt találhatóak az anyagok, melyekből a játékos később ágenst készíthet.
- **Ősosztályok**
 - Field
- **Interfészek**
 - Steppable
- **Attribútumok**
 - -
- **Metódusok**
 - **void arrive(v: Virologist):** Meghívja az őosztály arrive(v: Virologist) metódusát, majd a *wannaPickUp*-ot is. Ha a *wannaPickUp* true-val tér vissza, akkor meghívja a
 - **void addMat(v: Virologist):** A *wannaPickUp()* metódust követően ez a metódus hívódik meg, amely egy random számot kisorsol (annyi szám közül, ahány anyag található az aktuális játékban, jelen esetben ez 2), ehhez létrehoz egy anyagot, és annak hívja meg a megfelelő *PickUp()* metódusát.
 - **boolean wannaPickUp():** Ez a metódus kérdezi meg a felhasználót, hogy akar-e anyagot begyűjteni. Amennyiben igaz visszatérési értéket kap, úgy meghívódik az adott anyagnál a *pickUp()* metódus.

•

4.3.13 Shelter

- **Felelősség**
 - Az óvóhelyet reprezentáló osztály. Itt találhatóak a felszerelések/eszközök, melyeket később a játékos magára vehet.
- **Ősosztályok**
 - Field
- **Interfészek**
 - -
- **Attribútumok**
 - -
- **Metódusok**
 - **void arrive(v: Virologist):** Meghívja az ősosztály *arrive(v: Virologist)* metódusát, majd a *wannaPickUp*-ot is. Amennyiben ez true-val tér vissza, meghívja az *addEq* metódust.
 - **void addEq(eq: Equipment):** Kisorsol egy random számot (annyi szám közül, ahány eszköz található az aktuális játékban, jelen esetben ez 4), ehhez létrehoz egy eszközt, és annak hívja meg a megfelelő *PickUp()* metódusát.
 - **boolean wannaPickUp():** Ez a metódus kérdezi meg a felhasználót, hogy akar-e eszközt begyűjteni. Amennyiben igaz visszatérési értéket kap, úgy meghívódik az adott eszköznél a *pickUp()* metódus.

4.3.14 Equipment

- **Felelősség**
 - A felszerelések/eszközök ősosztálya.
- **Ősosztályok**
 -
- **Interfészek**
 - -
- **Attribútumok**
 - -
- **Metódusok**
 - **void pickUp(v: Virologist):** Absztrakt felvételi metódus.

4.3.15 Gloves

- **Felelősség**
 - Ez az osztály felelős a kesztyű, mint eszköz megszerzéséért. és a kesztyűk használatának a lebonyolításáért.
- **Ősosztályok**
 - Equipment
- **Interfészek**
 - -
- **Attribútumok**
 - -
- **Metódusok**
 - **void pickUp(v: Virologist):** Amikor a virológus egy óvóhelyre lép és fel akar venni egy kesztyűt onnan, akkor azt a virológus *inventory*jába teszi.
 - **use(vFr:Virologist, vTo: Virologist, a:Agent):** Először meghívja a *wannaUse()* metódust. Amennyiben ez false-al tér vissza, ugyanazzal az értékkel visszatérünk, nem történik semmi. Ellenkező esetben meghívjuk a vTo objektum *attack* metódusát, a vFr és a objektumot paraméterként átadva. Amikor használtuk a kesztyűt, true érékkel térünk vissza.
 - **boolean wannaUse():** Megkérdezi a felhasználót, hogy akarja-e használni a kesztyűt, és a felhasználó válaszának megfelelő értékkel visszatér.

4.3.16 Cloak

- **Felelősség**
 - Ez az osztály felelős a köpeny, mint eszköz megszerzéséért és a köpenyek használatának a lebonyolításáért.
- **Ősosztályok**
 - Equipment
- **Interfészek**
 - -
- **Attribútumok**
 - -
- **Metódusok**
 - **void pickUp(v: Virologist):** Amikor a virológus egy óvóhelyre lép és fel akar venni egy köpenyt onnan, akkor azt a virológus *inventory*jába teszi.
 - **Void evade():** Egy olyan algoritmus, mely 82,3%-os valószínűséggel true, 17.7%-os valószínűséggel false értéket ad vissza.

4.3.17 Sack

- **Felelősség**
 - Ez az osztály felelős a zsák, mint eszköz megszerzéséért.
- **Ősosztályok**
 - Equipment
- **Interfészek**
 - -
- **Attribútumok**
 - **Capacity: int:** A zsák kapacitása, amivel növeli a maximum megszerezhető anyagok (nukleotid, aminosav) számát (5 db).
- **Metódusok**
 - **void pickUp(v: Virologist):** Amikor a virológus egy óvóhelyre lép és fel akar venni egy zsákot onnan, akkor azt a virológus *inventoryjába* teszi.

4.3.18 Material

- **Felelősség**
 - A raktárban található anyagok Ősosztálya.
- **Ősosztályok**
 -
- **Interfészek**
 - -
- **Attribútumok**
 - -
- **Metódusok**
 - **void pickUp(v: Virologist):** Absztrakt felvevő metódus.

4.3.19 AminoAcid

- **Felelősség**
 - Az aminosavak megszerzését irányítja.
- **Ősosztályok**
 - Material
- **Interfészek**
 - -
- **Attribútumok**
 - -
- **Metódusok**
 - **void pickUp(v: Virologist):** Indítja az aminosavak felvételét az *inventoryba*.

4.3.20 Nukleotid

- **Felelősség**
 - A nukleotidok megszerzését irányítja.
- **Ősosztályok**
 - Material
- **Interfészek**
 - -
- **Attribútumok**
 - -
- **Metódusok**
 - **void pickUp(v: Virologist):** Indítja a nukleotidok felvételét az *inventory*ba.

4.3.21 GCode

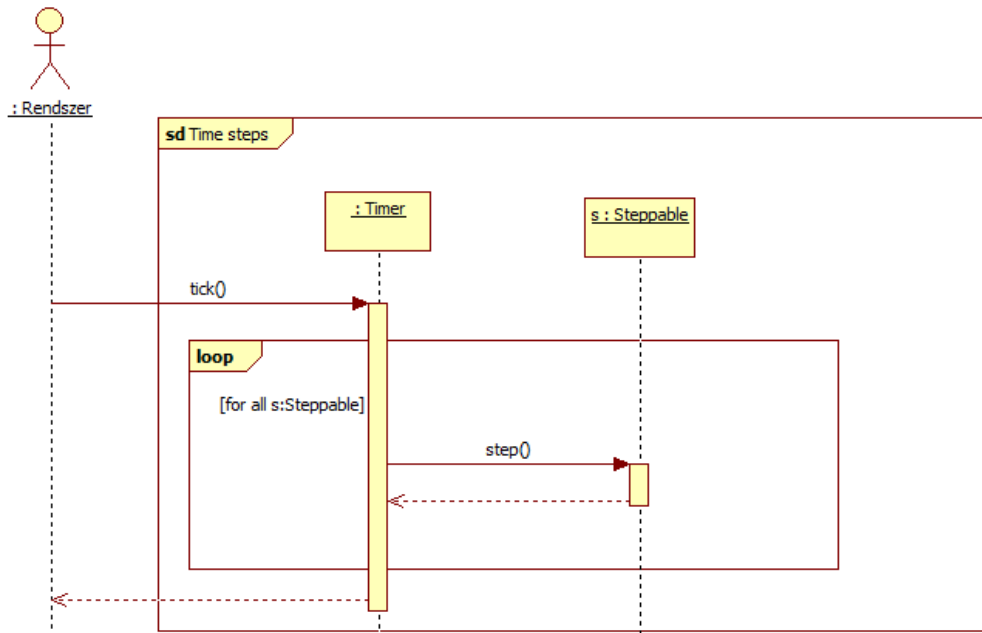
- **Felelősség**
 - A genetikai kódok felépítését tartalmazza, illetve az ágensek létrehozását irányítja.
- **Ősosztályok**
 - -
- **Interfészek**
 - -
- **Attribútumok**
 - **GCName:** A kód alapján előállítható ágens neve.
 - **UsedAmino:** Az ágens előállításához szükséges aminosavak listája.
 - **UsedNukleo:** Az ágens előállításához szükséges nukleotidok listája.
- **Metódusok**
 - **Agent create(inv: Inventory):** Ellenőrzi, hogy az ágens létrehozásához megvan-e a megfelelő mennyiségű anyag az *inventory*ban, ha nincs, akkor visszatér: NULL. Ha megvan, akkor azokat kiveszi a virológus *inventory*jából és létrehozza a megfelelő ágenszt.

4.3.22 Game

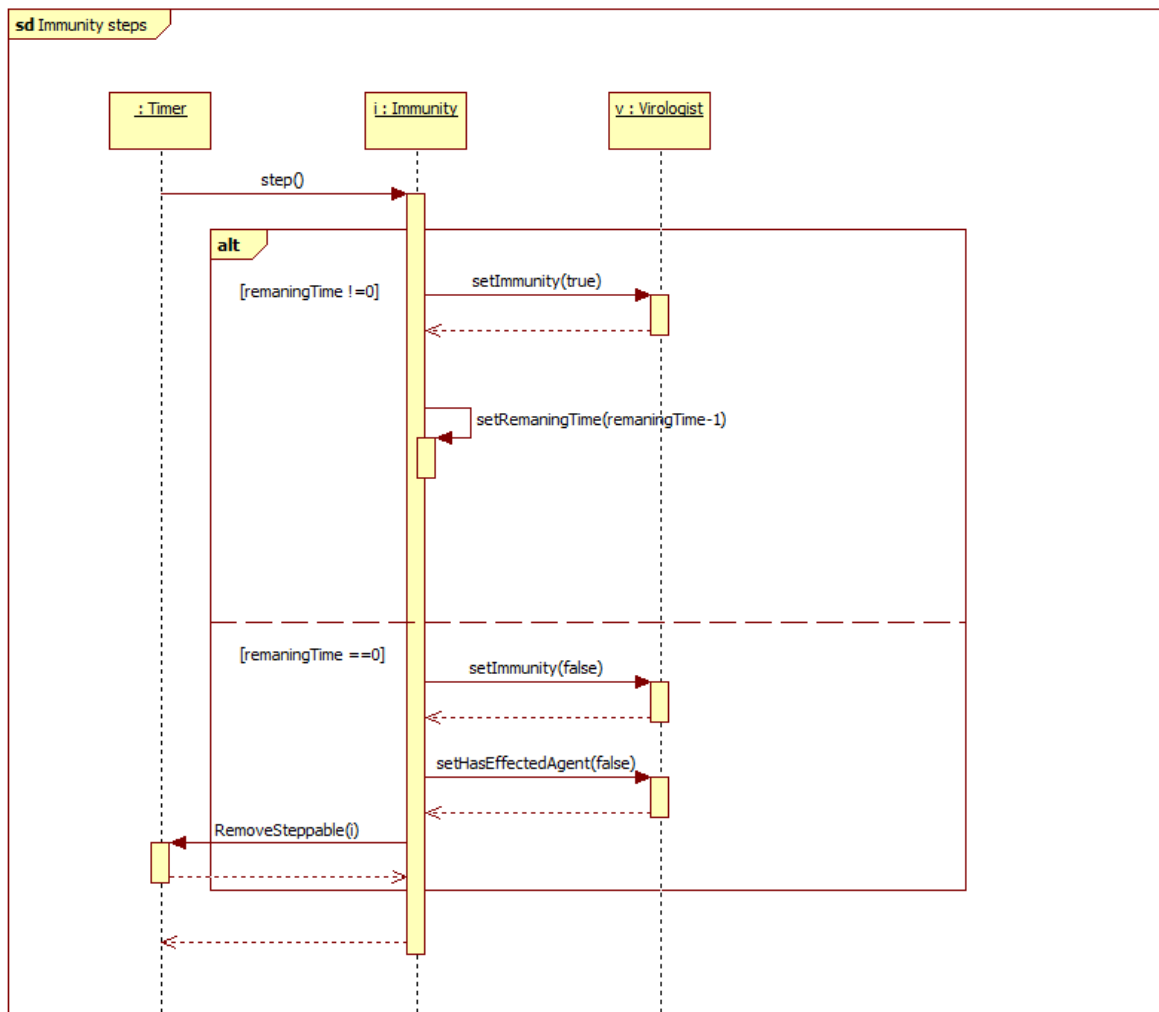
- **Felelősség**
 - A játék kezdetén a mezők kialakításáért, virológusok elhelyezéséért felel, majd a játék végén értesít minden virológust arról, hogy megnyerte vagy elvesztette a játékot.
- **Össztályok**
 - -
- **Interfészek**
 - -
- **Attribútumok**
 - **MaxGCode:** itt tároljuk azt, hogy hányféle kód létezik, melyet a játékosoknak a győzelem érdekében mind el kell sajátítaniuk.
 - **Fields:** a játékban létező mezők listája.
- **Metódusok**
 - **startGame():** A rendszer felépíti a mezőket, létrehozza a virológusokat, azokat elhelyezi kezdeti mezőjükön.
 - **endGame(v: Virologist):** A rendszer végigiterál az összes mezőn, a mezőktől elkéri a Virologists listáját, majd az összes virológusnak megjeleníti a játék végéről szóló értesítést, illetve, hogy az adott virológus nyert-e vagy nem. Ezután minden létező objektumot töröl.

4.4 Szekvencia diagramok

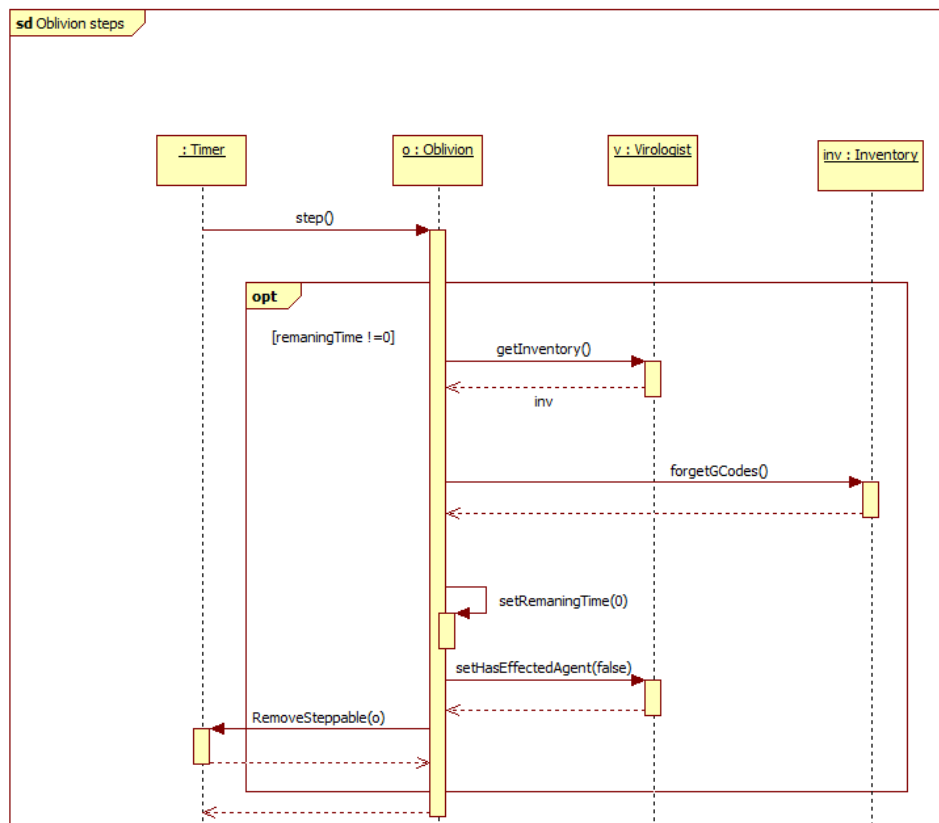
4.4.1 Timer steps



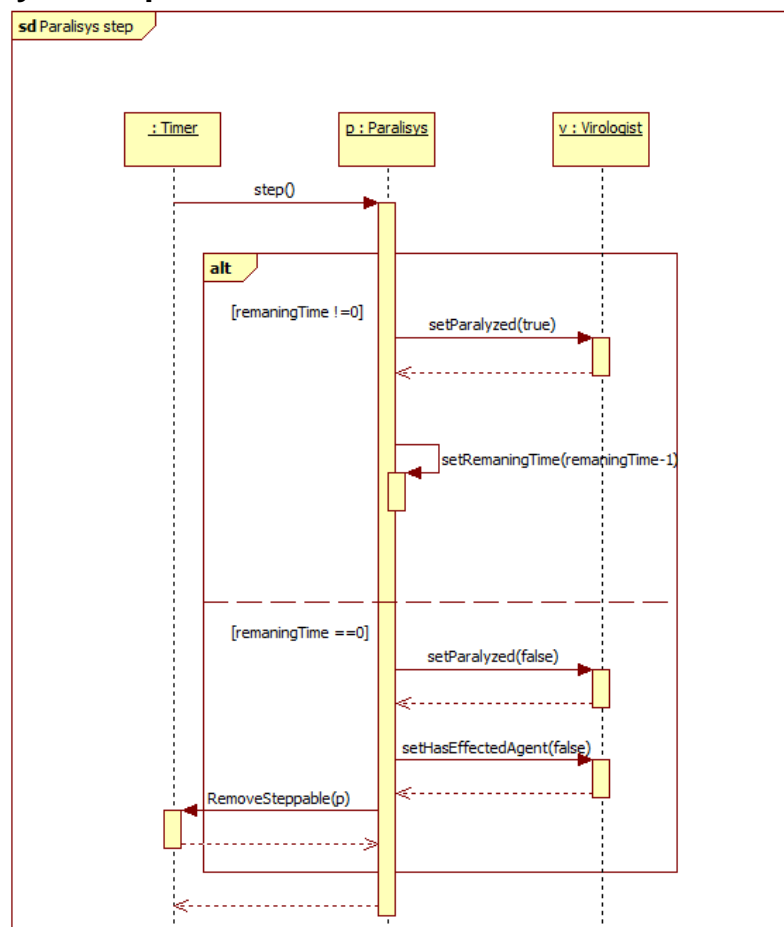
4.4.2 Immunity steps



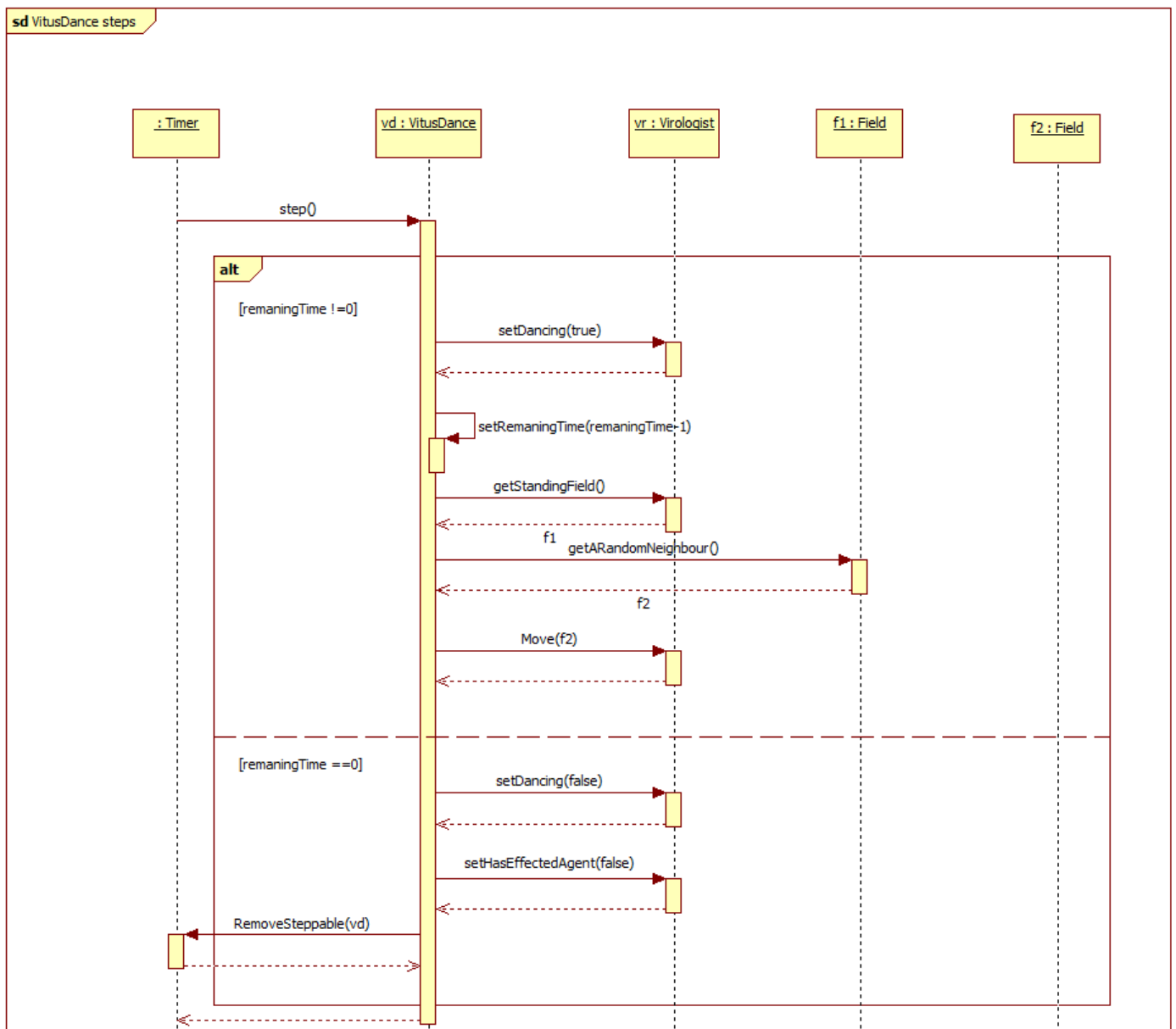
4.4.3 Oblivion steps



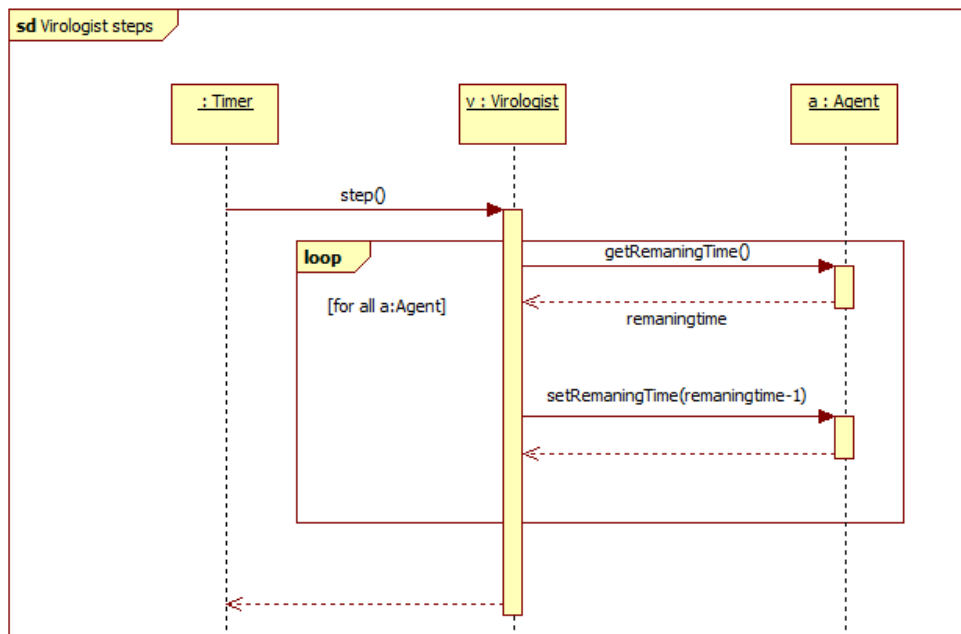
4.4.4 Paralysis steps



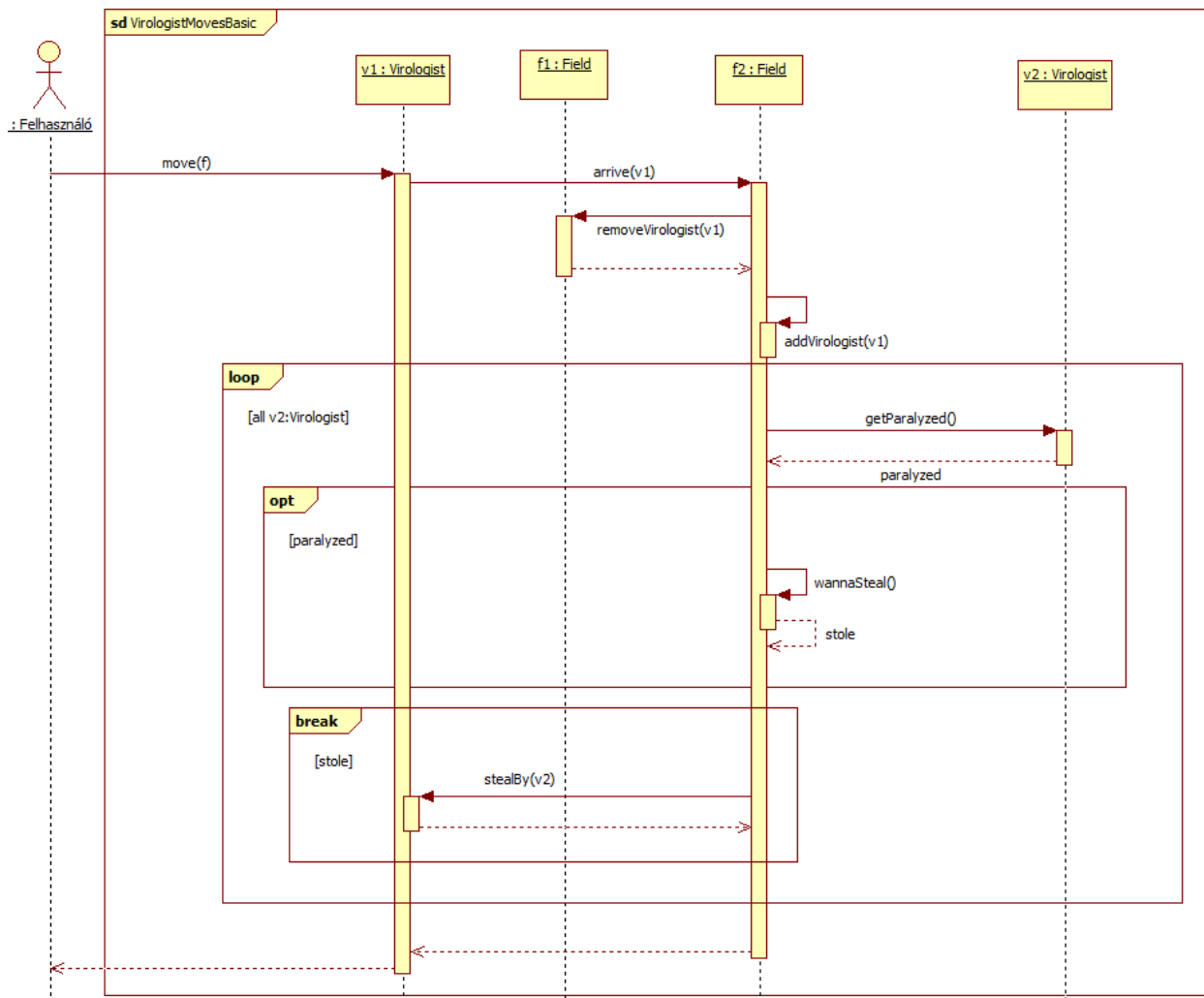
4.4.5 VitusDance steps



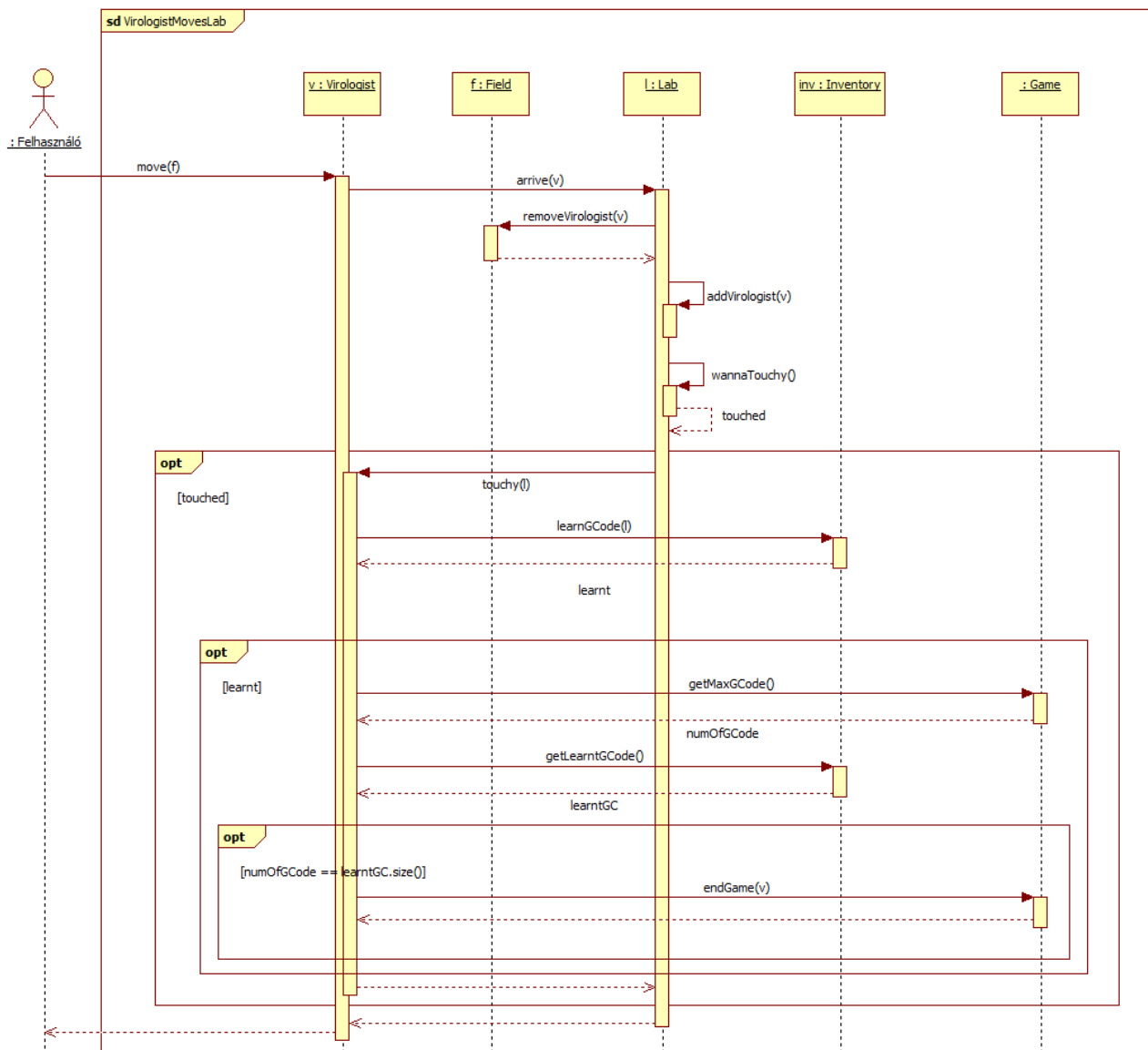
4.4.6 Virologist steps



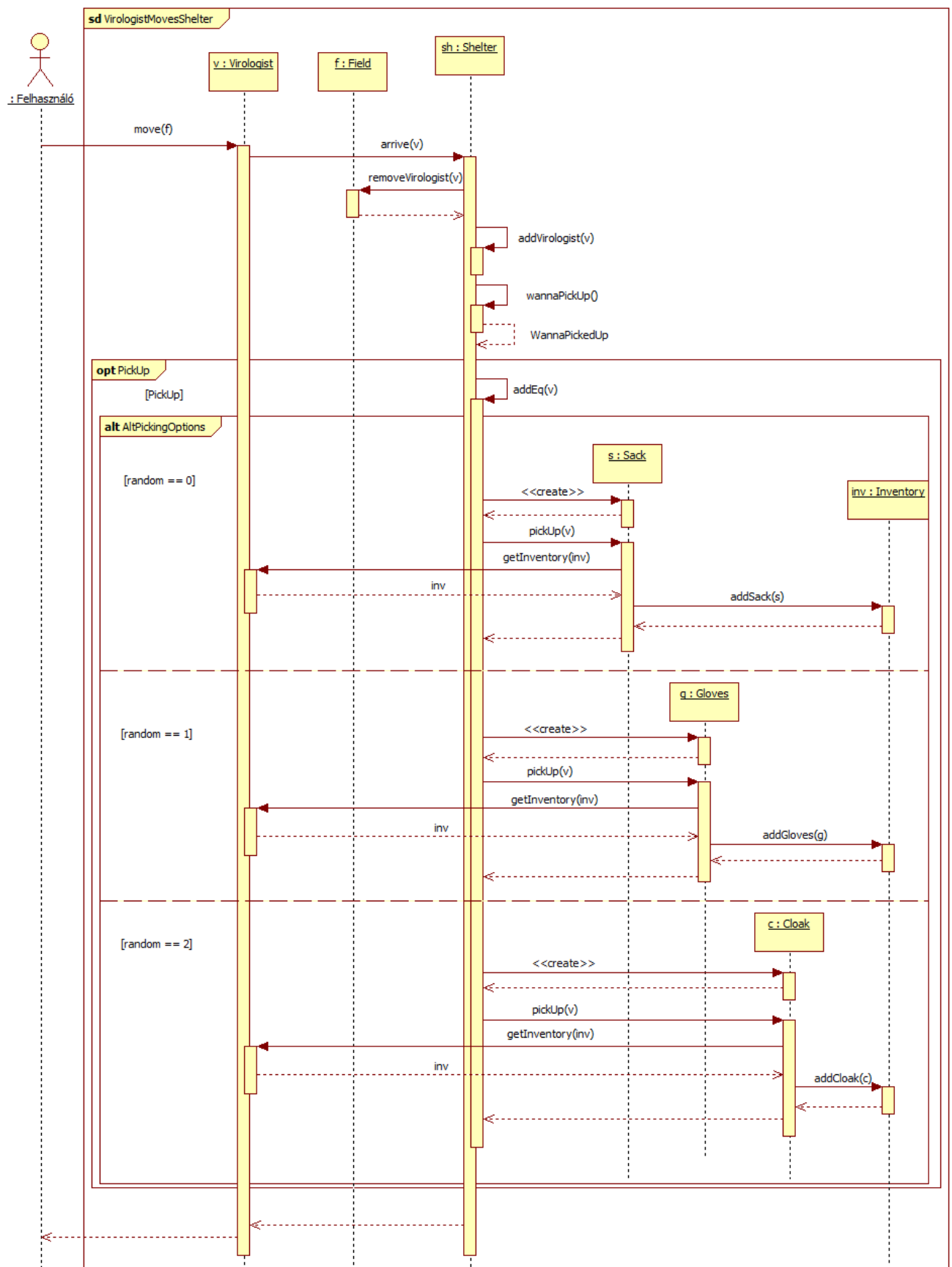
4.4.7 Virologist moves to a basic field



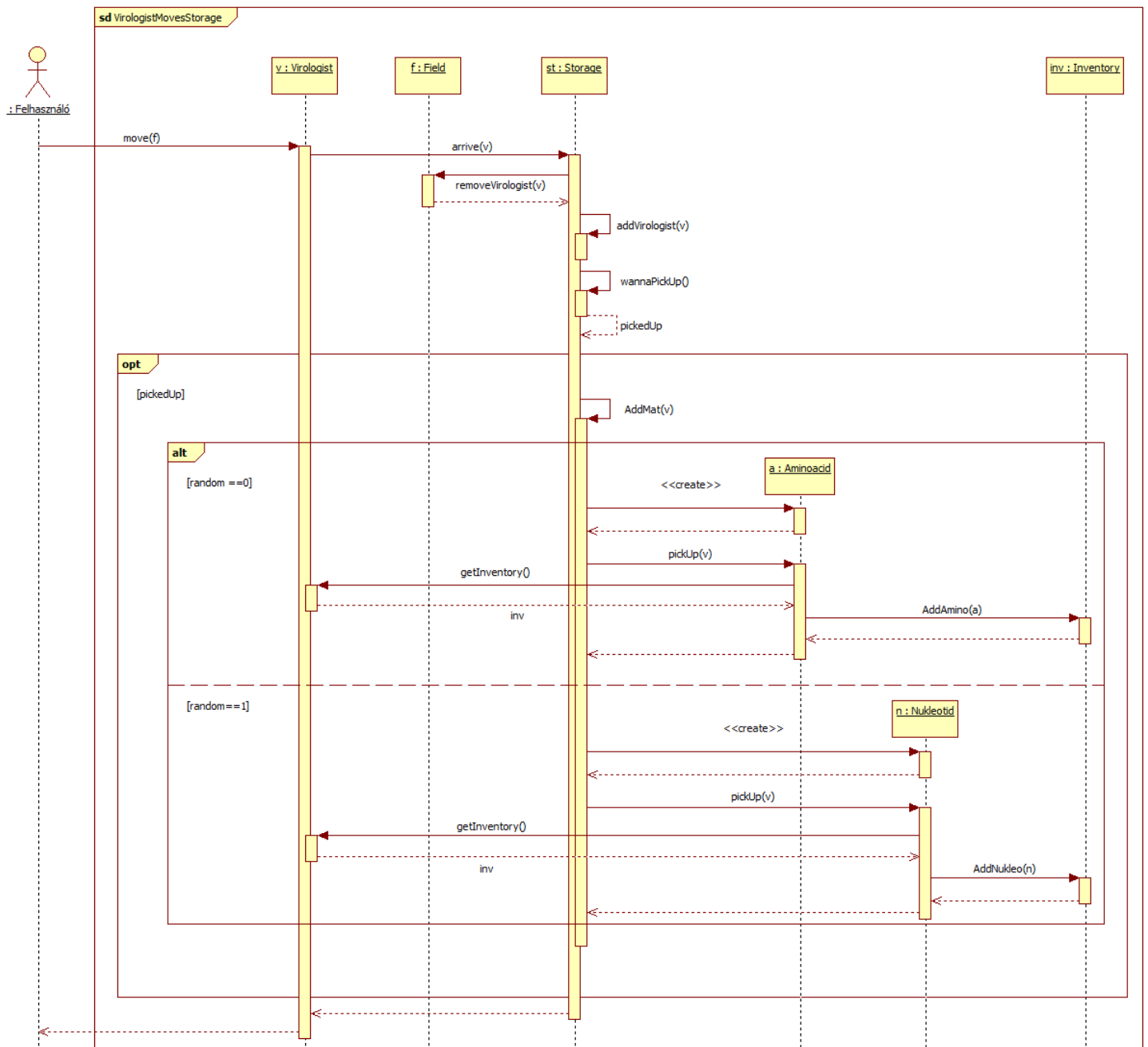
4.4.8 Virologist moves to a laboratory field



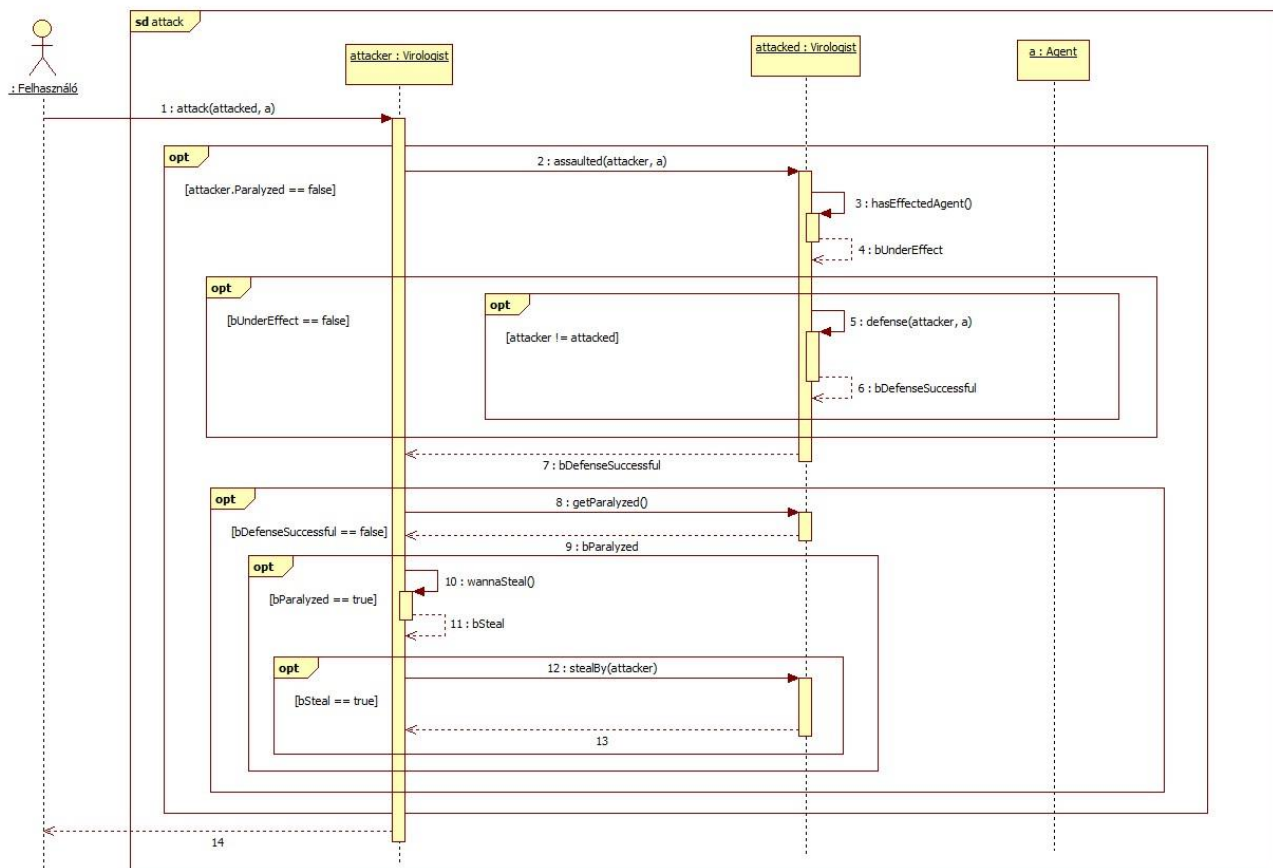
4.4.9 Virologist moves to a shelter field



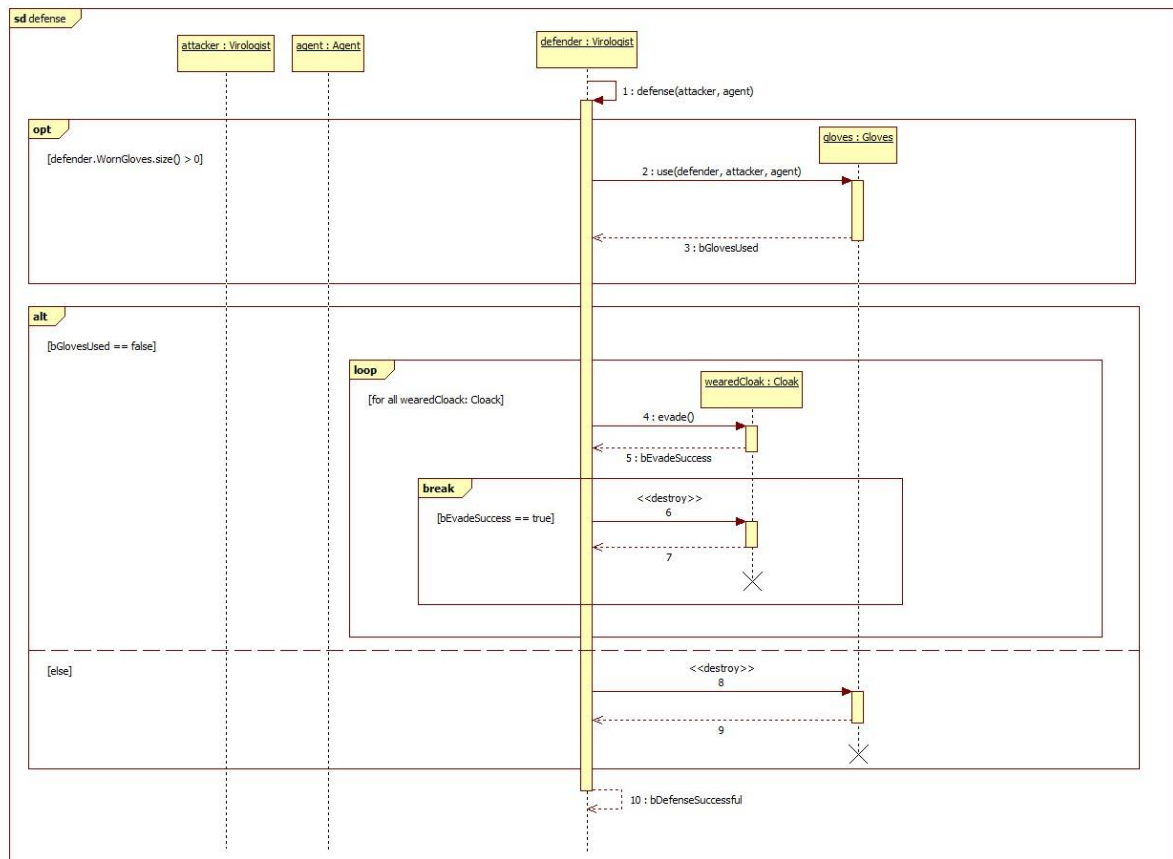
4.4.10 Virologist moves to a storage field



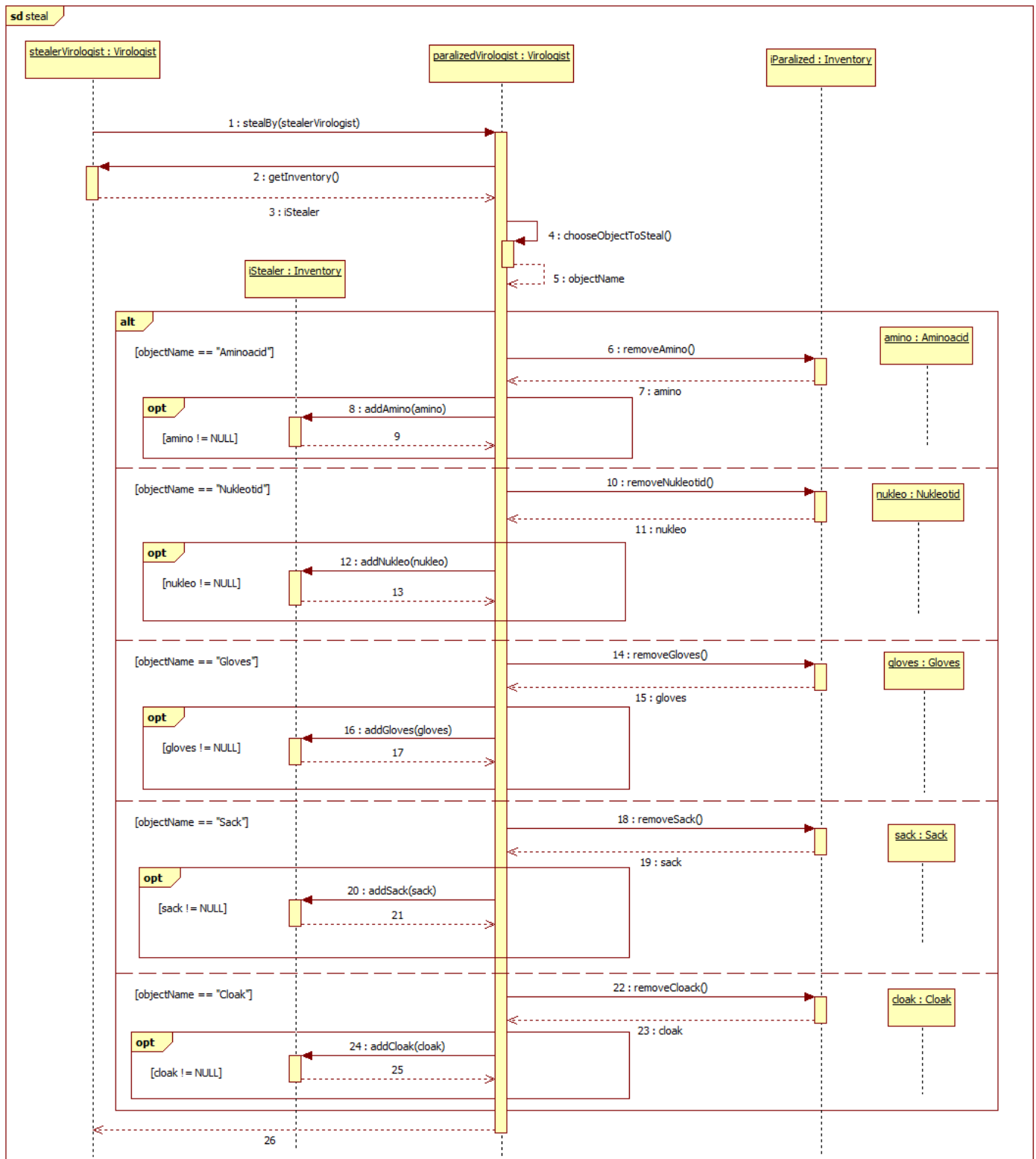
4.4.11 Attack



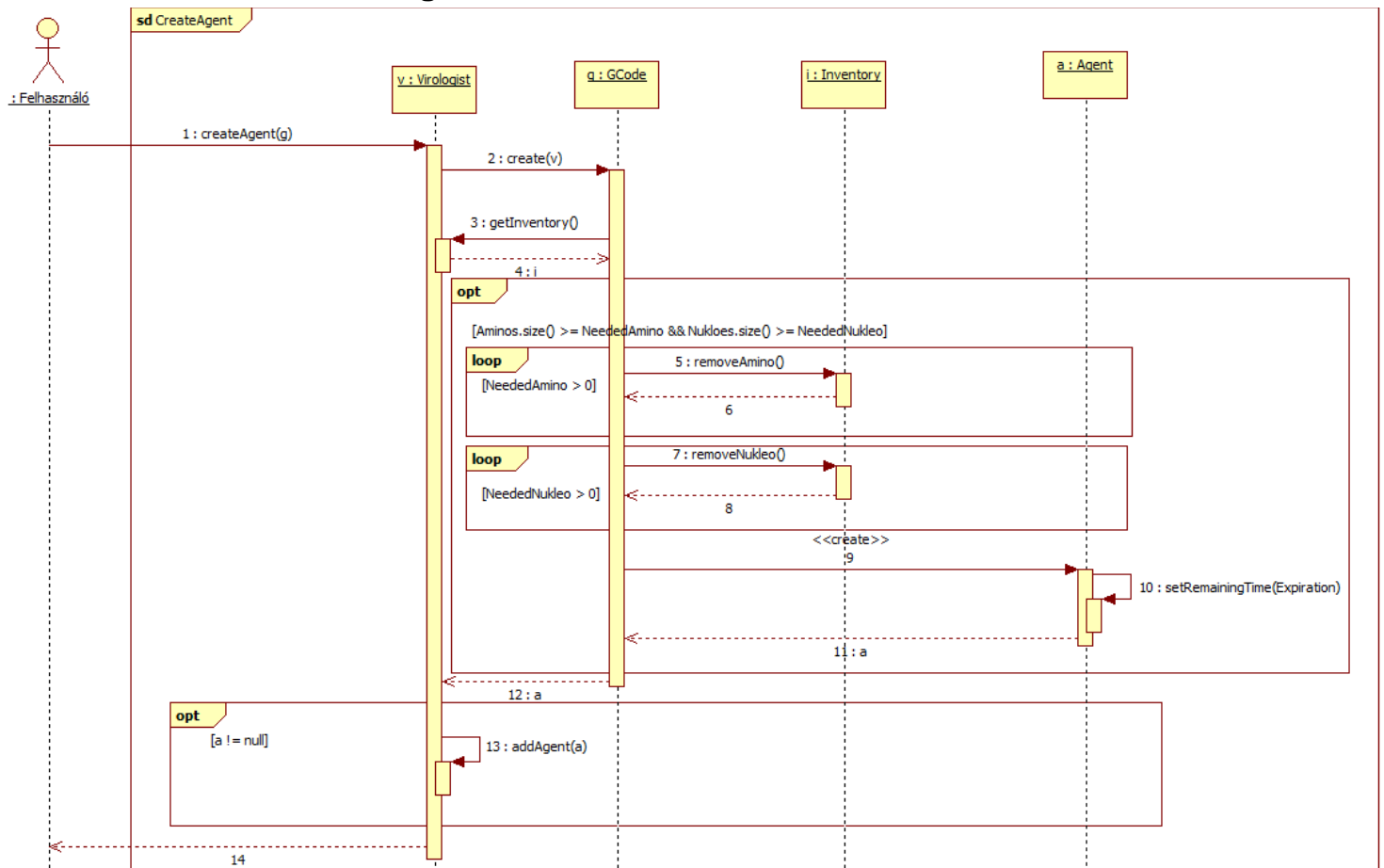
4.4.12 Defense



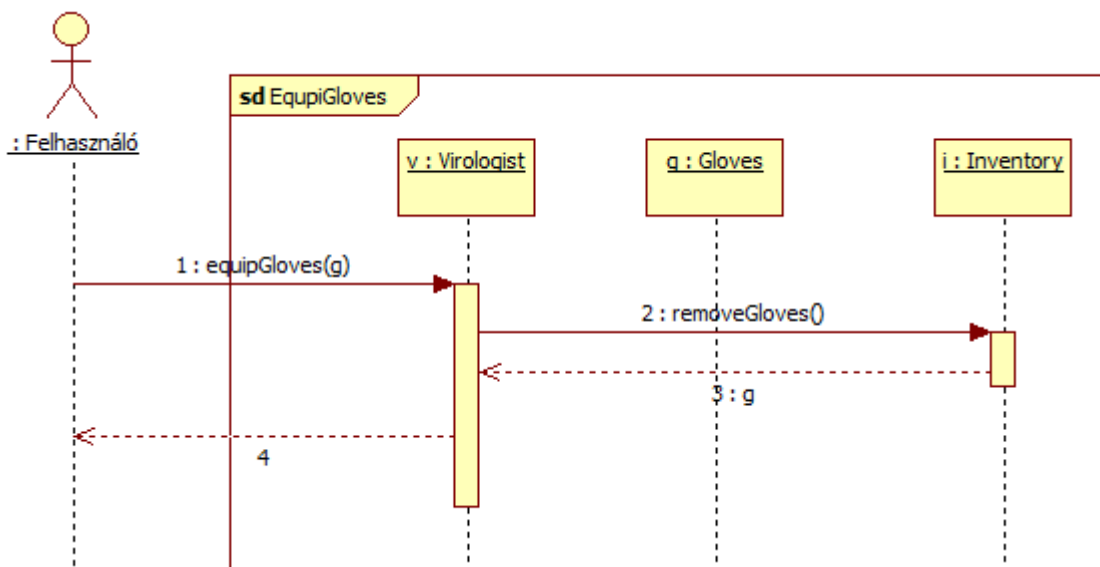
4.4.13 Steal



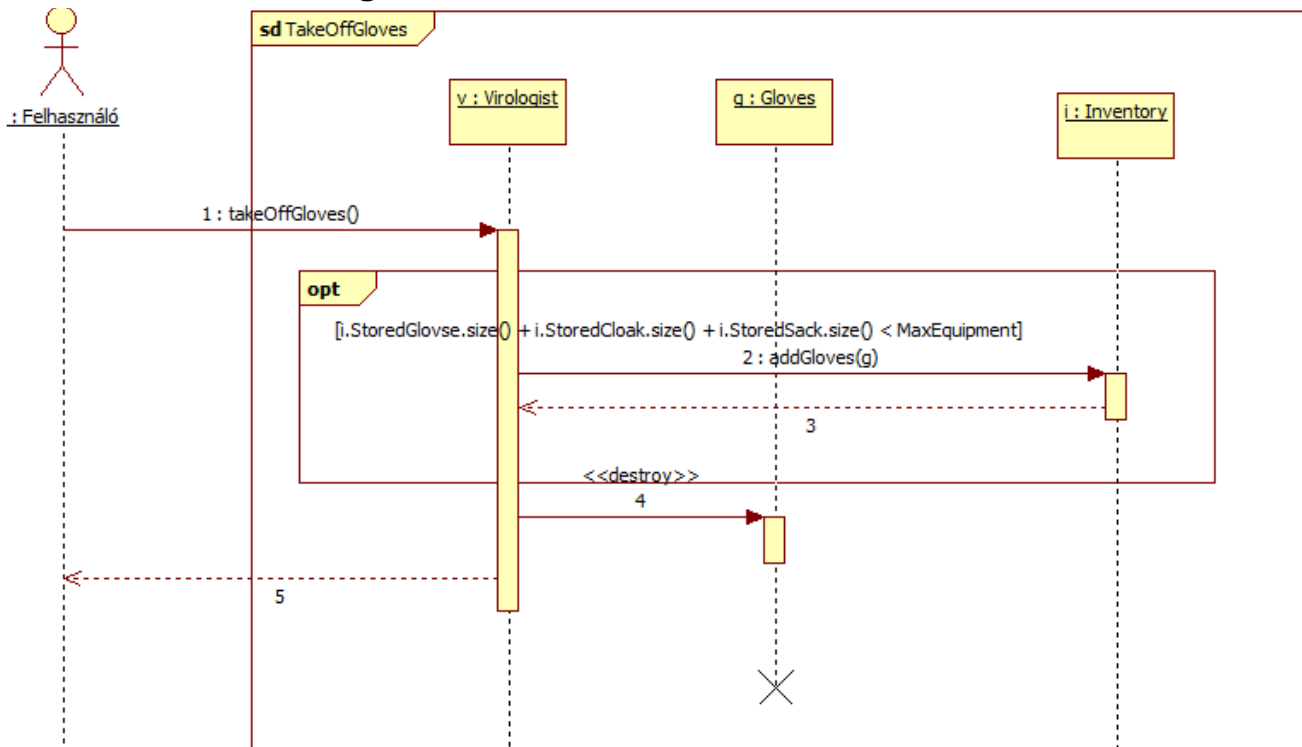
4.4.14 Create Agent



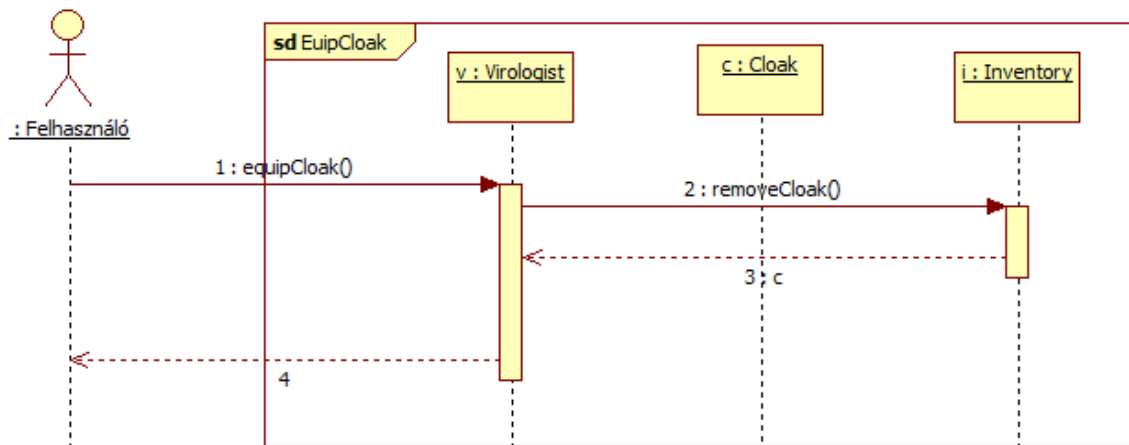
4.4.15 Equips gloves



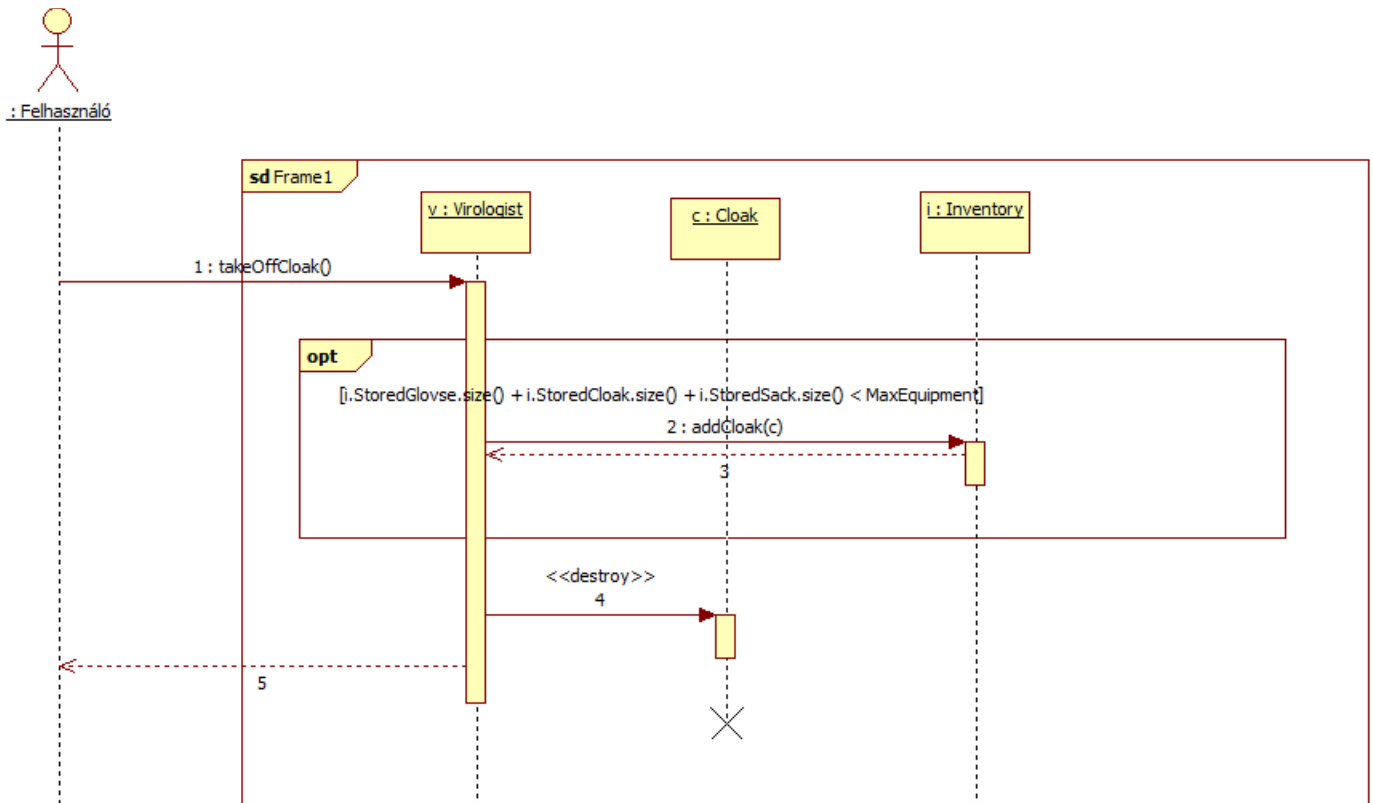
3.4.16 Takes off gloves



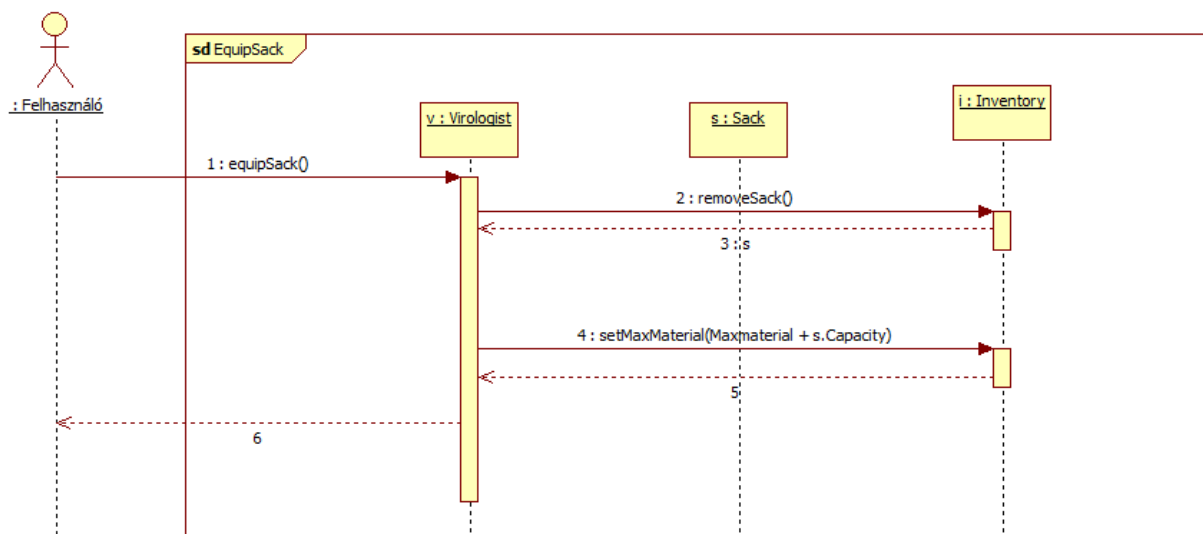
3.4.17 Equips cloak



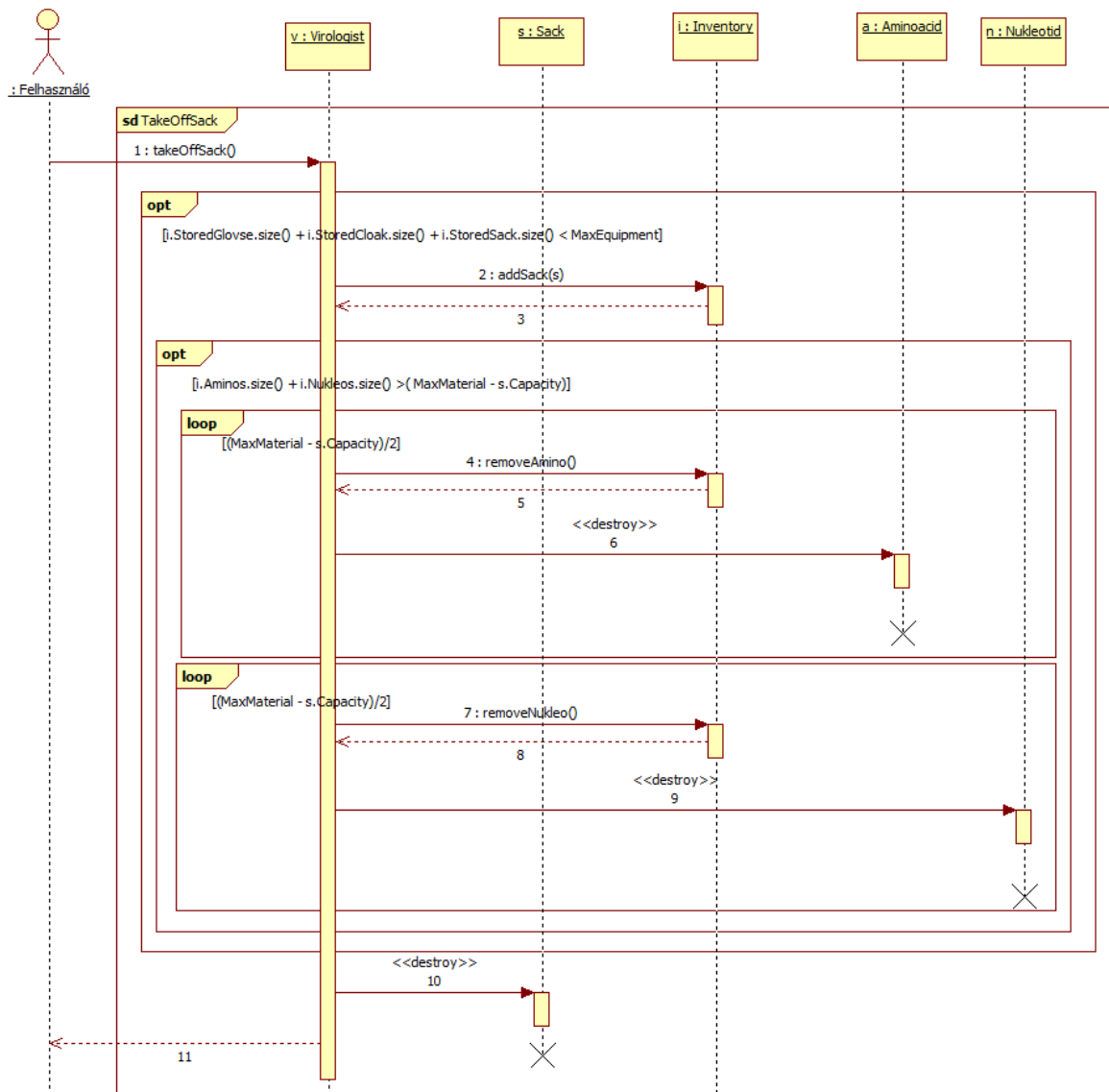
3.4.18 Takes off cloak



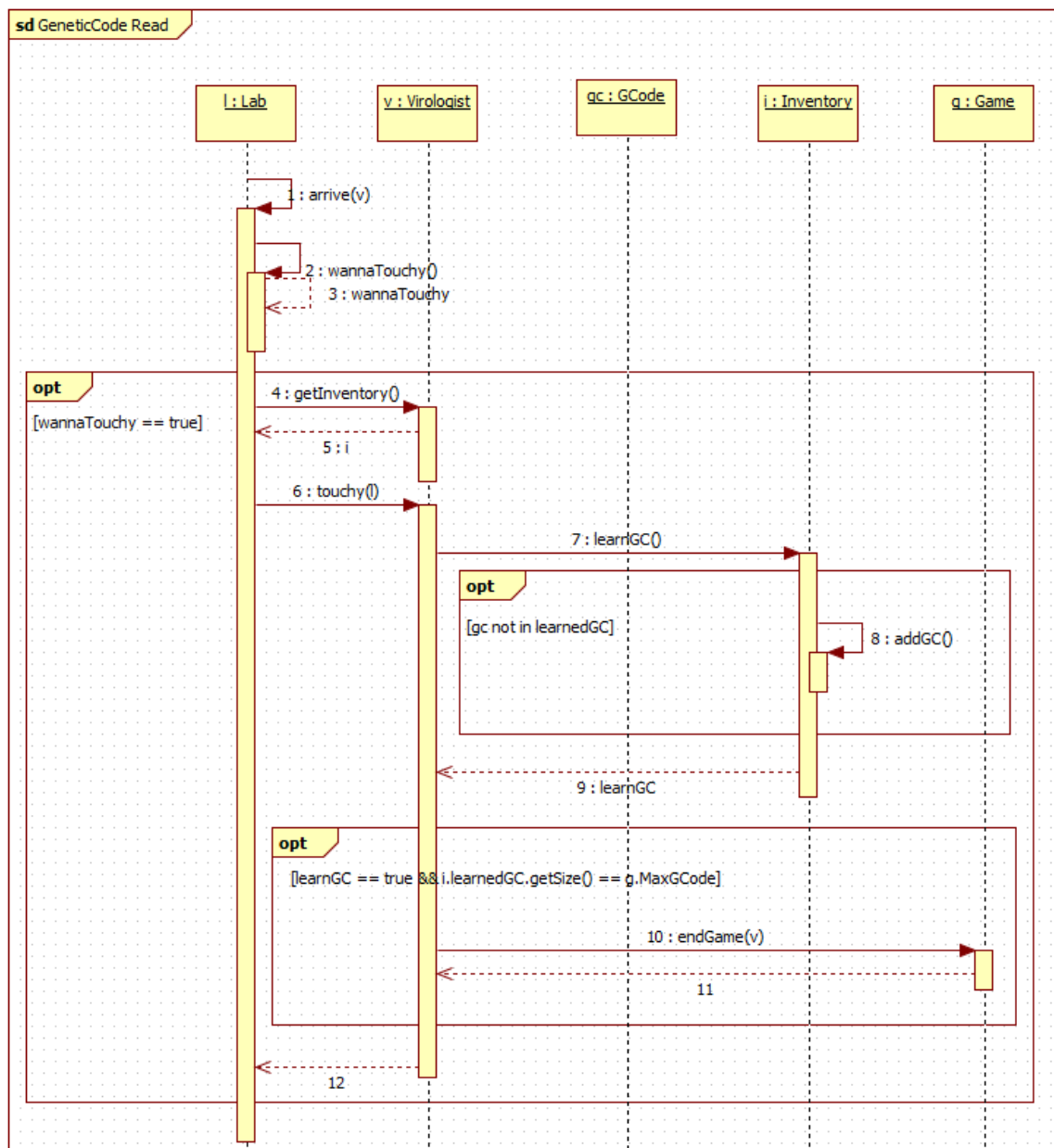
3.4.19 Equips sack



3.4.20 Takes off sack

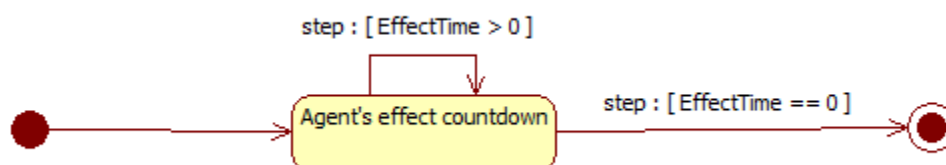


3.4.21 Genetic code read



3.5 Állapotgépek

3.5.1 Ágens effekt visszaszámláló



3.6 Ütemterv

Határidő	Feladat	Pont	Felelős
febr. 28.	Követelmény, projekt, funkcionalitás - beadás	10	Csia
márc. 7.	Analízis modell kidolgozása I. - beadás	20	Alpek
márc. 16.	Analízis modell kidolgozása II. - beadás	30	Marton
márc. 21.	Szkeleton tervezése - beadás	20	
márc. 28.	Szkeleton elkészítése - beadás	20	
ápr. 4.	Prototípus koncepciója – beadás Szkeleton bemutatás	20	
ápr. 11.	Részletes tervek - beadás	45	
ápr. 25.	Prototípus készítése, tesztelése	35	
máj. 2.	Grafikus változat tervei – beadás Prototípus - beadás és a forráskód, a tesztbemenetek és az elvárt kimenetek Herculesre való feltöltése	30	
máj. 16.	Grafikus változat készítése	40	
máj. 18.	Egységes dokumentáció - beadás és bemutatás Grafikus változat - beadás és a forráskód Herculesre való feltöltése, és teljes házi bemutatás	30	

3.7 Napló

Kezdet	Időtartam	Résztevő(k)	Leírás
2022.03.12 20:30	6 óra	Teljes csapat	<u>Értekezlet:</u> A konzultáción megbeszélte hibák átnézése. Közös ötletelés, hogy hogyan javítsuk ki ezeket.
2022.03.12 10:38	20 perc	Marton	<u>Feladat:</u> A következő konzultációig a feladatok szétosztása az emberek között, hogy kinek mi a dolga a saját határidőnkig.
2022.03.13 12:02	3 óra	Alpek	<u>Feladat:</u> Ágensgenerálás, illetve eszközök fel-le vételéhez szekvenciadiagram készítése, függvényleírások ennek megfelelően javítása.
2022.03.13 14:00	3 óra	Csia	<u>Feladat:</u> Az óvóhely és a labor mezők metódusaihoz szekvenciadiagram készítése, függvényleírások ennek megfelelően javítása. Stepper szekvenciadiagram javítása.
2022.03.13 15:34	3 óra	Litavec	<u>Feladat:</u> Az üres mező és a raktár mezők metódusaihoz, illetve a virológus step() metódusához (ágens effekt léptetéséhez) szekvenciadiagram készítése, függvényleírások ennek megfelelően javítása.
2022.03.13 11:27	3 óra	Marton	<u>Feladat:</u> Kesztyűhasználathoz szekvenciadiagram készítése, függvényleírások javítása, defense és attack szekvenciadiagram javítása.
2022.03.14 15:00	4 óra	Teljes csapat	<u>Értekezlet:</u> A mindenki által beadott szekvenciadiagrammok, és a hozzájuk tartozó függvények leírásának átnézése hibák keresése végett.
2022.03.14 20:00	1 óra	Csia	<u>Feladat:</u> Teljes dokumentum összefésülése, képek megigazítása, közös dokumentumba helyezése, helyesírási hibák keresése, dokumentum véglegesítése.

3.8 Százalékos teljesítés

Név	Százalék
Alpek	25%
Csia	25%
Litaveczi	25%
Marton	25%