

**Nagy Bozsó Zsombor**  
**RIPKBY**  
**Szoftvertchnológia pót nagy házi**  
**2022. 12. 8.**  
**Budapesti Műszaki és Gazdaságtudományi Egyetem**  
**Pamacsok**

**1. A feladat leírása**

A Szaturnusz légkörében különböző fajtájú pamacsok lebegnek egymással összekapcsolódva, kolóniát alkotva. Ha két szomszédos pamacs egyforma fajtájú, akkor néha egy új pamacsot állítanak elő, amelynek fajtája teljesen véletlenszerű, és nem feltétlenül egyezik meg egyik szülő fajtájával sem. Az új pamacsnak kezdetben csak a szülei a szomszédjai, a szülők szomszédossága pedig megszűnik. A pamacsoknak bármennyi szomszédja lehet. A pamacsokon különböző élőlények élnek, amelyek a pamacsokon tudnak mozogni. Vannak atkák, amelyek kiszívják a pamacs erejét. Ha egy pamacs ereje elfogy, a pamacs elpusztul, a pamacs szomszédjai összekapcsolódnak, az addig a pamacson álló élőlények a szomszéd pamacsokra kerülnek. Vannak hangyák, amelyek meg tudják növelni egy pamacs erejét, de ha egy pamacs túltöltődik, a pamacs osztódik, a két új pamacs egymás szomszédja lesz, az eredeti szomszédokat pedig egymás közt szétosztva megöröklik. Vannak katicák, amelyek a velük egy pamacson álló atkákat és hangyákat tudják megenni. A pamacsokon általában akármennyi élőlény állhat. Van olyan fajta pamacs, amelyik spontán osztódik, ha a rajta lévő élőlények száma meghaladja az 5-öt. Van olyan pamacsfajta, amelyik a rálépő lényeket megmérgezi. Ettől az atka elpusztul, a hangya elveszíti a pamacserő-növelő képességét. Van olyan pamacs, ami véletlenszerűen a rajta álló lényeket megrázza. Ettől az atka 5 pamacson kell átlépjen ahhoz, hogy újra képes legyen erőt szívni. A hangyát a rázás nem zavarja. A katicák a méregre immunisak, a rázástól viszont megbénulnak, amíg újabb rázás nem éri őket. A játékosok feladata a katicák irányítása. A játék célja, hogy minél nagyobb pamacskolónia jöjjön létre

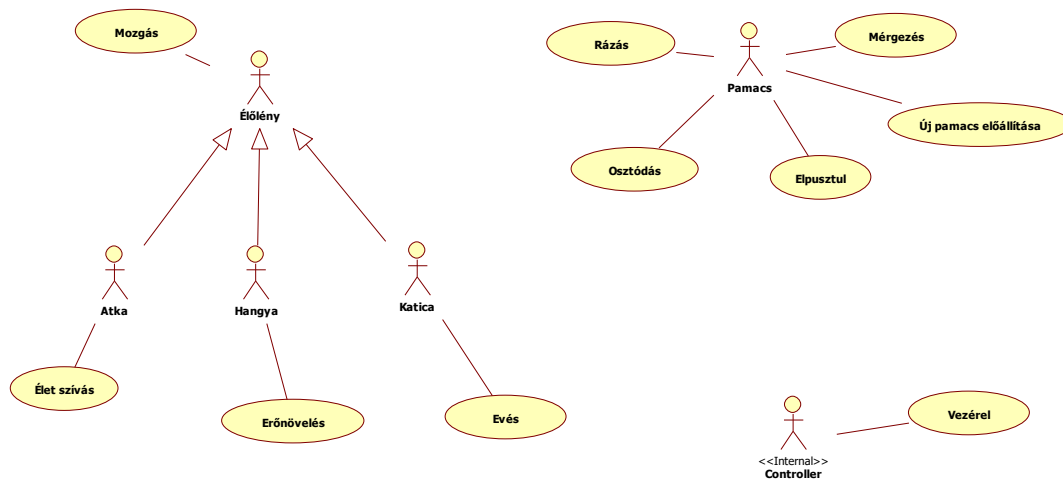
**2. Funkcionális követelmények**

Azonosító	Leírás	Use-case
R01	Két szomszédos egyforma fajtájú pamacs egy újat hoz létre.	Új pamacs előállítása, vezérel
R02	Az újonnan létrejött pamacs fajtája véletlenszerű.	Új pamacs előállítása
R03	Egy újonnan létrejött pamacs kezdetben csak a szüleivel szomszédos.	Új pamacs előállítása
R04	A szülők szomszédsága új pamacs létrejöttkor megszűnik.	Új pamacs előállítása
R05	A pamacsokon élőlények vannak.	Mozgás
R06	Az élőlények a pamacsokon mozognak.	Mozgás
R07	Az atka kiszívja a pamacs erejét.	Élet szívás

R08	A pamacs elpusztul, ha az ereje elfogy.	Élet szívás, elpusztul
R09	Az elpusztult pamacs volt szomszédjai összekapcsolódnak.	Élet szívás, elpusztul
R10	Az elpusztult pamacson levő élőlények a szomszédos pamacsokra kerülnek.	Élet szívás, elpusztul
R11	A hangyák növelni tudják a pamacsok erejét.	erőnövelés
R12	Ha a pamacs túltöltődik, osztódni fog.	erőnövelés, osztódás
R13	Az osztódott pamacs két része egymás szomszédja lesz.	erőnövelés, osztódás
R14	Az osztódott pamacs szomszédjai szétosztódnak a két új pamacs közt.	erőnövelés, osztódás
R15	A katicák meg tudják enni a velük egy pamacson álló atkákat és hangyákat.	evés
R16	Egy pamacson akármennyi élőlény állhat.	mozgás, vezérel
R17	Van olyan pamacs, amelyik osztódik, ha a rajta lévő élőlények száma meghaladja az 5-öt	osztódás, vezérel
R18	Van olyan pamacs, amelyik megmérgezi a rálépő lényeket.	mérgezés, vezérel
R19	Van olyan pamacs amelyik véletlenszerűen megrázza a rajta álló élőlényeket.	rázás, vezérel

### 3. Use-case-ek

#### 3.1. Use-case diagram



#### 3.2 Use-case leírások

Cím	Mozgás
Leírás	Az élőlények a pamacsokon tudnak mozogni
Aktor	Élőlény
Főforgatókönyv	1.A.1. Az élőlény mozog egy pamacsról, egy azzal szomszédos pamacsra.
Alternatív forgatókönyv	1.B.1. Ha a pamacs elpusztul az élőlények átkerülnek a szomszédos pamacsokra és azon mozognak.

Cím	Élet szívás
Leírás	Kiszívja a pamacs életerejét egy atka.
Aktor	Atka
Főforgatókönyv	1.A.1. Az atka kiszívja azon pamacs életerejét, amelyiken áll.
Alternatív forgatókönyv	1.B.1. Ha a pamacs ereje elfogy, akkor elpusztul és a rajt lévő élőlények a szomszédos pamacsokra kerülnek. Szomszédjai összekapcsolódnak.

Cím	Erőnövelés
Leírás	Egy hangya megnöveli a pamacs életerejét.
Aktor	Hangya
Főforgatókönyv	1.A.1. A hangya megnöveli azon pamacs életerejét, amelyiken áll.
Alternatív forgatókönyv	1.B.1. Ha a pamacs túltöltődik, akkor ketté osztódik és ezek a pamacsok egymás szomszédjai lesznek. Az rajta álló élőlények és a szomszédjai elosztódnak a két új pamacs közt.

Cím	Evés
Leírás	A Katica meg tudja enni a vele egy pamacson álló atkákat és hangyákat.
Aktor	Katica
Főforgatókönyv	1.A.1. A katica megeszi a vele egy pamacson álló atkákat és hangyákat, melyek ettől elpusztulnak.

Cím	Rázás
Leírás	A pamacs véletlenszerűen megrázza a rajta álló élőlényeket.
Aktor	Pamacs
Főforgatókönyv	1.A.1. A Pamacs megráz egy rajta álló élőlényt.
Alternatív forgatókönyv	1.B.1. A pamacs megráz egy atkát. Annak 5 pamacson kell átlépnie, hogy újra képes legyen erőt szívni.
Alternatív forgatókönyv	1.C.1. A pamacs megráz egy hangyát. A hangyát ez nem zavarja.
Alternatív forgatókönyv	1.D.1. A pamacs megráz egy katicát. A katica megbénul, amíg újabb rázás nem éri.

Cím	Mérgezés
Leírás	A pamacs véletlenszerűen megmérgez egy rajta álló élőlényt.
Aktor	Pamacs
Főforgatókönyv	1.A.1. A pamacs megmérgez egy rajta álló élőlényt.
Alternatív forgatókönyv	1.B.1. A pamacs megmérgez egy atkát. Az atka elpusztul.
Alternatív forgatókönyv	1.C.1. A pamacs megmérgez egy hangyát. A hangya elveszíti a pamacserő növelő képességét.
Alternatív forgatókönyv	1.D.1. A pamacs megmérgez egy katicát. A katica immunis a méregre.

Cím	Új pamacs előállítás
Leírás	Két szomszédos egyforma fajtájú pamacs néha előállít egy új teljesen véletlenszerű fajtájú pamacsot.
Aktor	Pamacs
Főforgatókönyv	1.A.1. Ha két szomszédos pamacs egyforma fajtájú, akkor előállítanak egy új, random fajtájú pamacsot. Ez nem függ a szülők fajtájától. Az új pamacs kezdetben a szülei szomszédja csak és a szülők szomszédsága megszűnik.

Cím	Osztódás
Leírás	A pamacs két új pamacsra osztódik.
Aktor	Pamacs
Főforgatókönyv	1.A.1. A pamacs két új pamacsra osztódik. A rajta levő élőlények és a szomszédjai szétosztódnak a két új pamacson. A két új pamacs egymás szomszédja lesz.
Alternatív forgatókönyv	1.B.1. Ha a pamacs erejét a hangyák túltöltik, az két új pamacsra osztódik.
Alternatív forgatókönyv	1.C.1. Van olyan pamacs amelyik, ha a rajta lévő élőlények száma meghaladja az 5-öt, osztódik két új pamacsra.

Cím	Elpusztul
Leírás	Ha a pamacs életereje elfogy, akkor elpusztul.
Aktor	Pamacs
Főforgatókönyv	1.A.1. A pamacsnak, ha elfogy az életereje, akkor elpusztul. A szomszédos pamacsai összekapcsolódnak és a rajta levő élőlények a szomszédos pamacsokra kerülnek.

Cím	Vezérel
Leírás	Random időközönként bekövetkező eseményeket kezeli és irányítja a hangyákat és atkákat.
Aktor	Controller
Főforgatókönyv	1.A.1. A random időnként bekövetkező eseményeket kezeli a pamacsokon.
Alternatív forgatókönyv	1.B.1. Irányítja a hangyákat és az atkákat.

## 4. Strukturális leírás

### 4.1 Az osztályok leírása

#### 4.1.1 Élőlény

Felelősségek: Egy élőlényt reprezentál. Tárolja, hogy melyik pamacson áll, lépteti azt és kezeli az élőlénnel történő eseményeket.

Attribútumok:

-ittÁll: Pamacs	Ezen a pamacson tartózkodik az adott élőlény.
-----------------	---

Metódusok:

+lép(p: Pamacs)	Az élőlény egyik pamacsról egy másik pamacsra lép át.
+setIttÁll(p: Pamacs)	Beállítja, hogy az élőlény melyik pamacson van.
+getIttÁll(): Pamacs	Megadja, hogy az élőlény melyik pamacson van.
+mérgeződik()	Az élőlény mérgeződik. Felül van definiálva.
+rázódik()	Az élőlény rázódik. Felül van definiálva.
+meghal()	Az élőlény meghal és törlődik a pamacsról.

#### 4.1.2 Katica

Felelősségek: Egy katicát reprezentál, melyet a játékos irányít.

Attribútumok:

-megrázva: boolean	Ha igaz, akkor a katica le van bénulva és nem tud mozogni. Minden rázáskor !megrázva-ra változik.
--------------------	---

Metódusok:

+megeszi(é: Élőlény)	A katica megeszik egy másik élőlényt, amely elpusztul, kivéve, ha az egy másik katica, mert akkor nem történik semmi.
+mérgeződik()	A katica immunis a méregre, így mérgezés hatására vele semmi nem történik.
+rázódik	A katica rázódás hatására megrázva változóját !megrázva-ra változtatja.
+meghal()	A katicára a meghal függvény hatástalan.
+setMegrázva(b: boolean)	Beállítja a megrázva értékét.
+lép(p: Pamacs)	A lépés felül definiálása, ha a katica megrázva változója igaz, akkor nem képes mozogni.

#### 4.1.3 Hangya

Felelősségek: Egy hangyát reprezentál. Tárolja, hogy képes-e pamacsot tölteni, tölteni tud egy pamacsot és kezeli, hogy rázás és mérgezés hatására mi történik vele.

Attribútumok:

-képesség: boolean	Amennyiben hamis értéket vesz fel, a hangya nem képes tölteni a pamacsot.
--------------------	---

Metódusok:

+erőtNövel()	Annak a pamacsnak, amelyiken a hangya áll képes növelni az erejét. Vizsgálja azt, hogy túltöltődik-e és meghívja ez esetben a pamacs osztódását.
+setKépesség(b: boolean)	Beállítja a pamacs töltő képességet.
+getKépesség(): boolean	Visszaadja a pamacs töltő képességet.
+mérgeződik	A hangya mérgezés hatására elveszíti a pamacstöltő képességét.
+rázódik	A rázás a hangyát nem zavarja, így nem történik vele rázás hatására semmi.

#### 4.1.4 Atka

Felelősségek: Egy Atkát reprezentál. Tárolja, hogy rázás után hányat lépett és kezeli, hogy mi történik vele mérgezés és rázás hatására, valamint végrehajtja a pamacs leszívást.

Attribútumok:

-rázásután: int	Ennyi lépést tett meg az atka miután egy pamacs megrázta. Rázás hatására 5 az értéke és egyesével csökken. [0;5]
-----------------	--

Metódusok:

+Szív()	Az atka kiszívja a pamacs életét. Figyeli, hogy a szívás után a pamacsnak maradt-e még élete és amennyiben nem, akkor meghívja a pamacs halálát és kezeli az élőlények leléptetését róla és a szomszédok átszervezését.
+mérgeződik	Mérgeződés hatására az atka meghal, megszűnik létezni.
+rázódik	Az atka rázásután változója 5-re változik.
+setRázásután(i: int)	Beállítja a rázásután változó értékét.
+getRázásután(): int	Visszaadja a rázásután értékét.
+lép(p: Pamacs)	A lépés felül definiálása, annyi különbséggel, hogy ha a rázásután értéke nagyobb mint 0, akkor minden lépéskor egyel csökken.

#### 4.1.5 Pamacs

Felelősségek: Egy pamacsot reprezentál. Tárolja a szomszédos pamacsokat, az életerejét, a rajta levő élőlényeket, típusát egy számként és a maximális életerejét. Létrehozáskor mindig egy random típusa jön létre megadott maximális életelővel.

Attribútumok:

-szomszédok: Pamacs[]	A pamacs szomszédos pamacsainak listája.
-életelő: int	Ennyi a pamacs aktuális életereje.
-élőlényekRajta: Élőlény[]	A pamacson aktuálisan tartózkodó élőlények.
-maxÉletelő: int	A pamacs maximális életereje.
-típus: int	A pamacs típusa azonosítva egy számmal.

Metódusok:

+újPamacs(másikszülő: Pamacs): Pamacs	A controller akkor hívja csak meg, ha két egymás melletti pamacs egyforma típusú. Egy új pamacsot hoz létre, amely csak a szüleivel szomszédos és a szülők szomszédsága megszűnik.
+addSzomszéd(p: Pamacs)	Egy pamacsot hozzáad a szomszédsági listához.
+delSzomszéd(p: Pamacs)	Egy pamacsot töröl a szomszédsági listáról.
+getÉletelő(): int	Visszaadja a pamacs aktuális életerejét.
+setÉletelő(i: int)	Beállítja a pamacs aktuális életerejét.
+meghal()	A pamacs elpusztul. A szomszédok összekapcsolódnak egymással, a rajta levő élőlények átlépnek a szomszédos pamacsokra.
+getMaxÉletelő(): int	Visszaadja a pamacs maximális életerejét.
+osztódás(): Pamacs	A pamacsból egy újabb ugyan olyan típusú pamacs jön létre, amely szomszédos lesz a másik pamacs szomszédjaival.
+szomszéde(p: Pamacs): boolean	Visszaadja, hogy két pamacs szomszédos-e.
+addÉlőlény(é: Élőlény)	Hozzáad egy élőlényt a pamacs élőlény listájához.
+deleteÉlőlény(é: Élőlény)	Töröl egy élőlényt a pamacs élőlény listájáról.
+getTípus(): int	Visszaadja a pamacs típusát reprezentáló számértéket.



#### 4.1.6 Osztódó

Felelősségek: Egy pamacs, amely osztódik, ha több mint 5 élőlény tartózkodik rajta.

Attribútumok: -

Metódusok:

+getÉlőlényekSzamaRajta(): int	Megszámolja, hogy hány élőlény tartózkodik a pamacson.
+osztódás(): Pamacs	Az osztódás felüldefiniálása, ha több mint 5 élőlény van a pamacson, akkor a controller meghívja és a pamacs osztódik.

#### 4.1.7 Mérgező

Felelősségek: Egy pamacs, amely meg tudja mérgezni az azon tartózkodó élőlényeket.

Attribútumok: -

Metódusok:

+mérgez()	Megmérgezi a rajta tartozó élőlényeket. Meghívja azok mérgeződik függvényeit.
-----------	--

#### 4.1.8 Rázó

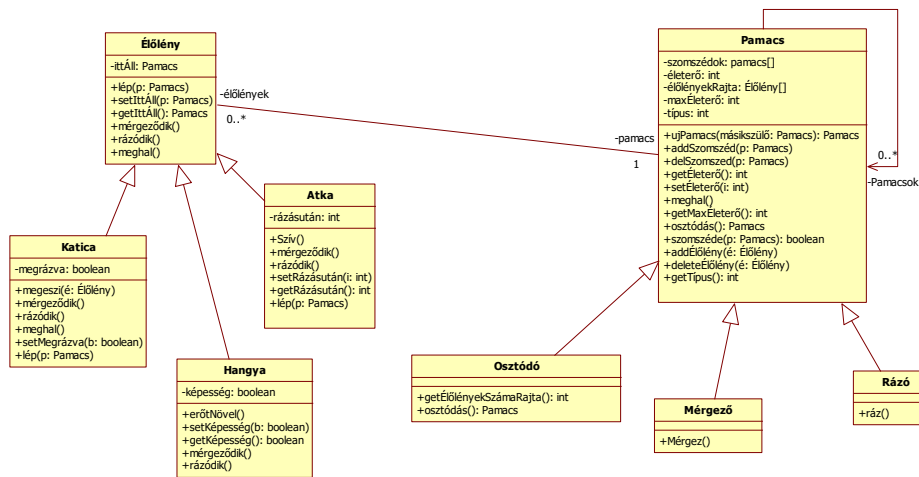
Felelősségek: Egy pamacs, amely meg tudja rázni a rajta tartózkodó élőlényeket.

Attribútumok: -

Metódusok:

+ráz()	Megrázza a rajta tartózkodó élőlényeket. Meghívja azok rázódik függvényeit.
--------	--

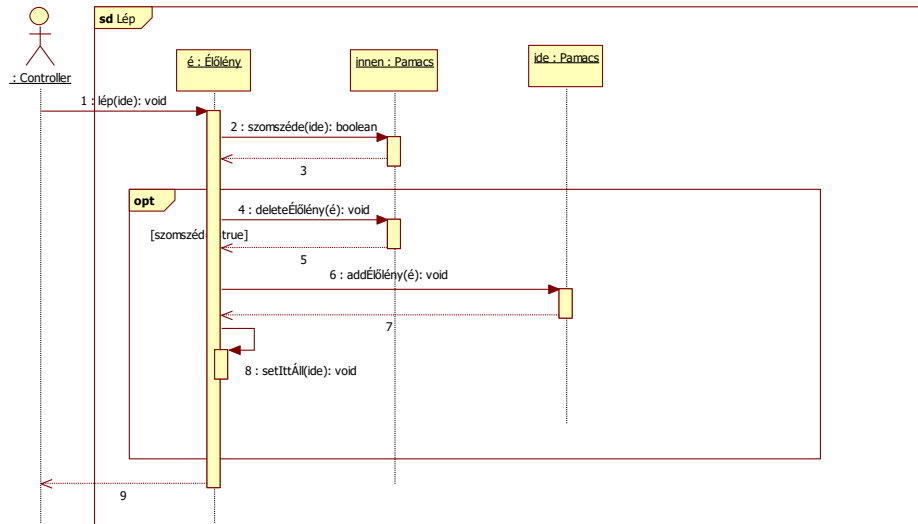
## 4.2 Osztálydiagram



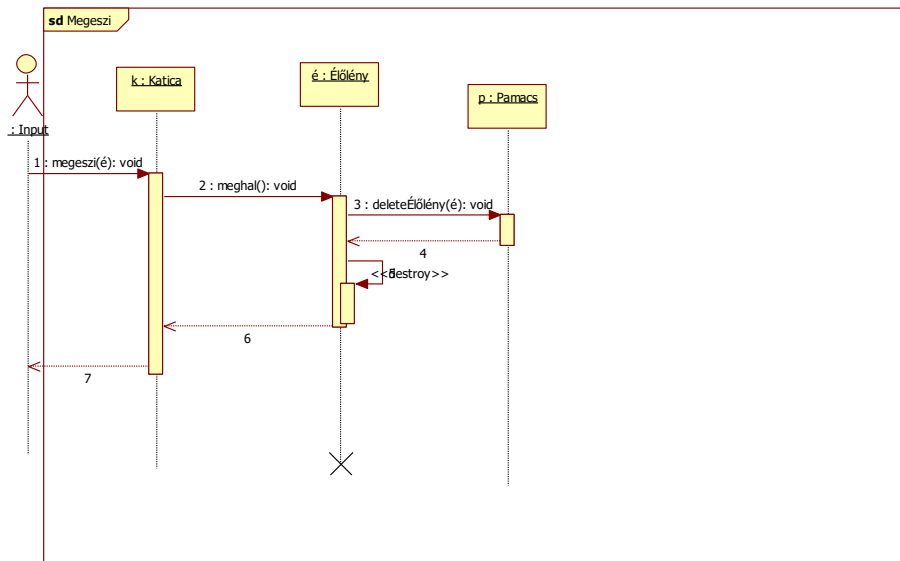
## 5. Viselkedés Leírása

### 5.1 Szekvencia diagrammok

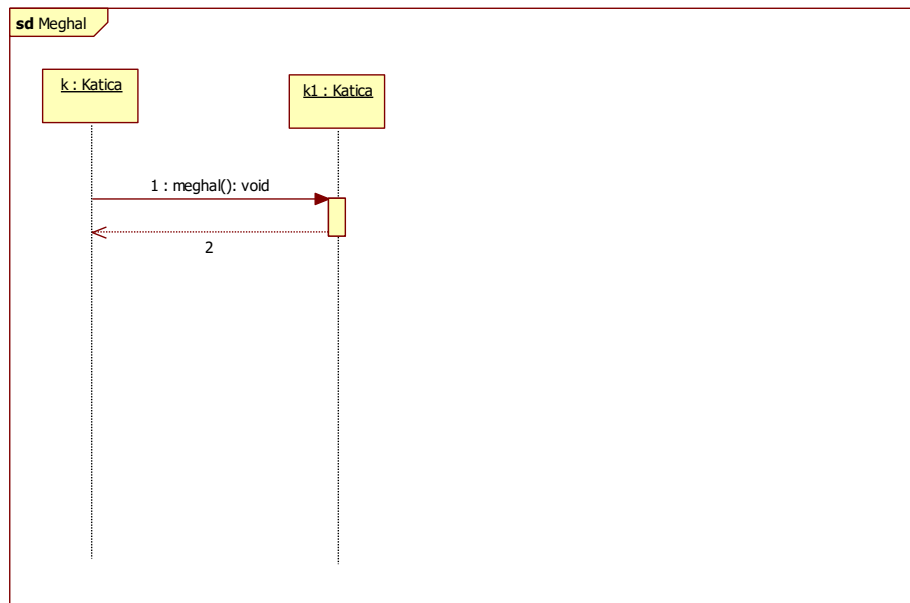
#### 5.1.1 Lép



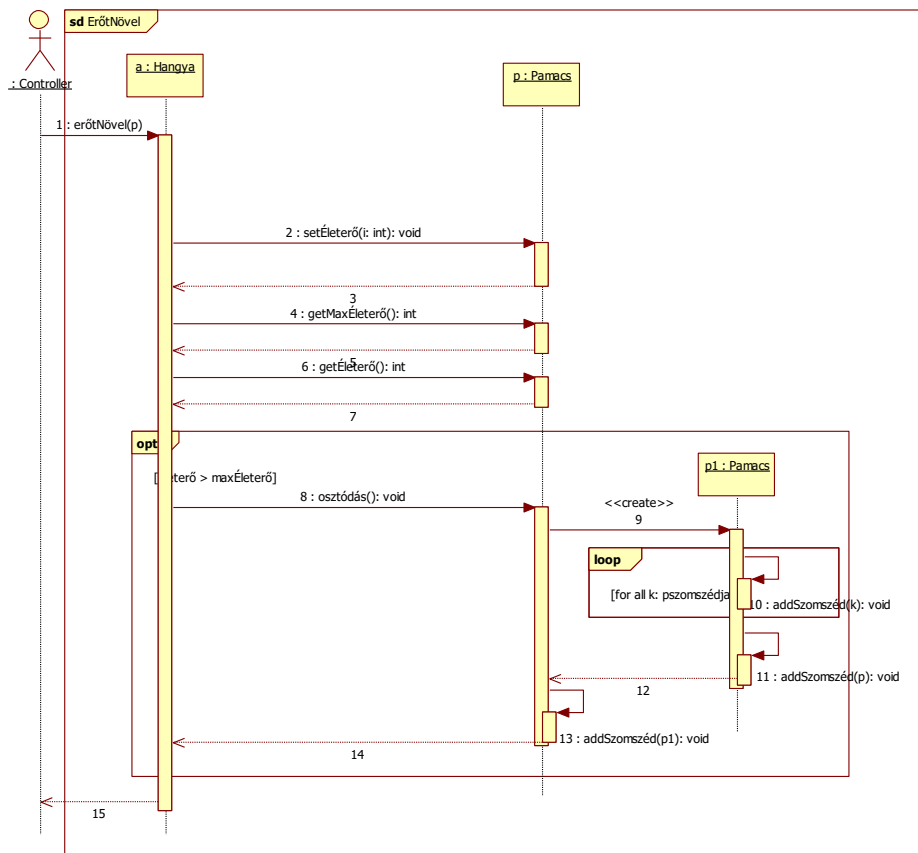
## 5.1.2 Megeszi



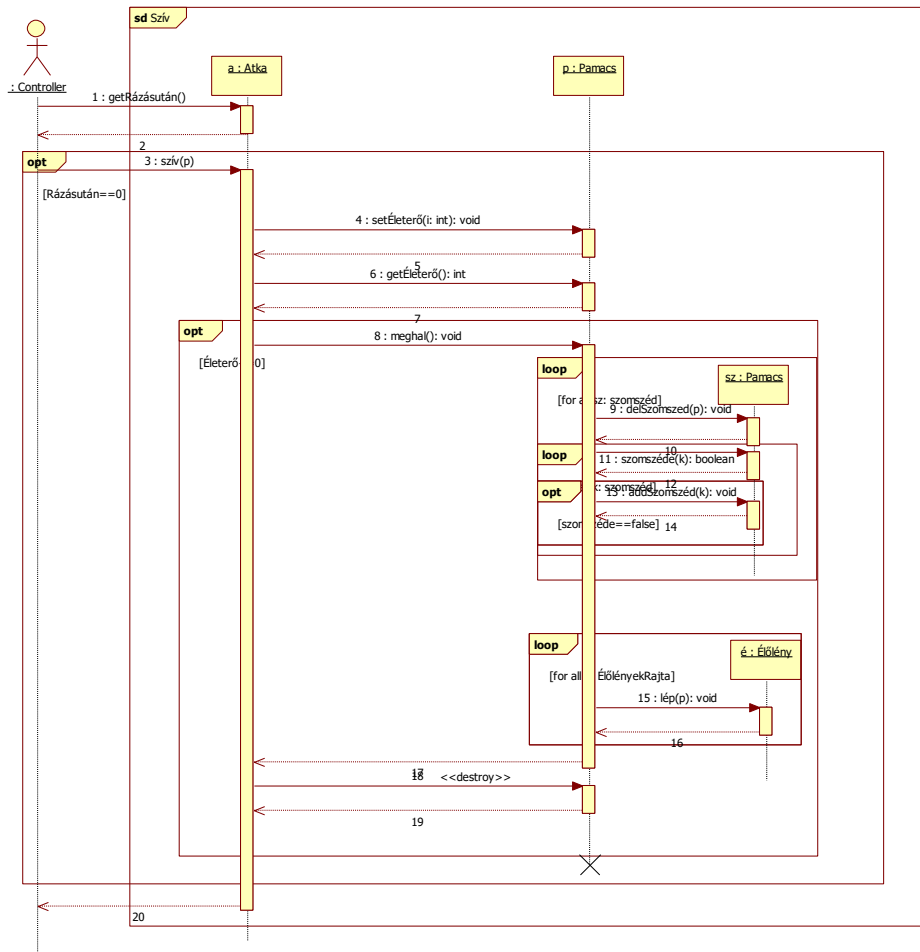
### 5.1.3 Katica megeszi a katicát



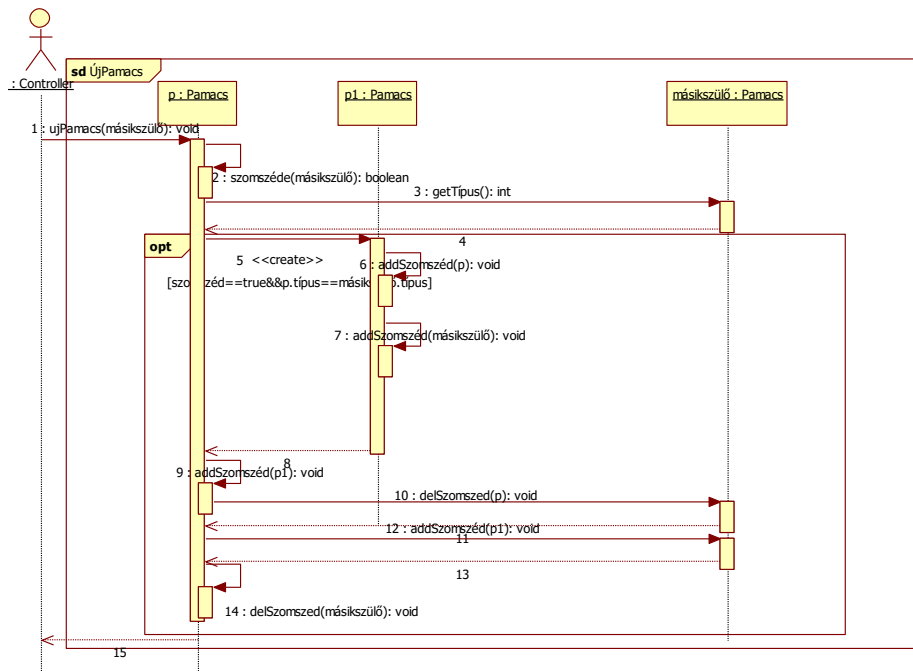
### 5.1.4 Erőt növel



## 5.1.5 Szív

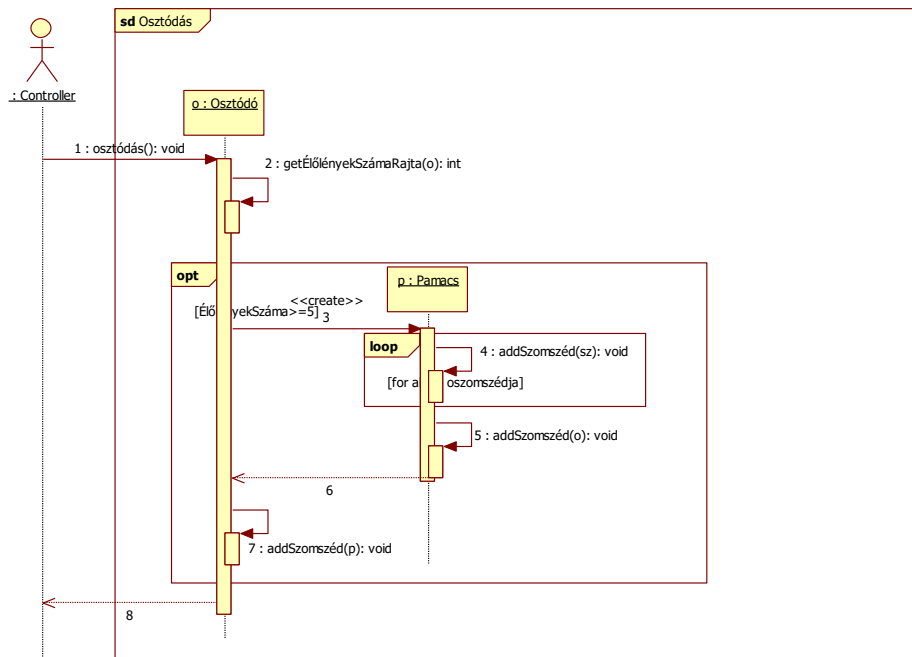


## 5.1.6 Új Pamacs

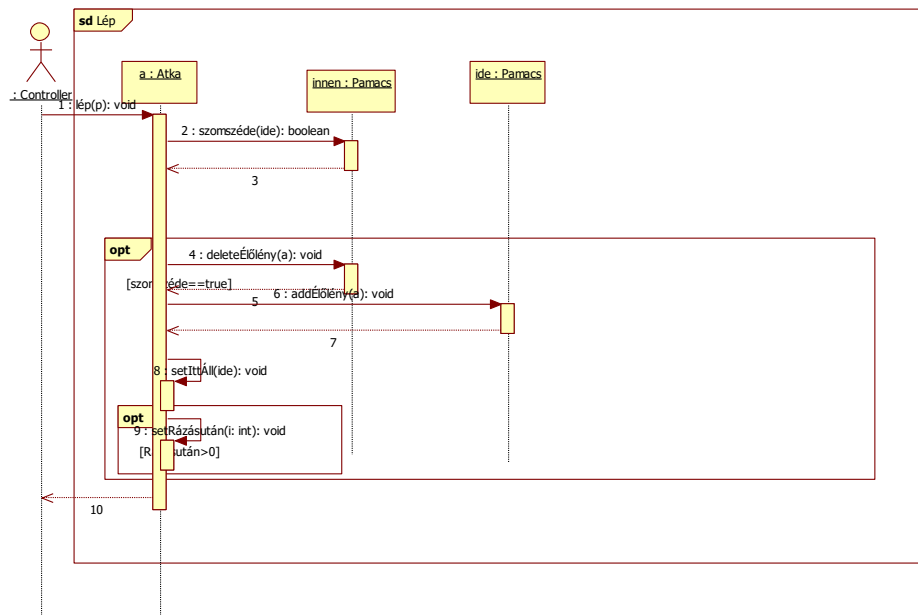




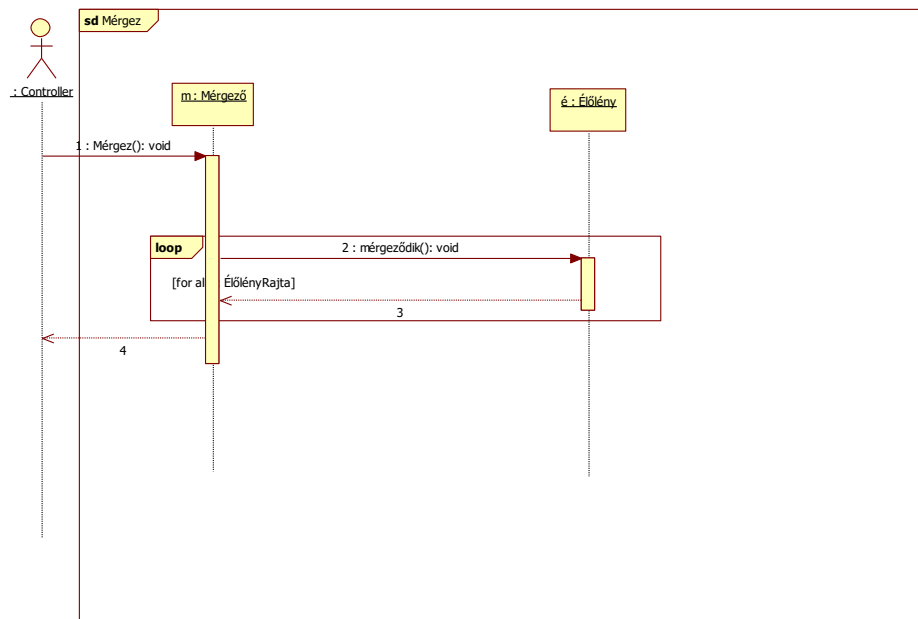
## 5.1.7 Osztdás



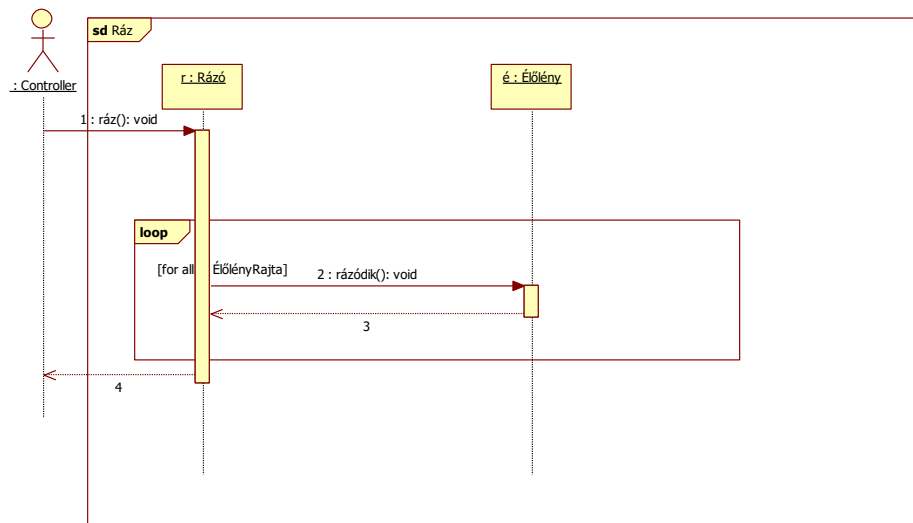
## 5.1.8 Atka lép



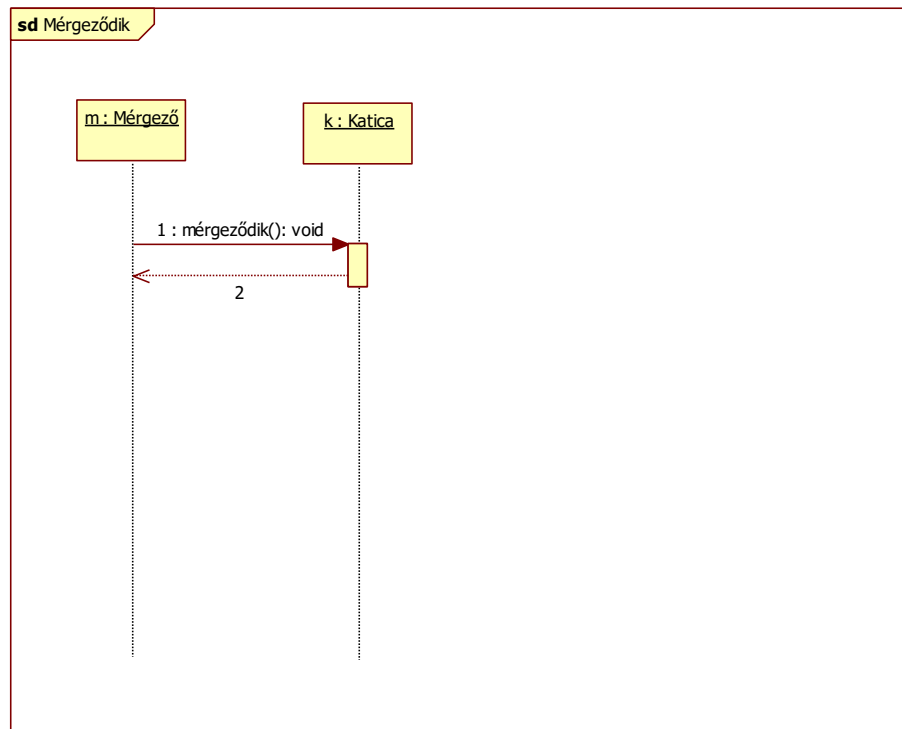
## 5.1.9 Mérgez



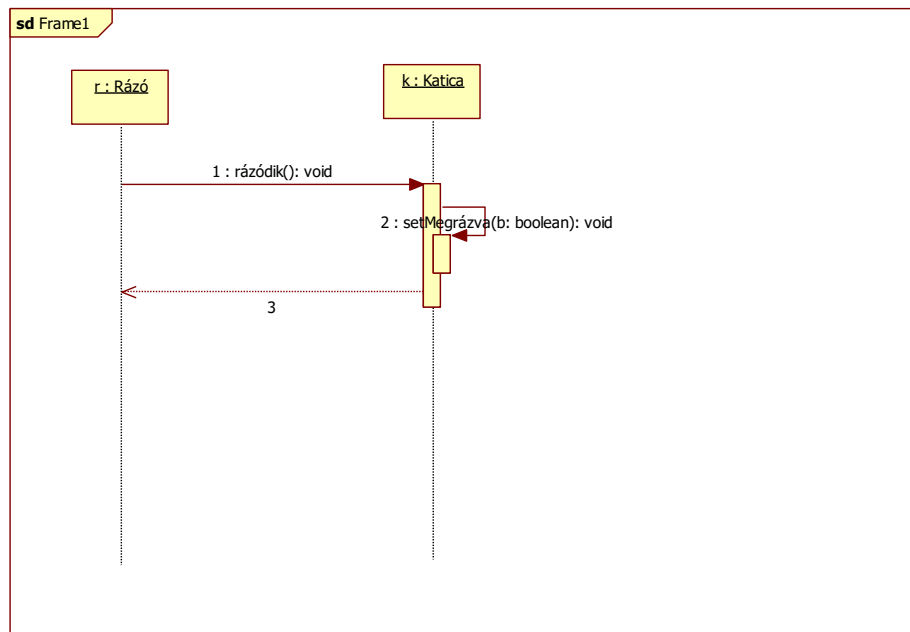
### 5.1.10 Ráz



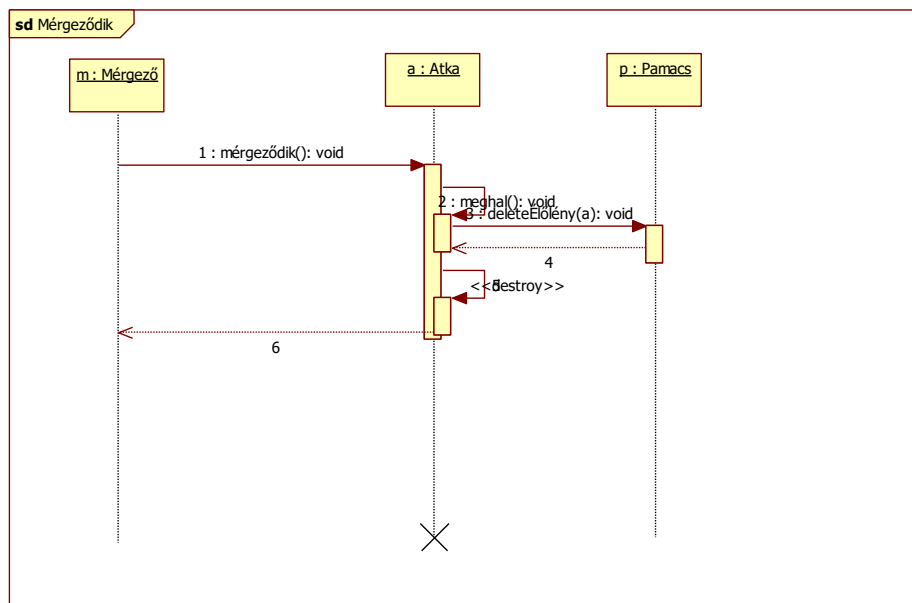
### 5.1.11 Katica mérgeződik



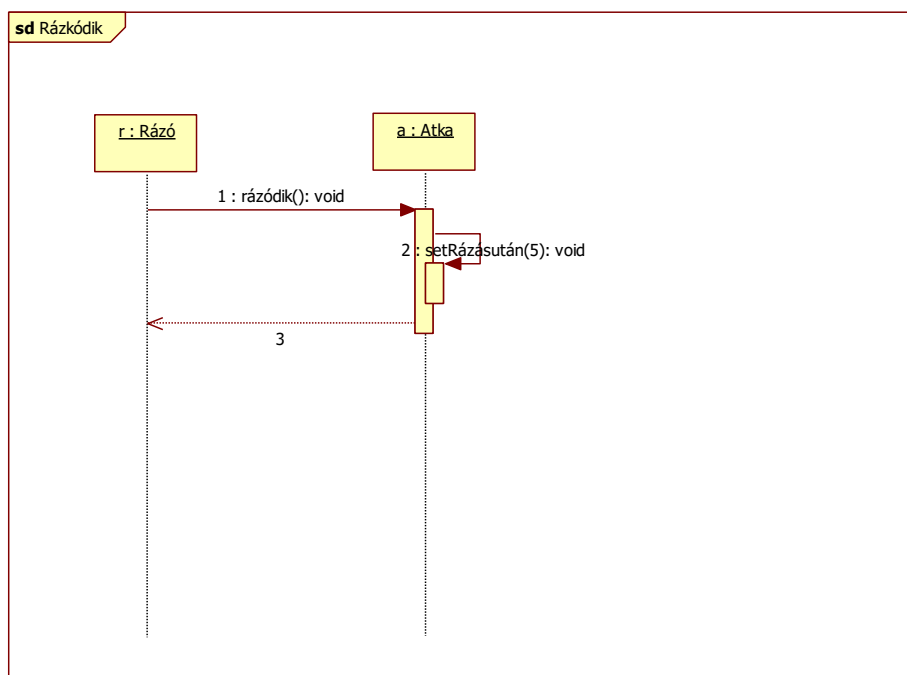
### 5.1.12 Katica rázódik



### 5.1.13 Atka mérgeződik

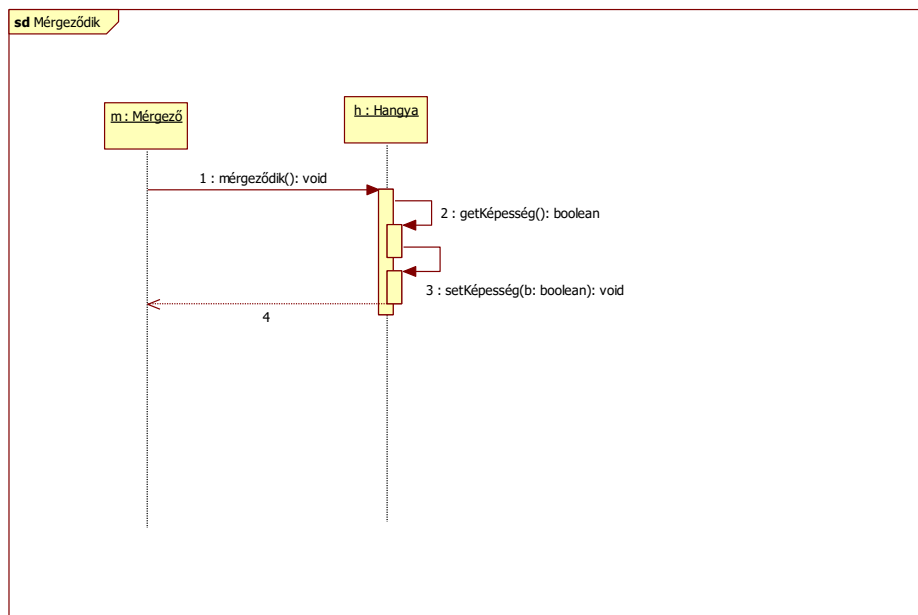


### 5.1.14 Atka rázódik

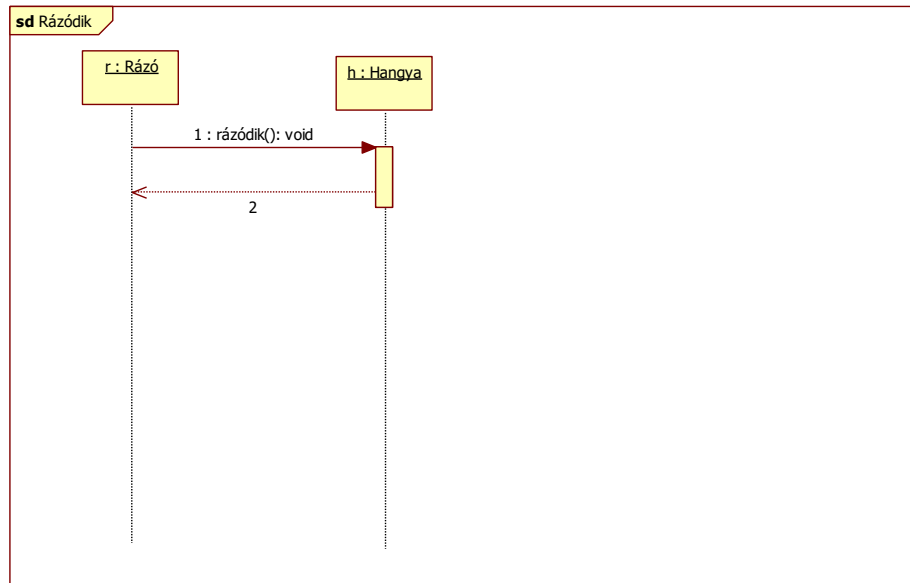




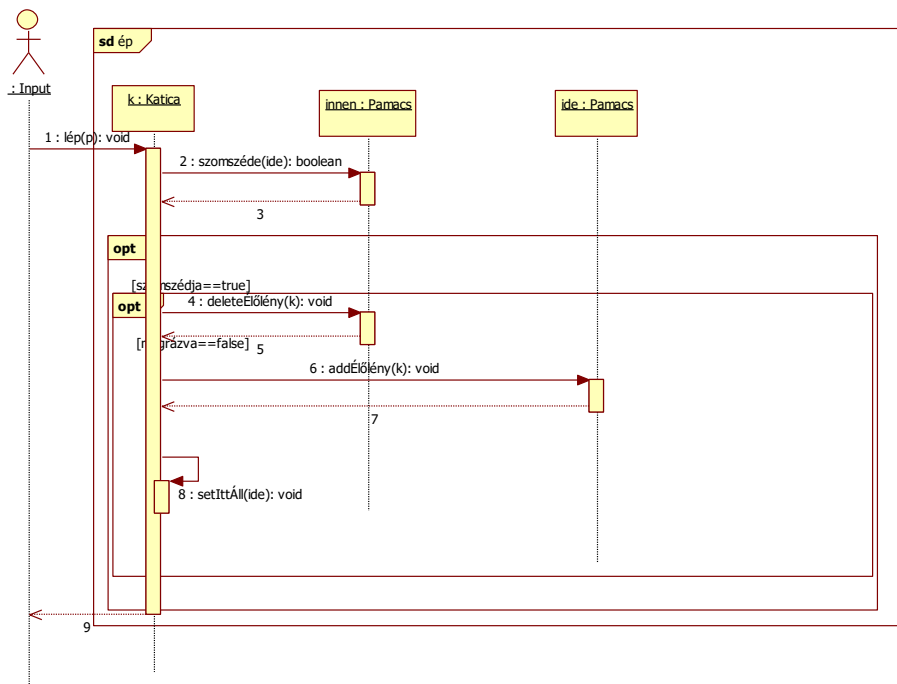
### 5.1.15 Hangya mérgeződik



### 5.1.16 Hangya rázódik

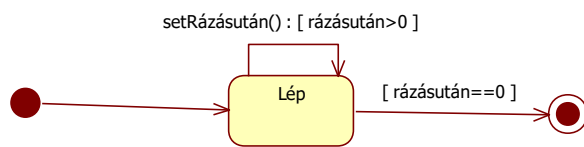


### 5.1.17 Katica lép



## 5.2 Állapotgép diagrammok

### 5.2.1 Rázásután State



## 6. Napló

2022.12.8.	3 óra	Követelménylista és use-case-ek elkészítése	2, 3
2022.12.9.	5 óra	Osztálydiagram elkészítése és osztályok leírása	4
2022.12.10.	2 óra	Eddigiek javítása	2, 3, 4
2022.12.11.	8 óra	Szekvencia diagrammok elkészítése, naplózás	5, 6

Összes elvégzett munka: 18 óra

Modellező eszköz: WhiteStarUML

Megjegyzés: az UML eszköz nem támogatja az execution specification jelölést így azt egy a framen kívüli input aktorral modelleztem.

Egyéb eszközök: Word