

## Analízis modell kidolgozása 2.

83 – leopold\_

Konzulens:  
**Simon Balázs**

### Csapattagok

Blaskó Gergő	S4GTQC	blasko.gergo.hun@gmail.com
Ugrin Dániel	SU1U4C	ugrindaniel@gmail.com
Matkovics Gábor	LAXFOA	matgabor@vipmail.hu
Márin Krisztián	KDSVSG	marin.krisztian@gmail.com
Zwick Barnabás	PCWGZF	zwick.barnabas@gmail.com
Szabó Benedek	QY9C39	szabbeni@gmail.com

2016.03.14

1

## 4. Analízis modell kidolgozása 2

### 4.1 **Objektum katalógus**

#### 4.1.1 **Doboz**

*Egy speciális tárgy a doboz, amely blokkol, de nem ér pontot.*

#### 4.1.2 **Ajtó**

*Egy speciális mező, amely egy nyitható, zárható ajtót valósít meg.*

#### 4.1.3 **Tárgy**

*A Tárgy osztály példányai tárgyak, mely maguk kezelik megsemmisülésüket. Blokkolhatnak vagy pontot érhetnek.*

#### 4.1.4 **Szakadék**

*Egy speciális mező, a szakadék, amelyre ha Oneil, vagy doboz kerül, az megsemmisül.*

#### 4.1.5 **ZPM**

*Egy speciális tárgy a ZPM, amely nem blokkol és pontot ér.*

#### 4.1.6 **Mező**

*Az osztály példányai egyes mezőket valósítanak meg, akik kezelik hogyha Oneil rájuk akar lépni. Emellett, a Mező tartalmazhat egy tárgyat.*

#### 4.1.7 **Program**

*Az osztály tárolja az összes Mező, Oneil, és a Féreglyukak tulajdonságait.*

#### 4.1.8 **Mérleg**

*Egy speciális mező, a mérleg, mely amennyiben tárgy van rajta, nyitva tartja a hozzárendelt ajtót.*

#### 4.1.9 **Mozgás**

*Ha Oneil vagy a lövedéke mozog, akkor ennek a cselekvésnek a kezelése ezen az osztályon keresztül történik.*

#### 4.1.10 **Lövedék**

*Ha egy lövedék megsemmisül, akkor ez az osztály felelős annak a végrehajtásáért.*

#### 4.1.11 **Fal**

*Egy speciális tárgy, a fal, amely nem engedi, hogy rálépjenek.ű, illetve csillagkapu helyezhető rá.*

#### 4.1.12 **Portál**

*A csillagkapuk kinyitásáért, illetve bezárásáért felelős osztály.*

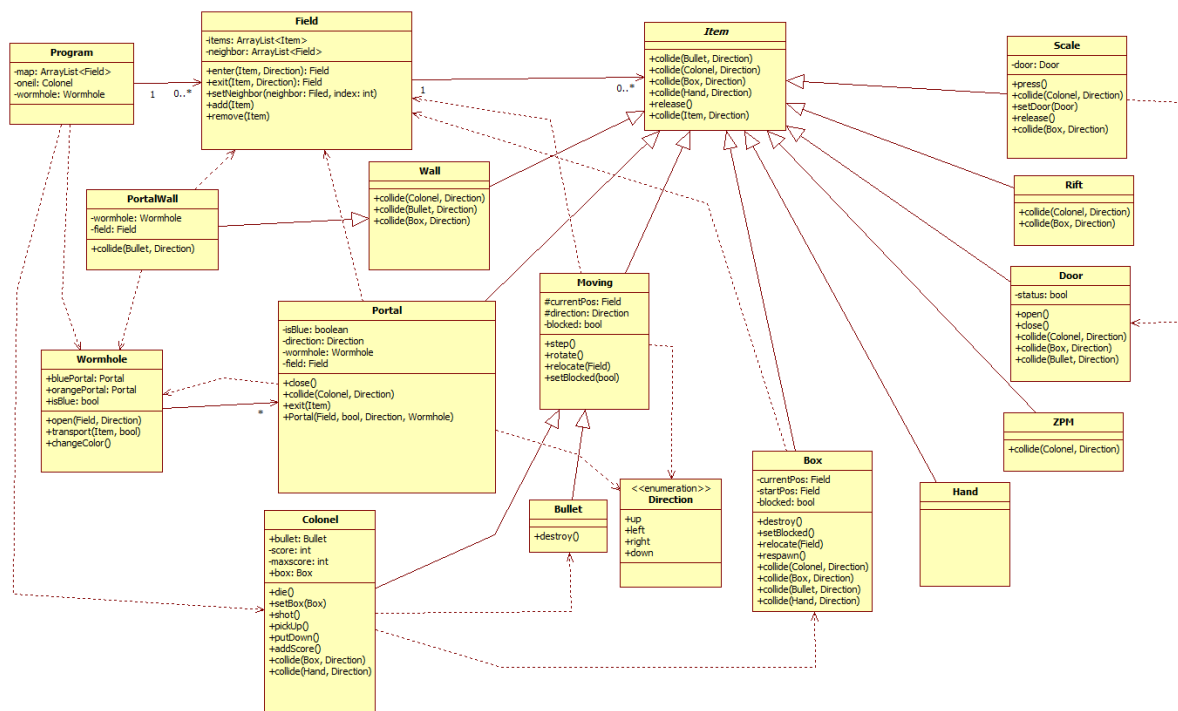
#### 4.1.13 **Féreglyuk**

*A csillagkaput színéhez megfelelően kell kezelniük. Ebben van segítségünkre a Féreglyuk osztály.*

## 4.1.14 Ezredes

Az Oneil osztály a játékost valósítja meg (egy speciális "tárgy"). Maga számolja pontszámát, avagy az összegyűjtött ZPM-eket, kezeli a csillagkapuk kilövését, valamint a saját halálát.

## 4.2 Statikus struktúra diagramok



## 4.3 Osztályok leírása

## 4.3.1 Box

## • Felelősség

A dobozok menedzseléséért felelős osztály.

## • Ősosztályok

Item  $\square$  Box

## • Interfészek

Nem valósít meg interfészt.

## • Attribútumok

- **private Field** currentPos

#### 4. Analízis modell kidolgozása 2

leopold\_

- **private Field** startPos
- **private bool** blocked
- **Metódusok**
  - void destroy(): Ha a doboz megsemmisül, ez a függvény hívódik meg a megsemmisülést okozó mezőtől.
  - void setBlocked(void): Átállítja hogy a doboz blokkol-e a mozgásban.
  - void relocate(Field f): A jelenlegi hely beállítása.
  - void respawn(void): Visszatérés a kiinduló mezőre.
  - void collide(Bullet b, Direction d): Ha a dobozt találat éri az adott irányból, akkor ez a függvény hívódik meg.
  - void collide(Box b, int Direction d): Ha a doboz egy másik dobozzal ütközik, ez a függvény hívódik meg.
  - void collide(Colonel c, Direction d): Az ezredessel való ütközés lekezelése.
  - void collide(Hand h, Direction d): A kézzel (felvétel/letevés) való érintkezést kezeli le. Az osztály nem írja felül többi örökölt függvényét.

##### 4.3.2 ZPM

- **Felelősség**

A pályán található ZPM-ek menedzseléséért felelős osztály.

- **Ősosztályok**

Item □ ZPM

- **Interfészek**

Nem valósít meg interfészt.

- **Attribútumok**

Nincsenek attribútumai.

- **Metódusok**

- void collide(Colonel Oneil, Direction d): Ha Oneil ZPM-hez jut az adott irányból, ez a függvény hívódik meg. Az osztály nem írja felül többi örökölt függvényét.

##### 4.3.3. Rift

- **Felelősség**

A szakadékok menedzseléséért felelős osztály.

- **Ősosztályok**

Item □ P1φτ

- **Interfészek**

Ez az osztály nem valósít meg interfészeket.

#### 4. Analízis modell kidolgozása 2

leopold\_

- **Attribútumok**  
Az osztálynak nincsenek attribútumai.
- **Metódusok**
- void collide(Colonel c, Direction d): Akkor hívódik meg, ha Oneil a szakadékba esik, Oneil halálát kezeli le.
- void collide(Box b, Direction d): Doboz szakadékba esését kezeli le.  
Az osztály nem írja felül többi örökölt függvényét.

##### 4.3.4 Door

- **Felelősség**

Az ajtók kezeléséért felelős osztály.

- **Ősosztályok**

Item □ Door

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**
- **private boolean** status
- **Metódusok**
- void open(): Az adott ajtó kinyitásáért felelős függvény.
- void close(): Az adott ajtó bezárásáért felelős függvény.
- void collide(Bullet b, Direction d): Lövedékkal való ütközés lekezelése, ha ajtó nyitva van, akkor áthalad, egyébként megáll.
- void collide(Colonel c, Direction d): Az ezredes átmenetét lekezelő függvény, amely csak akkor engedi át az ezredest, ha az ajtó nyitva van.
- void collide(Box b, Direction d): Doboz mezőre való elhelyezésének lekezelése.  
Az osztály nem írja felül többi örökölt függvényét.

##### 4.3.5 Field

- **Felelősség**
- **Ősosztályok**

Az osztálynak nincsenek ősosztályai.

- **Interfészek**

Az osztálynak nincsenek interfészei.

- **Attribútumok**
- **private ArrayList<Item>** items
- **private ArrayList<Field>** neighbor
- **Metódusok**
- Field enter(Item i): Ha egy tárgy kerül az adott mezőre, akkor ez a függvény hívódik meg.
- Field exit(Item i): Ha egy tárgy kerül le az adott mezőről, akkor ez a függvény hívódik meg.

#### 4. Analízis modell kidolgozása 2

leopold\_

- void setNeighbor(Field neighbor,int index): Minden egyes mezőnek ismernie kell a szomszédait. Ezen szomszédok beállításáért felelős ez a függvény.
- add(Item i): A mezőre érkező tárgy elmentése.
- remove(Item i): A mezőről távozó tárgy törlése a listából.

##### 4.3.6 Moving

- **Felelősség**

Ha az ezredes, vagy a lövedék mozog, akkor az ezen osztályon keresztül történik.

- **Ősosztályok**

Item-Moving

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**
  - **protected Field** currentPos
  - **protected Direction** direction
  - **private bool** blocked
  - **Metódusok**
  - void step(void): A mezőkön való áthaladásért felelős függvény.
  - void rotate(void): A mezőkön való megfordulásért felelős függvény.
  - void relocate(Field f): Másik mezőre való áthelyeződést lekezelő függvény.
- Az osztály nem írja felül többi örökölt függvényét.

##### 4.3.7 Bullet

- **Felelősség**

A lövedékek menedzseléséért felelős osztály.

- **Ősosztályok**

Item-Moving-Bullet

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**
  - **protected Field** currentPos
  - **protected int** direction
- Az osztálynak nincsenek saját egyedi változói.
- **Metódusok**
  - void destroy(): Ha a lövedék becsapódik, akkor lényegében megsemmisül. Ha megsemmisül, akkor azért ez a függvény a felelős.
- Az osztály nem írja felül többi örökölt függvényét.

#### 4.3.8 Item

- **Felelősség**

Mindenféle ütközésért felelős osztály.

- **Ősosztályok**

Az osztálynak nincsenek ősosztályai.

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**

Az osztálynak nincsenek attribútumai.

- **Metódusok**
- void collide(Bullet b,Direction d): Ha egy tárgy vagy az ezredes adott irányból lövedékekkel ütközik, akkor ez a függvény hívódik meg.
- void collide(Colonel c,Direction d): Ha egy tárgy az ezredessel ütközik adott irányból, akkor ez a függvény hívódik meg.
- void collide(Box b,Direction d): Ha egy tárgy vagy az ezredes a dobozzal ütközik adott irányból, akkor ez a függvény hívódik meg.
- void collide(Hand h,Direction d): A tárgyak és az ezredes kezének adott irányban végbemenő interakciójának lekezelése.
- void release(): Akkor hívódik meg, ha a tárgyon nincsen más tárgy. Ez a függvény a azt teszi lehetővé, hogy a mérlegnek ugyanilyen nevű függvénye meghívható legyen a heterogén kollekcióból is.
- void collide(Item i,Direction d): Tárgy-tárgy interakció lekezelése adott irányban.

#### 4.3.9 Wall

- **Felelősség**

A játék során felhasznált falak ősosztálya. Ez az osztály továbbá azokat falakat tartalmazza, amikre nem lehet portált nyitni.

- **Ősosztályok**

Item-Wall

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**  
**Az osztálynak nincsenek attribútumai.**
- **Metódusok**
- void collide(Bullet b,Direction d): A golyó falba csapódásának függvénye, megsemmisíti a golyót hatás nélkül.
- void collide(Colonel c,Direction d): Az ezredes falba ütközésének függvénye, meggátolja a falra lépést.
- void collide(Box b,Direction d): Dobozok falba ütközését lekezelő függvény.

#### 4. Analízis modell kidolgozása 2

leopold\_

Az osztály nem írja felül többi örökölt függvényét.

##### 4.3.10 Portal

- **Felelősség**

A portálfalak menedzseléséért felelős osztály.

- **Ősosztályok**

Item□Portal

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**
- **private boolean** isBlue
- **private Direction** direction
- **private Wormhole** wormhole
- **private Field** field
- **Metódusok**
- void close(): Ha egy portál bezárul, akkor arról ez a függvény tehet.
- void collide(Colonel c, Direction d): Az ezredes portálra lépését vezérlő metódus.
- void exit(Item i): Egy tárgy portálból való kilépésének metódusa.

Az osztály nem írja felül többi örökölt függvényét.

##### 4.3.11 Colonel

- **Felelősség**

Az ezredes menedzseléséért felelős osztály.

- **Ősosztályok**

Item-Moving-Colonel

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**
- **public Bullet** bullet
- **private int** score
- **private int** maxscore
- **private Box** box
- **public Bullet** bullet
- **Metódusok**
- void die(): Ha az ezredes meghal, ezt a függvényt hívja meg a halált okozó mező.
- void setBox(Box b): A dobot felvételekor ez a függvény menti el a dobozt.
- void shot(): Ha a játékos lőni szeretne, akkor ez a függvény hívódik meg. Külső behatás (a játékos) hatására hívódik meg.
- void pickUp(): Ha az ezredes egy tárgyat vesz fel, akkor ez a függvény hívódik meg.
- void putDown(): Ha az ezredes egy tárgyat tesz le, akkor ez a függvény hívódik meg.



#### 4. Analízis modell kidolgozása 2

*leopold\_*

- `void addScore()`: Ha az ezredes sikeresen elér egy ZPM-et, akkor ezért pontot kap, amit a játék el is tárol. Továbbá ellenőrzi, elérte-e az ezredes a maximum pontszámot, ami a játék megnyeréséhez szükséges.
  - `void collide(Box b, Direction d)`: Ha az ezredes egyik irányban dobozzal ütközik, ez a metódus hívódik meg.
  - `void collide(Hand h, Direction d)`: Az ezredesnek és saját kezének ütközését irányító függvény..
- Az osztály nem írja felül többi örökölt függvényét.

##### 4.3.12 **Wormhole**

- **Felelősség**
- **Ősosztályok**

Az osztálynak nincsenek őszosztályai.

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**
- **public Portal** bluePortal
- **public Portal** orangePortal
- **public bool** isBlue
- **Metódusok**
- `void open(Field f, Direction d)`: Kinyitja a kapott paraméterek alapján az adott mezőre az adott irányba a féreglyukat.
- `void transport(Item i, bool b)`: A portálon való áthaladást levezérlő függvény.
- `void changeColor()`: A színváltás metódusa.

##### 4.3.13 **Program**

- **Felelősség**

A játék működéséhez és sikeres elindulásához szükséges osztály.

- **Ősosztályok**

Az osztálynak nincsenek őszosztályai.

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**
- **private ArrayList<Field>** map
- **private Colonel** oneil
- **private Wormhole** wormhole
- **Metódusok**
- `static void main(String args[])`: Megvalósítja a játékhoz szükséges osztályokat.

##### 4.3.14 **Scale**

#### 4. Analízis modell kidolgozása 2

leopold\_

- **Felelősség**

A mérleg menedzseléséért felelős osztály.

- **Össztályok**

Item-Scale

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**
- **private Door** door
- **Metódusok**
- void press():Ha az ezredes rálép a mérlegre, vagy tárgy kerül rá ez a függvény hívódik meg.
- void release():Ha az ezredes lelép a mérlegről, vagy tárgy kerül le róla ez a függvény hívódik meg.
- void collide(Colonel c,Direction d): Az ezredes mérlegre lépését levezénylő metódus.
- void collide(Box b,Direction d): A mérlegre doboz helyeződés funkciója.
- void setDoor(Door d): A metódus, ami beállítja, mely ajtót fogja a mérleg irányítani.

#### 4.3.15 Hand

- **Felelősség**

Az ezredes kezét menedzseléséért felelős osztály.

- **Össztályok**

Item-Hand

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**
- **Nincs attribútuma.**
- **Metódusok**
- Az osztály nem írja felül örökölt függvényt.

#### 4.3.16 PortalWall

- **Felelősség**

Azon falak osztálya, amelyekre portál nyitható.

- **Össztályok**

Item-Wall-PortalWall

- **Interfészek**

Az osztály nem valósít meg egyetlen interfészt sem.

- **Attribútumok**
- **private Wormhole** wormhole

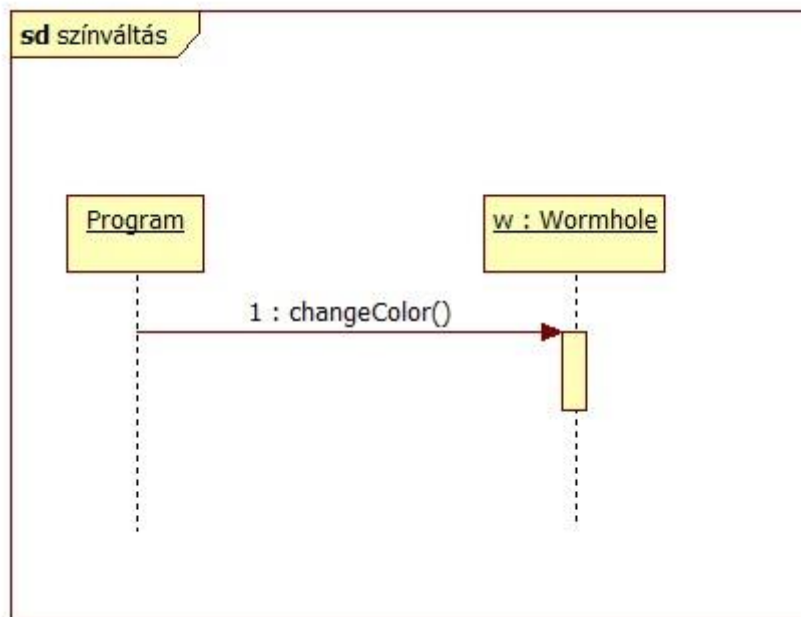
#### 4. Analízis modell kidolgozása 2

leopold\_

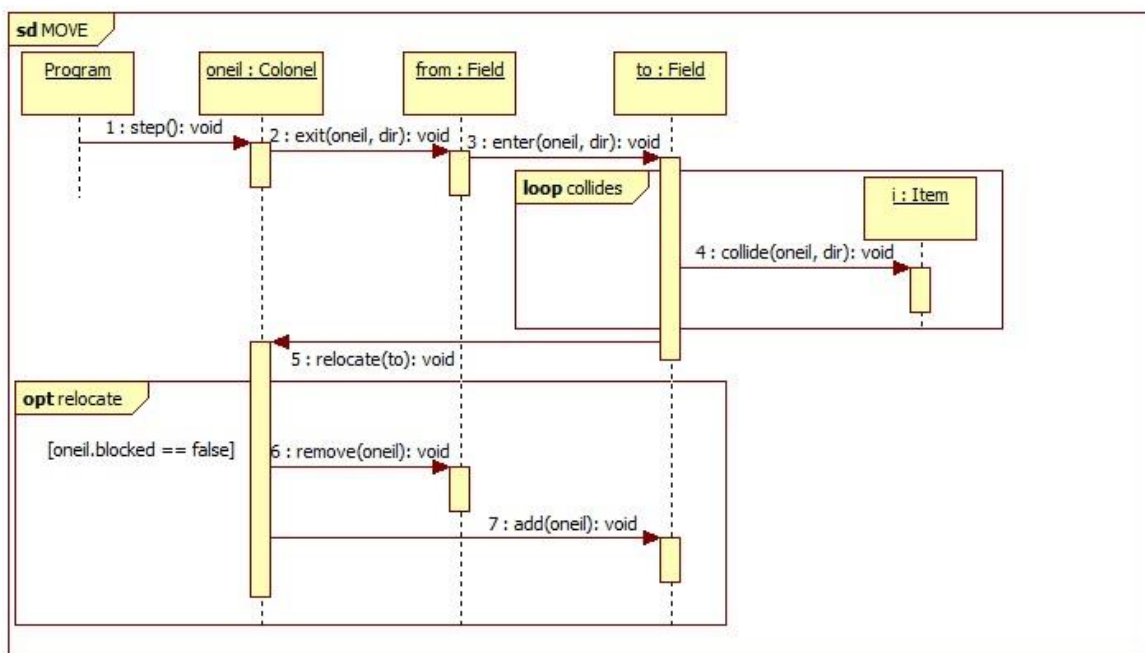
- **private Field** field
- **Metódusok**
- void collide(Bullet b, Direction d): A portálfalra lőtt golyót lekezlő függvény. Az osztály nem írja felül többi örökölt függvényét.

#### 4.4 Szekvencia diagramok

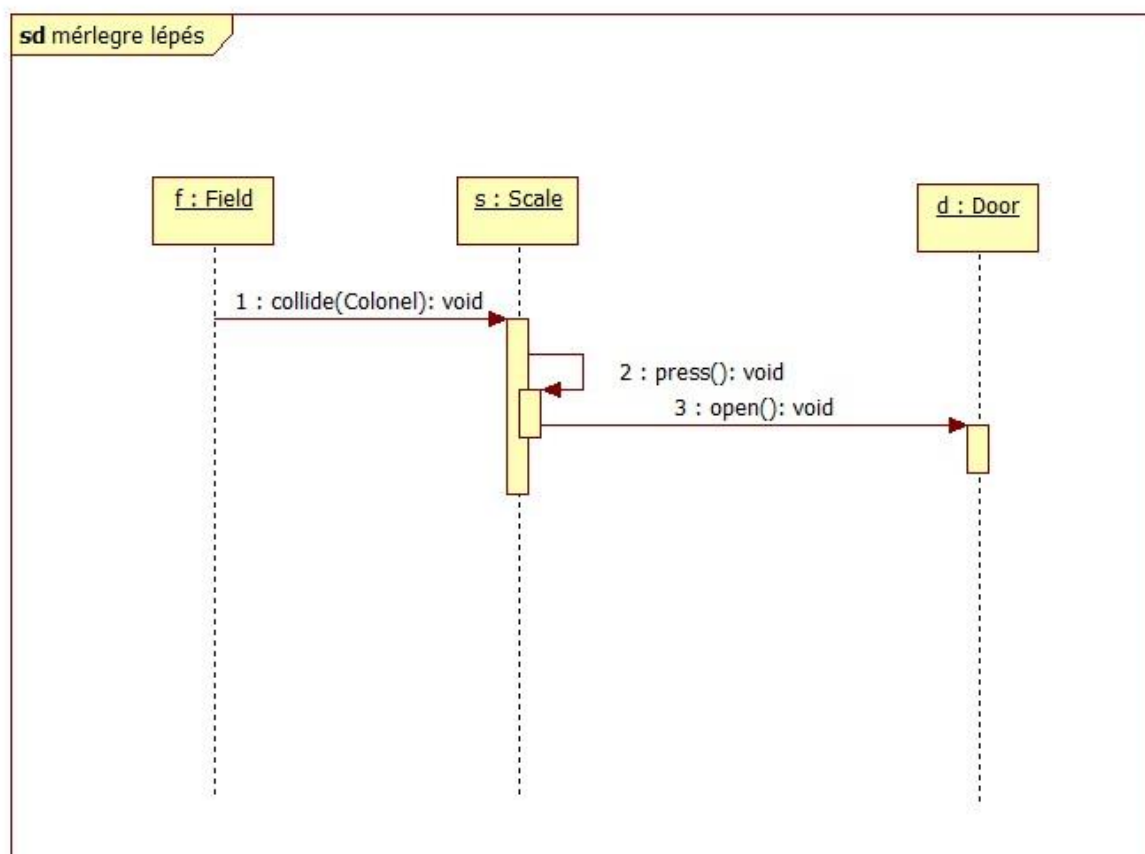
“színváltás”- a játékos megváltoztatja a kilövendő lövedék színét.



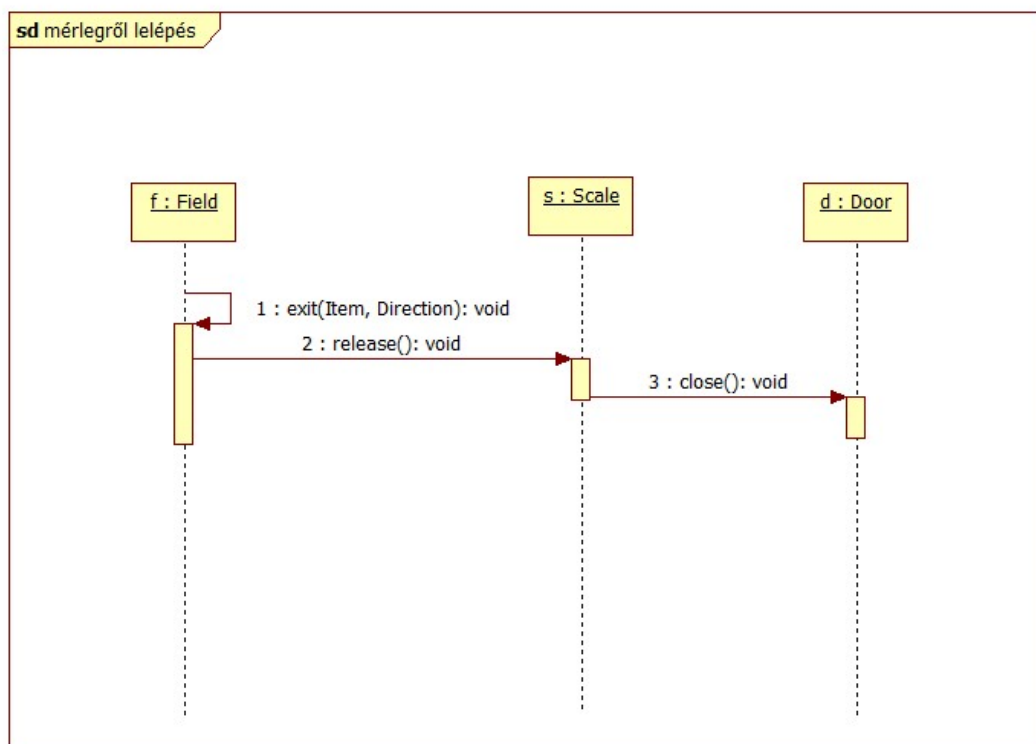
“MOVE”-Az ezredes mozog mezőről mezőre, miközben nem ütközik akadályba.



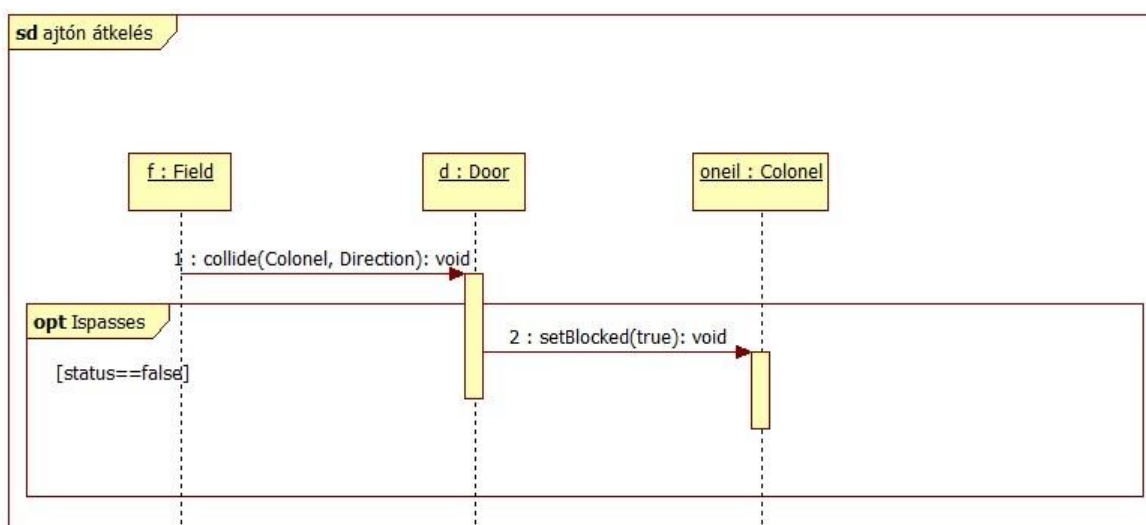
“**mérlegre lépés**”-Az ezredes rálép egy mérlegre, és a hozzátartozó ajtó kinyílik.



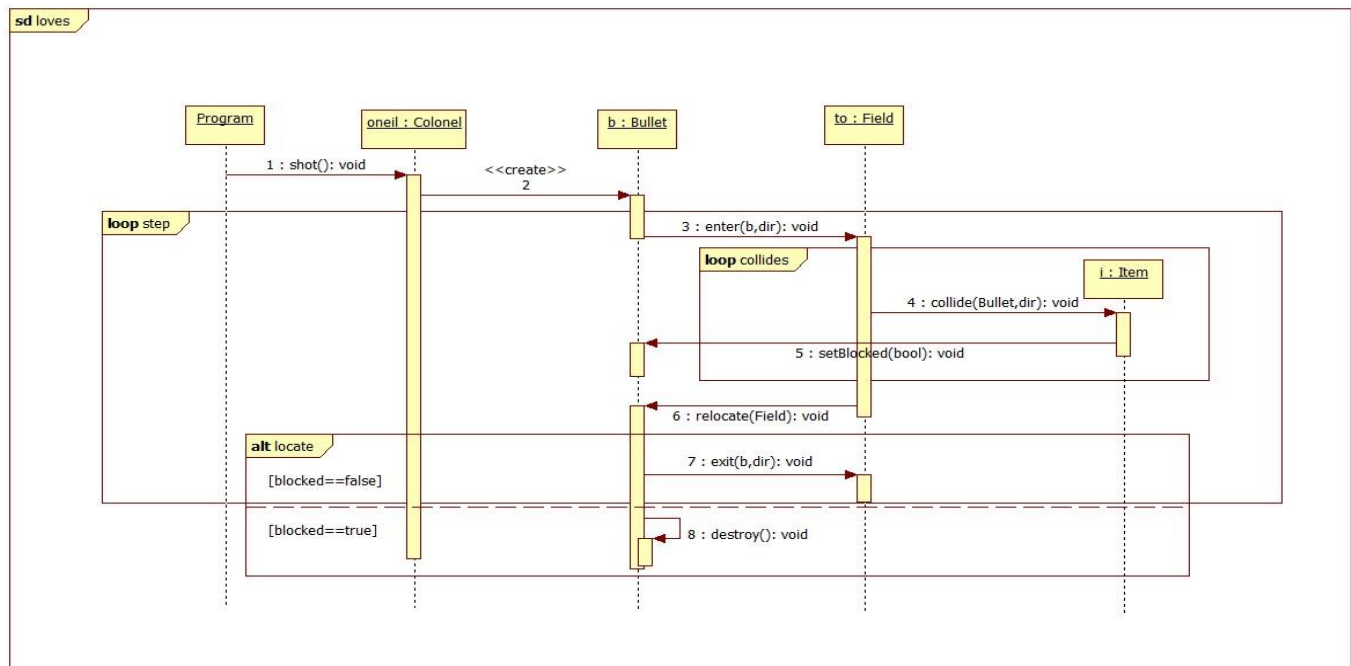
“**mérlegről lelépés**”-Az ezredes lelép egy mérlegről, így a hozzátartozó ajtó becsukódik.



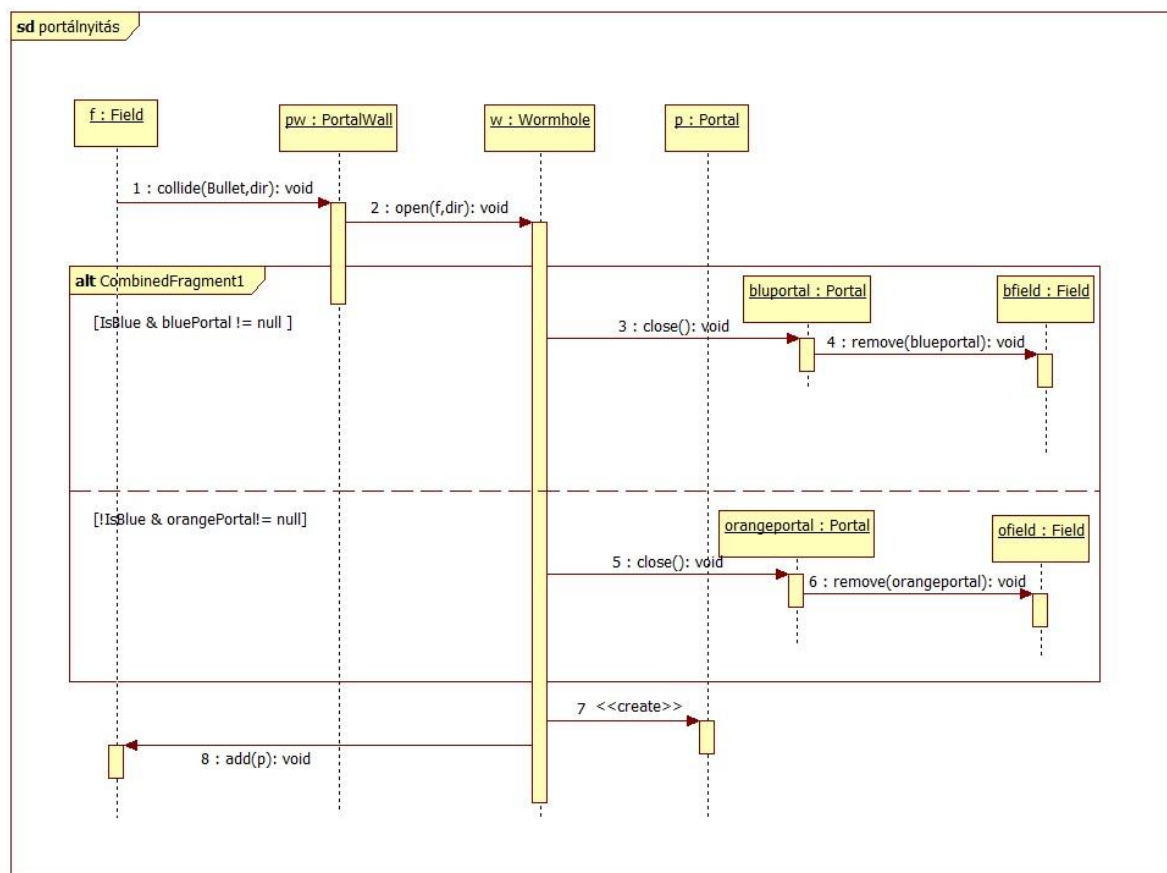
“**ajtón átkelés**”-Az ezredes megpróbál átkelni egy ajtón.



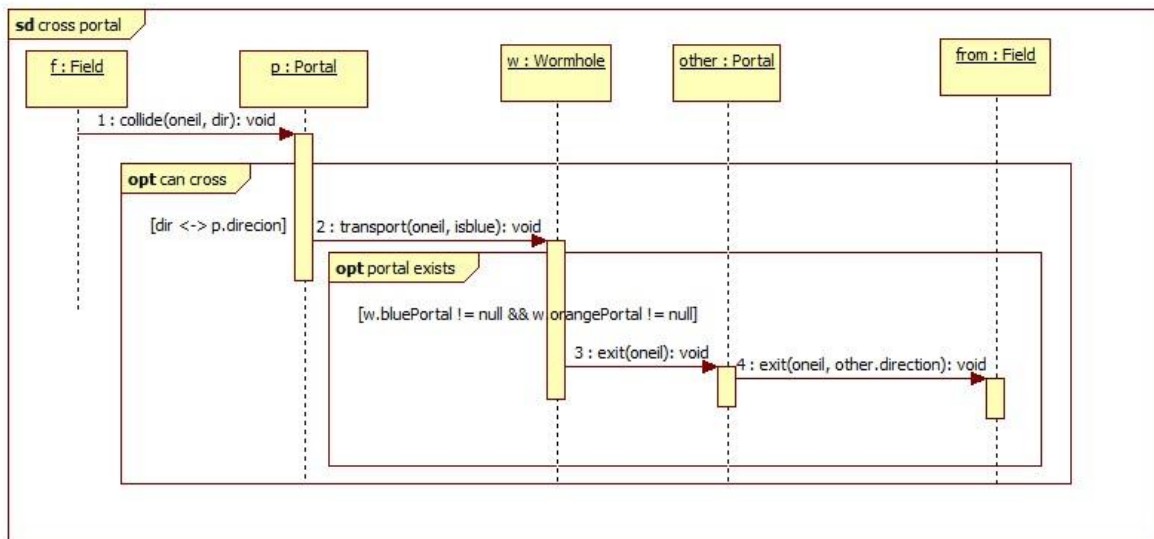
“**loves**”-Az ezredes kilő egy lövedéket egy adott irányba,



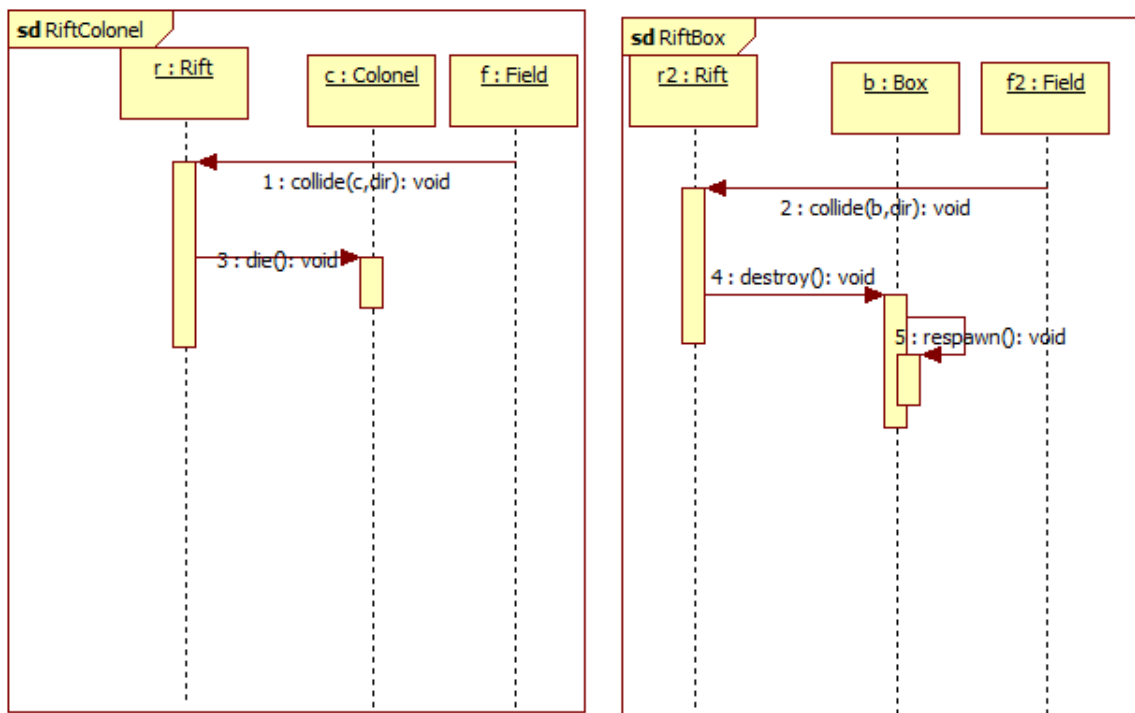
**“portálnytítás”**-Az ezredes portált nyit egy portálfalon(és az esetlegesen létező azonos színű portál megsemmisül)



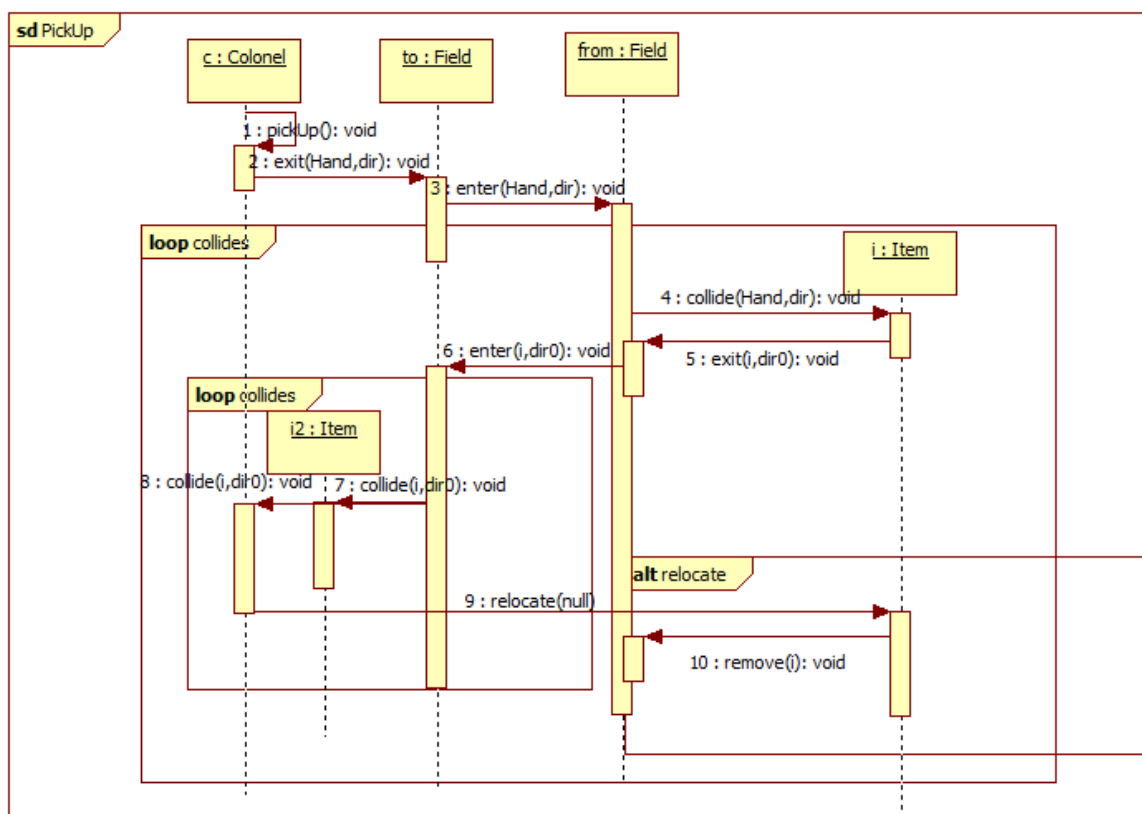
“**cross portal**”-Az ezredes át akar kelni egy nyitott fűregijáraton.



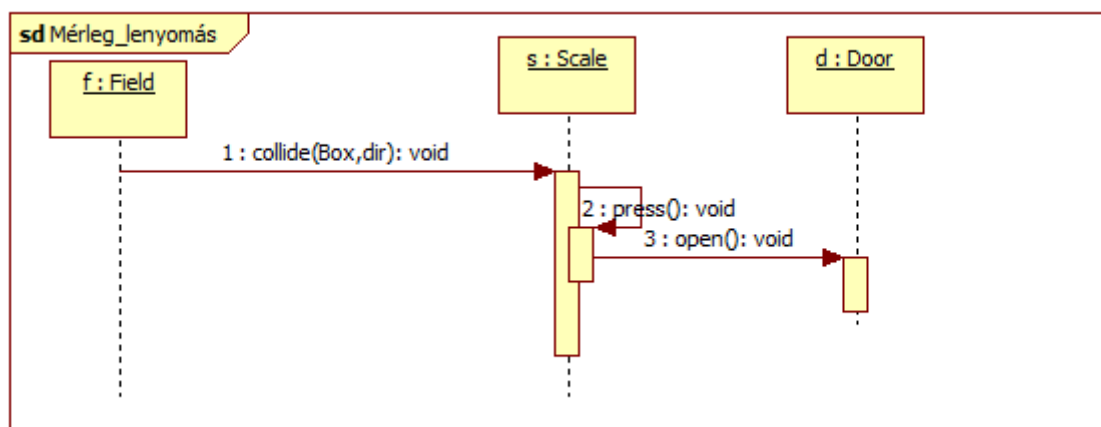
“**RiftColonel**”- Az ezredes beleesik a szakadékba és meghal. “**RiftBox**”- A doboz beleesik a szakadékba és meghal.



”**PickUp**”- Felveszünk egy dobozt.

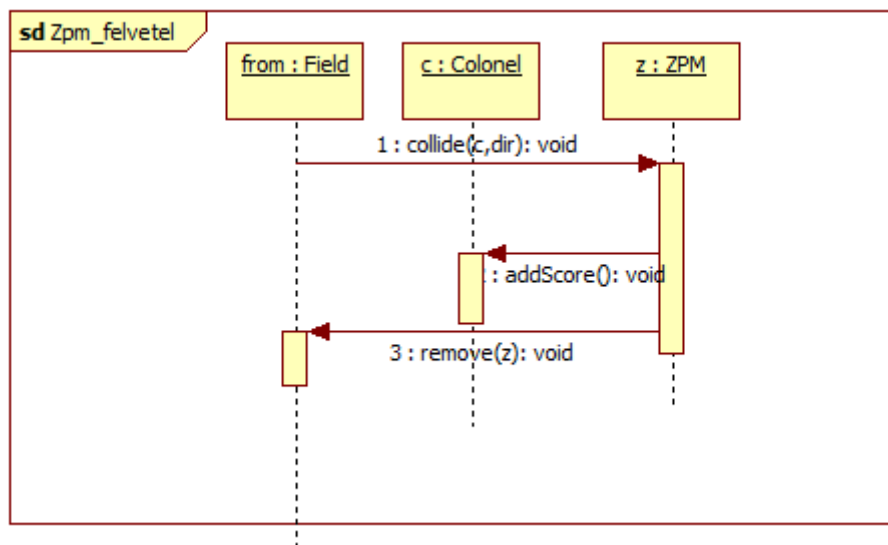


”Mérleg\_lenyomás”-Mérleg lenyomása dobozzal

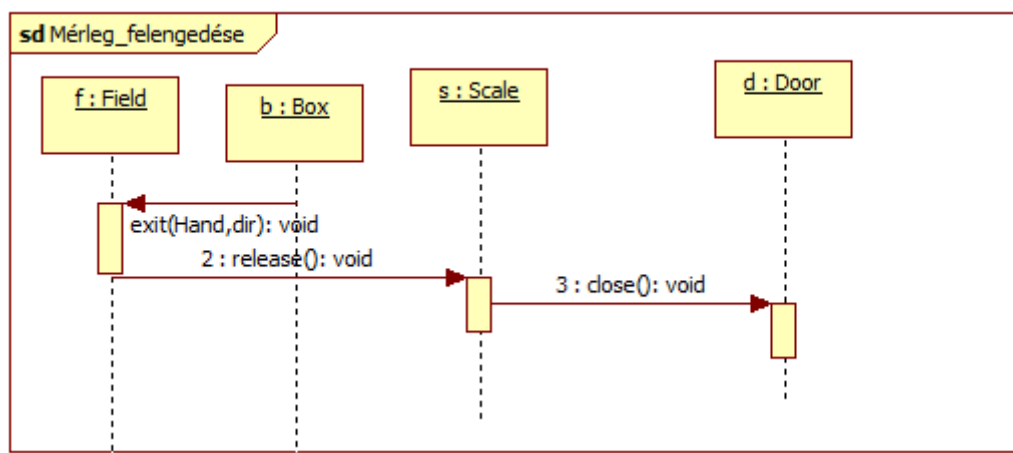


”ZPM felvétel”: ONeil felveszi a ZPMet

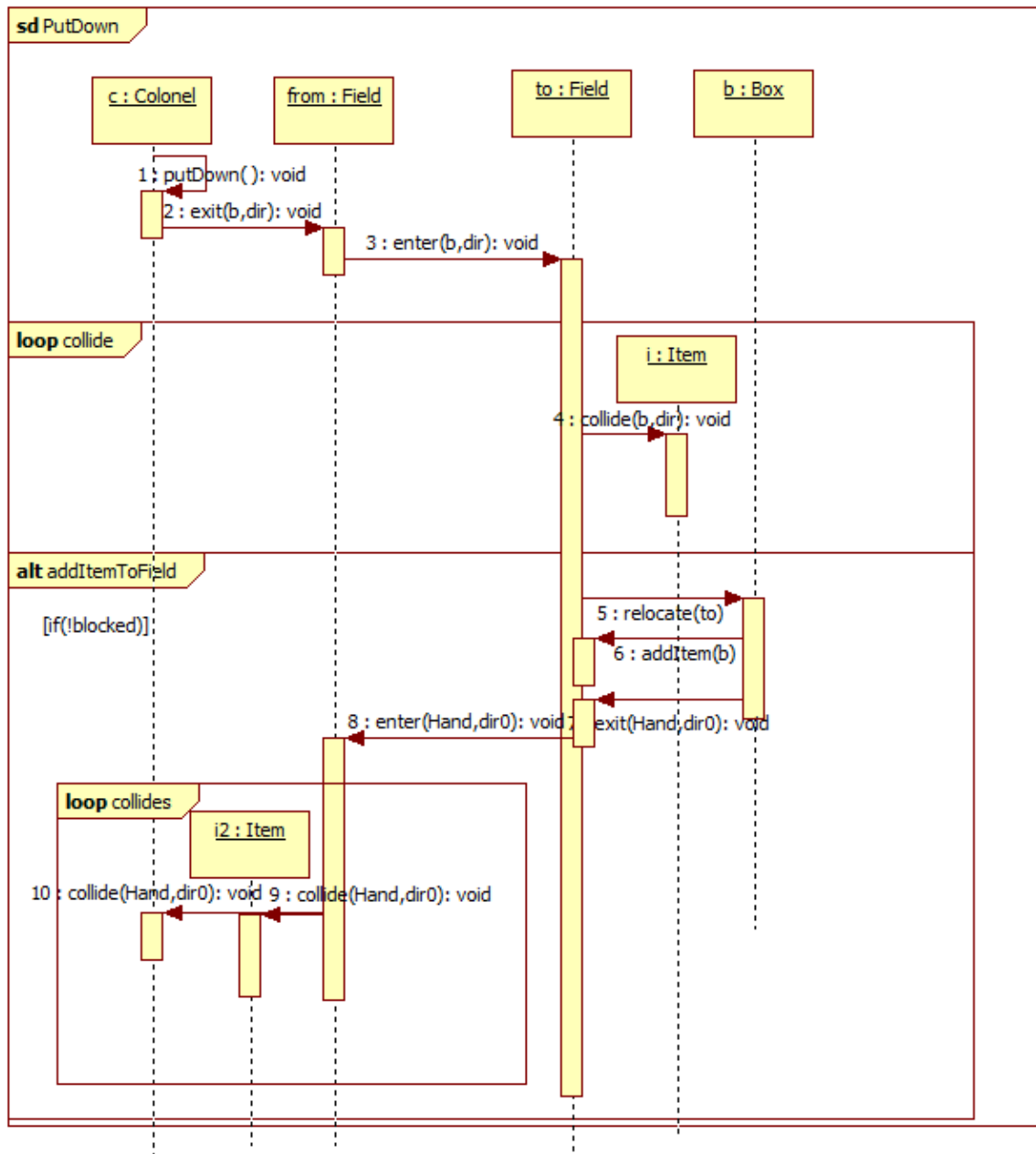




“**Release\_Scale**”-Mérleg felengedése doboz levétellel



“PutDown”- Oneil leteszi a dobozt az előtte lévő mezőre.



#### 4.5 State-chartok

Az osztálydiagram felépítéséből és a meghozott döntésekből kifolyólag a feladat megoldása nem igényeli state-chartok megvalósítását.

4.6 **Napló**

<b>Kezdet</b>	<b>Időtartam</b>	<b>Résztevők</b>	<b>Leírás</b>
2016.03.09. 20:00	1 óra	Ugrin	Osztálydiagramm tervezés,készítés
2016.03.10. 22:00	1 óra	Ugrin	Osztálydiagramm tervezés,készítés
2016.3.13. 18:00	1.5 óra	Zwick	2 szekvenciadiagramm prototípusának elkészítése, hibakeresés a modellben
2016.03.14 12:00	3 óra	Matkovics, Szabó, Márin, Ugrin	Osztálydiagramm módosítás és Szekvencia diagram készítés
2016.03.13 14:00	2,5 óra	Blaskó	Objektumkatalógus elkészítése, Osztályleírás
2016.03.14 22:00	30 perc	Blaskó	Napló szerkesztése
2016.03.14 22:00	1,5 óra	Zwick	Osztályleírások frissítése és ellenőrzése