

## 2. Követelmény, projekt, funkcionalitás

23 – totoro

Konzulens:

Haragos Gergő Viktor

### Csapattagok

Bohus János	XYOVFZ	janos.bohus1@gmail.com
Dúcz Ákos	GC1RTE	akos.ducz@gmail.com
Galambos Csaba	BVBQKN	galambos.csaba05@gmail.com
Horváth	OODMXV	avarkahorvath@gmail.com
Avarka	AW3JH1	vegvaripeter21@gmail.com
Végvári Péter		m

2025.02.18

## 2. Követelmény, projekt, funkcionalitás

### 2.1 Bevezetés

#### 2.1.1 Cél

A dokumentum célja a Fungorium nevű játék és az azt elkészítő projekt részletes leírása. A dokumentum tájékoztatást ad a játék belső és külső működéséről, illetve pontos specifikációt ad a játékhoz.

#### 2.1.2 Szakterület

A szoftver egy többszemélyes, turn-based videojáték, amiben a kitalált Fungorium bolygó ökoszisztémájának szereplőit irányítják a játékosok.

#### 2.1.3 Definíciók, rövidítések

Turn: Az a játétfázis, amikor egy játékos végrehajtja az *action*-jét. (Másnéven lépés)

Action: Azok az események amit egy játékos a *turn*-je alatt csinál.

Round: Az a játékfázis, amely során minden játékos végrehajtja a lépését. (Másnéven kör)

Turn-based: A játékosok előre meghatározott sorrendben, egymás után lépnek. Adott mennyiségű *round*-ok sorozatából áll a játékmenet.

Local multiplayer: Helyi többjátékos mód.

#### 2.1.4 Hivatkozások

Tárgy oldal: [https://www.iit.bme.hu/oktatas/tanszeki\\_targyak/BMEVIIIAB02?language=hu](https://www.iit.bme.hu/oktatas/tanszeki_targyak/BMEVIIIAB02?language=hu)

Feladatleírás: <https://www.iit.bme.hu/file/11582/feladat>

Feladatbeadó platform: <https://devil.iit.bme.hu:9181/hercules/>

GitHub repository: <https://github.com/janosbohus/projlab>

#### 2.1.5 Összefoglalás

A dokumentum további részei ismertetik a játékkal kapcsolatos funkcionális és egyéb követelményeket. Az előbbieket megértését a use-case leírások segítik. A dokumentumban használt szakmai kifejezésekhez tartozik egy szótár is. A projekt elkészítéséhez szükséges információk a projekt tervben vannak összegyűjtve. Az értekezleteket, valamint a csapattagok tevékenységeivel kapcsolatos információkat a napló tartalmazza.

## **2.2 Áttekintés**

### **2.2.1 Általános áttekintés**

#### **Pálya:**

- Ezen helyezkedik el a játék összes objektuma.
- A pályán helyezkednek el a tektonok, ill. ahol nincs tekton, ott viszkózus réteg található.

#### **Játékosok:**

- Gombász vagy rovarász lehet.
- A gombász a gombakolóniának fejlődését irányítja.
- A rovarász egy rovar irányít

#### **Energia, tápanyag:**

- A kettő ekvivalens értékkel rendelkezik.
- Ugyanannyi tápanyagot kap egy rovar, ha megesz egy spórát, mint amennyi energiát a gombász a spórára költött.

#### **Gombász:**

- A gombász célja, hogy minél több gombatestet termeljen.
- *Action*-ök:
  - Gombatest építése
  - Gombatest fejlesztése
  - Gombafonal növesztése szomszédos tektonra
  - Spóra kilövése gombatestből

#### **Gombakolónia:**

- Egy gombász azon gombatestei és gombafonalai tartoznak egy gombakolóniába, amik össze vannak kötve gombafonallal.
- A gombafonalak szomszédos tektonokat kötnek össze.
- Egy gombakolóniába kell tartozon gombatest, különben elpusztulnak a fonalak.

#### **Rovarász:**

- A rovarászok fejenként egy rovar irányítanak.
- Célja, hogy minél több tápanyagot fogyasszon el.
- *Action*-ök:
  - Rovar mozgatása gombafonalon keresztül
  - Spóra elfogyasztása
  - Gombafonal elvágása

## 2.2.2 Funkciók

### Tekton:

- A tektonok a pálya alapelemei, 1 tektonon:
  - legfeljebb 1 gombatest,
  - gombafonalak,
  - illetve rovarok lehetnek.
- Egy tektonnak legalább 2 és legfeljebb 12 szomszédja lehet.
- A kör végén egy tekton ketté válhat, ilyenkor:
  - a rajta lévő gombafonalak elszakadnak,
  - a rajta lévő gombatest és rovar(ok) valamelyik tektonra átkerülnek,
  - a rajta lévő spórák
    - ha van rajta gombatest, akkor a gombatesthez tartozó tektonra kerülnek
    - ha nincs, akkor a két tekton között elosztódnak.
- Többféle típusú tekton létezik:
  - van olyan, ahonnan a fonalak pár kör után felszívódnak,
  - van olyan, amin nem tud gomba nőni,
  - van olyan, amin csak egy fonál tud nőni.

### Játékos:

- Egy játékos lehet gombász, vagy rovarász.
- 4, 6 vagy 8 játékos lehet egy játékban.

### Gombász:

- Egy-egy gombafaj terjedését irányítja.
- Célja, hogy minél több gombatest fejlődjön ki (élő és halott gombatestek is számítanak).

### Gombászok:

- A gombászok egymással versenyeznek.
- Az nyer, akinek a legtöbb gombatestje fejlődik ki a játék végére.

### Gombafonal:

- Két szomszédos tektont köt össze.
- A gombász növeszthet új fonalat a körében már lent lévő fonalakkal szomszédosan.
- A rovarok csak gombafonalon tudnak közlekedni a tektonok között.
- Ha nem csatlakozik gombatesthez, elpusztul.

### Gombatest:

- A gombász egy tektonon tudja elhelyezni, ha van ott fonál és megfelelő mennyiségű spóra található ott.
- Energiát termel a gombakolóniának, amit a gombász a *turn*-je kezdetén megkap.
- A gombász a termelt energiából különböző tulajdonságú spórákat tud venni, amiket a gombatesttel szomszédos tektonokra tud kilőni.
- Minél nagyobb szintű, annál több energiát termel a gombatest, és annál messzebbre tudja kilőni spóráit.
- Adott számú spóra kilövés után elpusztul.

**Spóra:**

- A spórát egy gombatestből lehet kilőni szomszédos tektonra.
- Energiát felhasználva különböző tulajdonságú spórákat lehet természeteni egy gombatesten, amiket a gombatest kilőhet egy szomszédos tektonra.
- Egy tektonon több gombász spórái is elhelyezkedhetnek.
- Több fajta létezik, melyek a rovarokat befolyásolják:
  - Lassítja a rovar, emiatt kevesebbet tudnak lépni a körükben.
  - Gyorsítja a rovar, emiatt többet tudnak lépni a körükben.
  - Több tápérték van benne, ezért fogyasztása előnyösebb a rovarnak.
  - Megbénítja a rovar, így a rovar kimarad 3 körből.
  - Megbénítja a rovar csáprágóját, így 3 körig nem tud fonalat elvágni.

**Gombakolónia:**

- Egy gombász azon gombatestei és gombafonalai tartoznak egy gombakolóniába, amik össze vannak kötve gombafonallal.

**Rovar:**

- Spórákat fogyaszt, amiből tápanyagot kap.
- Tektonok között csak gombafonalakon keresztül tud áthaladni.
- Csáprágójával képes elvágni gombafonalakat.

**Rovarász:**

- Egyetlen darab rovar irányít.
- Célja, hogy minél több spórát fogyasszon el.

**Rovarászok:**

- A rovarászok egymással versenyeznek.
- Az nyer, akinek a rovара a legtöbb spórát tudta lefogyasztani a játék végig.

**2.2.3 Felhasználók**

A játékot 4, 6, vagy 8 fő játszhatja- pontosan ugyanannyi játékos lesz gombász, mint rovarász.

**2.2.4 Korlátozások**

Kizárólag local multiplayer támogatott és csak a 2.2.3-ban definiáltak szerint lehetnek felhasználók. Nem valósidejű, hanem *turn-based*.

**2.2.5 Feltételezések, kapcsolatok**

A tárgy oldal rögzíti a dokumentum formális elvárásait, leadással kapcsolatos követelményeit. A feladatleírás tartalmazza a projekt, az elkészítendő játék alapszabályait, valamint a főbb elemeinek leírását.

A feladatbeadó platformon kell a megadott határidőig leadni a (rész)feladatokat.

A GitHub repositoryban vannak összegyűjtve a csapat számára legfontosabb információkat tartalmazó dokumentumok, valamint a feladathoz tartozó kód, maga a program megvalósítása.

## 2.3 Követelmények

### 2.3.1 Funkcionális követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Use-case	Komment
TKT001	Egy tekton két részre válhat, <i>round</i> végekor, véletlenszerűen. Ekkor a rajta lévő fonalak elpusztulnak.	Megnézzük, hogy kettévált-e.	MUST	Feladatleírás	Kör vége	
TKT002	Van olyan tekton, amin <i>turn</i> végekor elpusztul az összes gombafonal.	Megnézzük elpusztult-e gombafonal egy ilyen tektonon.	MUST	Feladatleírás	-	
TKT003	Van olyan tekton, amire nem tud gombatest nőni.	Megnézzük, hogy nem tud-e nőni gombatest.	MUST	Feladatleírás	Gombok irányítása	
TKT004	Van olyan tekton, amin csak egy gombásznak lehet fonala.	Megnézzük, hogy csak egy gombásznak lehet-e fonala.	MUST	Feladatleírás	-	
TKT004	Van olyan tekton, amin adott idő után felszívódnak a gombafonalak	Megnézzük, hogy az adott idő elteltével tényleg felszívódnak-e a gombafonalak	MUST	Feladatleírás	-	
RVR001	A rovar tud mozogni a fonalak mentén.	Rovart megpróbáljuk mozgatni két tekton között, amik össze vannak kötve fonallal.	MUST	Feladatleírás	Rovarok irányítása	
RVR002	A rovar csak a fonalakon tud közlekedni.	Kipróbáljuk, hogy át tud-e menni a fonalakon, és máson pedig nem.	MUST	Feladatleírás	Rovarok irányítása	

RVR003	A rovar el tudja vágni a fonalat.	Odamegy a rovar a tektonra, ahol van fonal, és elvágja.	MUST	Feladatleírás	Rovarok irányítása	
RVR004	A rovar meg tudja enni a spórákat.	Odamegy a rovar egy tektonra, ahol van spóra és megeszi, ezzel tápanyagot szerez.	MUST	Feladatleírás	Rovarok irányítása	
GMB001	Gomba tud spórát dobni a szomszédos tektonokra.	Spórát dobunk szomszédos tektonra.	MUST	Feladatleírás	Gombák irányítása	
GMB002	Gombatest tud szintet lépni, ez energiába kerül.	Gombatesttel szintet lépünk, megnézzük, hogy az energia levonásra került-e.	MUST	Feladatleírás	Gombák irányítása	
GMB003	Gombász választott irányba növeszthet egy gombatestből egy másik fonállal szomszédos tektonra fonalat.	Fonalt növesztünk szomszédos tektonra.	MUST	Feladatleírás	Gombák irányítása	
GMB004	Ha egy fonál kapcsolata megszakad minden gombatesttel, akkor az néhány körön belül elpusztul.	Megnézzük elpusztul-e a fonal ha megszakad a kapcsolata a gombatesttel.	MUST	Feladatleírás	-	
GMB005	Megfelelő mennyiségű spóra hatására, új gombatest keletkezik egy fonalon.	Megnézzük új gombatest keletkezik-e.	MUST	Feladatleírás	Gombák irányítása	
GMB006	A gombatest adott számú spóradozás után meghal.	Megnézzük meghal-e a gombatest.	MUST	Feladatleírás	Gombák irányítása	

GMB007	A spóradobás energiába kerül.	Megnézzük, kevesebb energiája lesz-e, ha spórát dobunk.	SHOULD	Csapat	Gombák irányítása	
SPO001	Többféle spóra létezik, melyeket a gomba testek dobhatnak szomszédos tektonra.	Gombatestből spórát dobunk, szomszédos tektonra.	MUST	Feladatleírás	Gombák irányítása	
SPO002	A spóra fajták különböző energiába kerülnek.	Spóradobásnál más spóra fajtáknál különböző értékű energiát von le.	SHOULD	Csapat	-	
SPO003	A spórafajták különböző tápértékkel rendelkeznek.	Rovar különböző spórák elfogyasztásánál más-más tápértéket kap.	MUST	Feladatleírás	-	
SPO004	Olyan spóra, amely megevésekor <b>gyorsítja</b> a rovarokat. (kisebb tápérték)	Megnézzük, hogy a rovar gyorsabb lesz-e, ha megeszi.	MUST	Feladatleírás	Spóra dobása	
SPO005	Olyan spóra, mely megevésekor <b>lassítja</b> a rovarokat. (nagyobb tápérték)	Megnézzük, hogy a rovar lassabb lesz-e, ha megeszi.	MUST	Feladatleírás	Spóra dobása	
SPO006	Olyan spóra, mely megevésekor <b>megbénítja</b> a rovarokat néhány körre. (nagyobb tápérték)	Megnézzük, hogy a rovar megbénul-e, ha megeszi.	MUST	Feladatleírás	Spóra dobása	
SPO007	Olyan spóra, mely megevése után a rovarok egy ideig <b>nem képesek fonalat vágni</b> . (közepes tápérték)	Megnézzük, hogy a rovar nem tud-e fonalat elvágni ha megeszi	MUST	Feladatleírás	Spóra dobása	



MAP001	A pálya egymással összekötött tektonokat tartalmaz, melyek között víz van.	Ellenőrizzük, hogy össze vannak-e kötve a tektonok	MUST	Feladatleírás	-	
MAP002	A játék indításakor véletlenszerű pálya generálódik, melyben a tektonok és kapcsolataik randomizáltak.	Megnézzük, hogy többféle pálya generálódik-e.	SHOULD	Csapat	Játék kezdése	
MAP003	Egyes tektonfélék kisebb/nagyobb valószínűséggel generálódnak, mint mások.	Megnézzük, hogy ugyanolyan arányban vannak-e jelen a pályán.	SHOULD	Csapat	Játék kezdése	
MAP004	Minden játékosnak van kezdésből néhány gombája és fonala.	Megnézzük, hogy új játéknál a gombászok gombákkal kezdenek-e.	SHOULD	Csapat	Játék kezdése	
MAP005	Kezdetkor legyenek olyan gombák is, amelyek egyik játékoshoz sem tartoznak.	Megnézzük, hogy vannak-e olyan gombák a pályán amik nem tartoznak egyik játékoshoz sem.	MAY	Csapat	Játék kezdése	
GME001	A játékosok körökre osztva felváltva jönnek egymás után.	Megnézzük, hogy tényleg egymás után jönnek-e a játékosok.	SHOULD	Csapat	Kör vége	
GME002	A játék előre meghatározott számú kör után véget ér.	Megnézzük véget ér a játék X db kör után.	SHOULD	Csapat	Kör vége	
GME003	A játék végén az a játékos a nyertes, akinek a legtöbb gombatest volt összesen, illetve a legtöbb tápanyagot fogyasztotta el.	Megnézzük, hogy az lett-e a nyertes, aki eleget tett ezeknek a feltételeknek.	MUST	Feladatleírás	-	

**2.3.2 Erőforrásokkal kapcsolatos követelménye**

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
LOGH001	A feladatot Java nyelven készítjük el.	Leadások előtt fordítás, futtatás, csapattagok felülvizsgálata.	MUST	Tárgykövetelmény, csapat döntés	
LOGH002	A forrásprogramnak a kari felhőben biztosított környezetben, az ott megtalálható JDK alatt lefordíthatónak és futtathatónak kell lennie.	Leadások előtt a felhőben futtatás.	MUST	Tárgykövetelmény	
LOGH003	Alapvető perifériák (egér, billentyűzet).	Futtatás előtt a perifériák ellenőrzése.	SHOULD	Csapat döntése.	
LOGH004	Megfelelő mennyiségű szabad tárhely.	Futtatás előtti szabad tárhely lekérése.	SHOULD		Ellenkező esetben a program elindulhat, de a leállásmentes működés nem garantált.
LOGH005	Windows 10-en ajánlott a futtatás.	Beadás előtt Windows 10 operációs rendszeren futtatás.	MAY	Csapat döntése.	Más operációs rendszeren is futthat, de ez nem garantált.
LOGH006	Meg kell adni valamennyi installálási útmutatást és kezelési leírásokat.	Beadás előtt a csapattagok ellenőrzik.	MAY	Tárgykövetelmény	

**2.3.3 Átadással kapcsolatos követelmények**

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
LOGS001	A dokumentumokat a “Hercules” feladat beadó rendszerben le kell adni a határidőig.	Részfeladatért felelős csapattag ellenőrzi.	SHOULD	Tárgy Oldal	
LOGS002	A dokumentumokat nyomtatottan, átlátszó műanyag irattartóban le kell adni az I épület földszintjén, általában hetfőnként.	A feladatot leadó csapattag ellenőrzi.	SHOULD	Tárgy Oldal	
LOGS003	A dokumentum fedőlapja a minta szerinti.	Részfeladatért felelős csapattag ellenőrzi.	MUST	Tárgy Oldal	
LOGS004	A dokumentum tartalmazza a naplót.	Részfeladatért felelős csapattag ellenőrzi.	MUST	Tárgy Oldal	
LOGS005	A napló tartalmazza az előző beadás óta eltelt időszak történéseit időrendben. A naplóból egyértelműen ki kell derülnie, hogy az egyes anyagrészeket ki és mennyi idő alatt készítette.	Minden csapattag felelős a saját naplóbejegyzéseiért.	MUST	Tárgy Oldal	
LOGS005	Minden beadandó anyagot az egyes feladatokhoz elérhető sablonok alapján kell elkészíteni.	Részfeladatért felelős csapattag ellenőrzi.	MUST	Tárgy Oldal	
LOGS006	Valamennyi lap oldalszámozással ellátott, tartalmazza az anyag címét, a csapat nevét és a címlapon szereplő dátumot.	Részfeladatért felelős csapattag ellenőrzi.	SHOULD	Tárgy Oldal	

**2.3.4 Egyéb nem funkcionális követelmények**

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
LOGH001	A szoftvernek könnyen tesztelhetőnek kell lennie.	Tesztek elkészítése.	SHOULD	Tárgy oldal	
LOGH002	A forráskód nem függhet külső könyvtártól.	Leadások előtt a csapattagok átnézik a kódot.	MUST	Tárgy oldal	
LOGH003	A program a felhasználó által kiadott utasításokat végrehajtja.	Futtatás, utasítások eredményének ellenőrzése.	MAY	Tárgy oldal	

## 2.4 Lényeges use-case-ek

### 2.4.1 Use-case leírások

<b>Use-case neve</b>	Új játék indítása
<b>Rövid leírás</b>	A felhasználó indíthat új játékot 4, 6, vagy 8 játékosos módban.
<b>Aktorok</b>	Felhasználó(k)
<b>Forgatókönyv</b>	A felhasználó megnyomja a “játék indítása” gombot, és kiválasztja a játékosok számát, illetve egyéb beállításokat. Ekkor új pálya generálódik, majd elkezdődik az első játékos köre.

<b>Use-case neve</b>	Kör vége
<b>Rövid leírás</b>	A felhasználó a saját körét bármikor befejezheti annak során.
<b>Aktorok</b>	Felhasználó(k)
<b>Forgatókönyv</b>	A felhasználó megnyomja a “kör vége” gombot, és ezzel a köre véget ér, a következő játékos jön.

<b>Use-case neve</b>	Gombász: spóra dobása
<b>Rövid leírás</b>	Egy gombász játékos a saját körében spórát dob egy szomszédos tektonra energiáért cserébe.
<b>Aktorok</b>	Felhasználó(k)
<b>Forgatókönyv</b>	A gombász a saját körében kiválasztja a “spóra dobása” akciót, és amennyiben van elég energiája, egy szomszédos tektont, illetve a spóra fajtáját. Itt egy új spóra jelenik meg.

<b>Use-case neve</b>	Gombász: gomba fejlesztése
<b>Rövid leírás</b>	Egy gombász játékos a saját körében kiválaszt egy gombát, mely elegendő energiáért cserébe szintet lép.
<b>Aktorok</b>	Felhasználó(k)
<b>Forgatókönyv</b>	A gombász a saját körében kiválasztja a “szintlépés” akciót, és amennyiben van elég energiája, egy gombatestet. Ez a gombatest szintet lép.

<b>Use-case neve</b>	Gombász: fonal növesztése
<b>Rövid leírás</b>	Egy gombász játékos a saját körében kiválaszt olyan tektont, amely mellett van gombafonala. Itt új gombafonal nő.
<b>Aktorok</b>	Felhasználó(k)
<b>Forgatókönyv</b>	Egy gombász játékos a saját körében kiválaszt olyan tektont, amely mellett van gombafonala. Itt új gombafonal nő.

<b>Use-case neve</b>	Rovarász: rovar irányítása
<b>Rövid leírás</b>	Egy rovarász a saját körében kiválaszt egy, a rovarával fonalak mentén szomszédos tektont, és ide lépteti a rovarját.

<b>Aktorok</b>	Felhasználó(k)
<b>Forgatókönyv</b>	Egy rovarász a saját körében kiválaszt egy, a rovarával fonalak mentén szomszédos tektont, és ide lépteti a rovarját.

<b>Use-case neve</b>	Rovarász: fonal elvágása
<b>Rövid leírás</b>	A rovarász a saját körében elvághat egy olyan gombafonalt, amelyen éppen a rovara áll.
<b>Aktorok</b>	Felhasználó(k)
<b>Forgatókönyv</b>	Egy rovarász a saját körében a “fonal elvágása” opciót választja. Ekkor az(ok) a fonal(ak) melyen áll elvágódnak.

<b>Use-case neve</b>	Rovarász: spóra megevése
<b>Rövid leírás</b>	Egy rovarász a saját körében megehet egy olyan spórát, amely azonos tektonon van a rovarával.
<b>Aktorok</b>	Felhasználó(k)
<b>Forgatókönyv</b>	Egy rovarász a saját körében a “spóra megevése” opciót választja. Ekkor az azonos tektonon lévő spórákat a rovar megeszi.

## 2.4.2 Use-case diagram

Rovarok irányítása diagram:

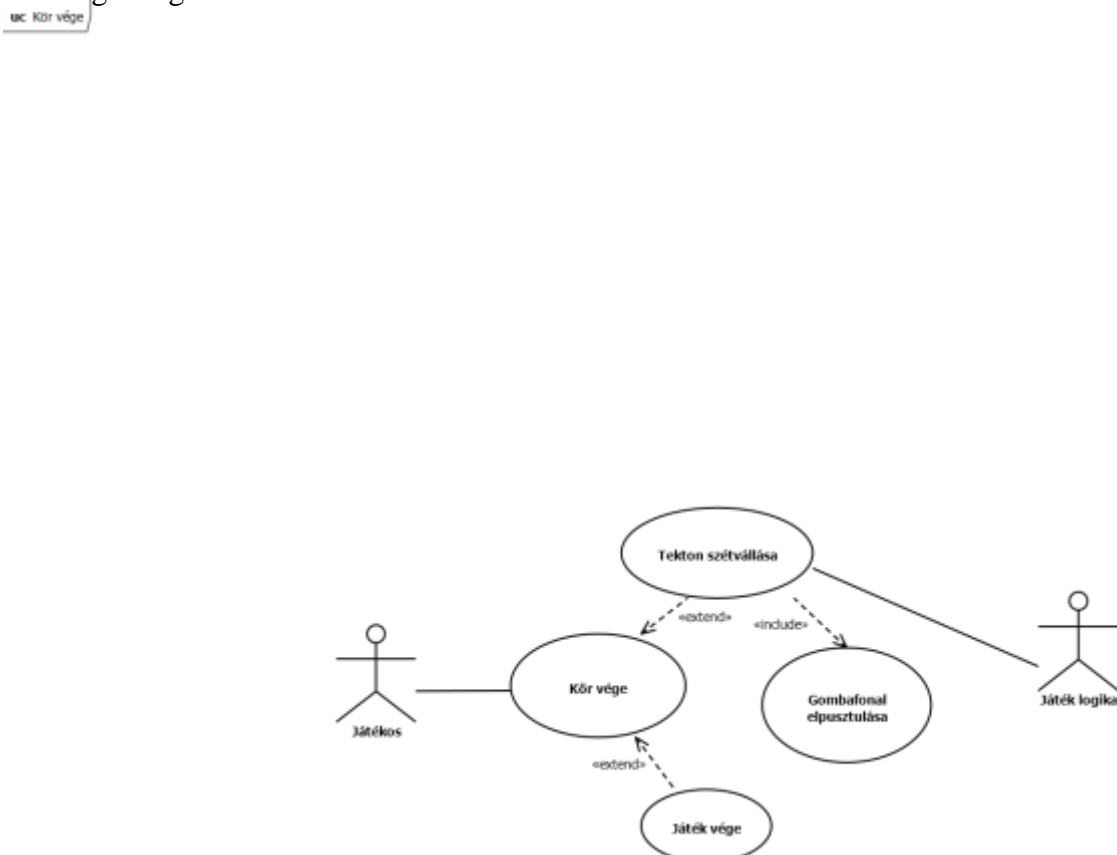
uc: Rovar irányítása



Gombák irányítása diagram:



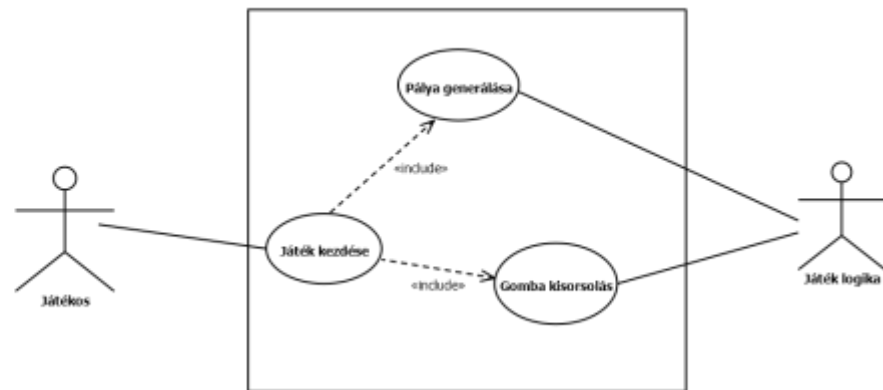
Kör vége diagram:





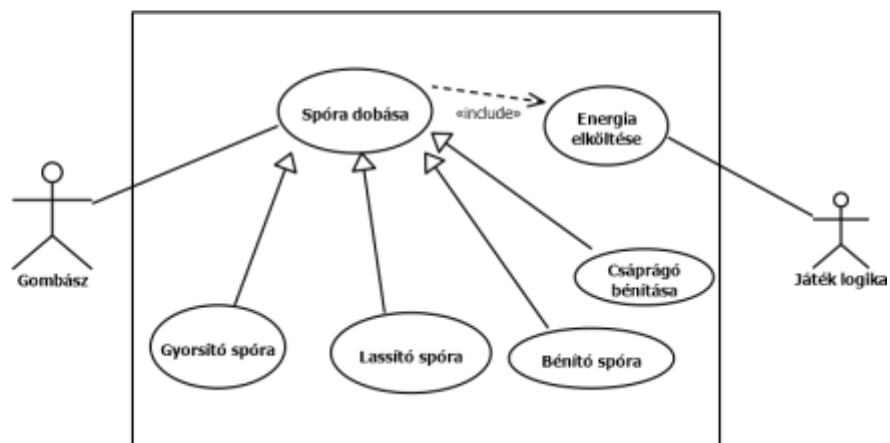
Játék kezdése diagram:

uc Új játék



Spóra dobása diagram:

uc Spóra dobás



**2.5 Szótár**

Energia	Gombászok elsődleges erőforrása
Fungorium	A kitalált bolygó, ahol a játék elhelyezkedik
Gombafajta	Egy gombász által növesztett összes organizmus
Gombafonal	Gombatestekből nő és a tektonok közötti áthidalást teszi lehetővé
Gombakolónia	Azok az egy gombafajtába tartozó organizmusok, amik gombafonallal össze vannak kötve
Gombász	Egy gombafajtát irányít
Gombatest	Tektonokra építhetik gombászok, az számít a legjobb gombásznak akinek a legtöbb gombatestet nőtt
Rovar	Életcélja, hogy minél több tápanyagot fogyasztson el
Rovarász	Egy rovar irányít
Spóradobás	Spórák eljuttatása gombatestből egy szomszédos (fejlettebb gombatest esetén akár távolabbi) tektonra
Spóra	Gombatestek reprodukcióját teszik lehetővé, valamint a rovarok meg tudják enni, ezzel tápanyagot és esetleg pozitív illetve negatív hatásokat szerezhetnek
Tápanyag	Spórák rendelkeznek tápanyaggal, az számít a legjobb rovarásznak, akinek a rovára a legtöbbet fogyasztja el
Tekton	Ezekon helyezkedik el Fungorium élővilága
Víz	A tektonok közötti résen víz található

## **2.6 Projekt terv**

Feladat	Felelős személy	Feladat határideje
---------	-----------------	--------------------

atalakítás	-	14. 12:00
etelmény, projekt, funkcionalitás	us	24. 14:15
lízis modell (I. változat)	z	c. 3. 14:15
lízis modell (II. változat)	mbos	c. 10. 14:15
leton tervezése	váth	c. 17. 14:15
leton elkészítése	vári	c. 24. 14:15
otípus koncepciója	us	31. 14:15
letes tervek	z	14. 14:15
otípus elkészítése	mbos	28. 14:15
ikus változat tervei	váth	5. 14:15
ikus változat elkészítése	vári	19. 14:15
esített dokumentáció	us	23. 12.00

Részenként a felelős személyeket csapatmegbeszélésen választottuk ki- ők bonyolítják le a rész feladattal kapcsolatos főbb tevékenységeket.

A dokumentumot Google Docs segítségével írjuk, legfőképp az egyidejű szerkeszthetőség és a verziókezelés miatt.

A forráskódokat a GitHub felületen teszik közzé a csapattagok.

A fejlesztőkörnyezetet nem kötjük meg, minden csapattag választhat.

**2.7 Napló**

<b>Kezdet</b>	<b>Időtartam</b>	<b>Résztevők</b>	<b>Leírás</b>
2025.02.18. 12:00	1 óra	Bohus Galambos Horváth Végvári	Értekezlet, feladat értelmezése. Döntés: Meetingek hétfő délután és péntek délelőtt lesznek.
2025.02.18. 16:30	1 óra	Bohus	Tevékenység: Funkciók írása
2025.02.18. 16:30	0.5 óra	Horváth	Tevékenység: Funkciók írása
2025.02.19. 12:00	2 óra	Bohus Galambos Horváth Végvári Dúcz	Értekezlet: Döntés: Feladat megoldásának szétosztása a csapattagok között
2025.02.19. 16:00	1 óra	Horváth	Tevékenység: Áttekintések, követelmények megírása, már megírt részek kiegészítése
2025.02.19 17:50	1 óra	Galambos	Tevékenység: Követelmények írása
2025.02.19 20:00	2 óra	Végvári	Tevékenységek: követelmények, funkcionalitás írása
2025.02.20 12:00	1 óra	Végvári	Tevékenységek: követelmények, funkcionalitás írása
2025.02.20 15:00	2 óra	Horváth	Tevékenységek: Projekt terv írása Követelmények írása
2025.02.20 11:00	2.5 óra	Bohus	Tevékenységek: funkciók átnézése, use case diagramok írása

## 2. Követelmény, projekt, funkcionalitás

totoro

2025.02.21 11:00	2 óra	Dúcz	Tevékenységek: hibák javítása, követelmények írása
------------------	-------	------	--

2025.02.21 14:00	1 óra	Bohus Galambos Horváth Végvári Dúcz	Értekezlet. Döntések: játékosok számossága, rovar számossága, új játék use case
2025.02.21 19:00	1 óra	Dúcz	Tevékenységek: use-casek írása, átnézés, hibajavítás
2025.02.21 20:00	1 óra	Végvári	Tevékenységek: Szótár
2025.02.22 18:00	0.5 óra	Végvári	Tevékenységek: Funkcionális követelmények írása
2025.02.22 20:00	2 óra	Horváth	Tevékenységek: Dokumentum részletes áttekintése, formázása Hibák javítása
2025.02.23. 10:00	1 óra	Galambos	Tevékenység: Szótár írása, dokumentum áttekintése
2025.02.23 12:00	0.5 óra	Végvári	Tevékenységek: Diagramok írása
2025.02.23 14:00	0.5 óra	Bohus	Tevékenységek: Diagramok írása



# Tab 1

### 3. Analízis modell (II. változat)

23 – totoro

Konzulens:

Haragos Gergő Viktor

#### Csapattagok

**Bohus János**

Dúcz Ákos

Galambos Csaba

Horváth

Avarka

Végvári Péter

**XYOVFZ**

GC1RTE

BVBQKN

OODMXV

AW3JH1

**janos.bohus1@gmail.com**

akos.ducz@gmail.com

galambos.csaba05@gmail.com

avarkahorvath@gmail.com

vegvaripeter21@gmail.co

m

2025.02.24

## 3. Analízis modell kidolgozása

### 3.1 *Objektum katalógus*

#### 3.1.1 Pálya

Itt helyezkedik el a játék összes eleme. Tektonokból áll, a tektonok között pedig víz van. Felelőssége a pályán lévő tektonokat számon tartani.

#### 3.1.2 Tekton

A tektonok a pálya alapelemei, egy tektonon egy gombatest, gombafonalak, illetve rovarok lehetnek. Egy tekton szomszédos lehet más tektonokkal. Nyilvántartja tehát, hogy mik van rajta, és milyen szomszédjai vannak. Minden kör végén ketté válhatnak. Több fajta tekton létezik, különböző tulajdonságokkal.

#### 3.1.3 Fonalelnyelő tekton

Olyan tektonon, ahonnan a fonalak pár kör után felszívódnak.

#### 3.1.4 Gombamentes tekton

Olyan tekton, amin nem tud gombatest nőni.

#### 3.1.5 Egyfonalas tekton

Olyan tekton, amin csak egy fonál lehet.

#### 3.1.6 Gombatest

A gombász egy tektonon tudja elhelyezni, ha van ott fonal és megfelelő mennyiségű spóra található ott. Energiát termel minden turn-re. A termelt energiát spórák kilövésére tudja költeni, szomszédos tektonra és ha elég spóra van egy tektonon akkor ott nő egy új gombatest. Minél nagyobb szintű, annál több energiát termel a gombatest, és annál messzebbre tudja kilögni spóráit. Adott számú spóra kilövelés után elpusztul.

#### 3.1.7 Fonál

A tektonokon helyezkedik el, és szomszédos tektonokat köt össze. Minden gombásznak saját fonaltípusa van. Két szomszédos tekton akkor van összekötve egy adott fonaltípus által, ha mindkettőn van adott típusú fonal. Ha egy tektonon van spóra akkor arra gyorsabban nő gombafonal.

#### 3.1.8 Gombakolónia

Egy gombafaj összefüggő gombatestjei és fonalai alkotják.

#### 3.1.9 Gyorsító spóra

Ha rovar megeszi, akkor több gombafonalon tud áthaladni a turn-jében.

#### 3.1.10 Lassító spóra

Ha rovar megeszi, akkor kevesebb gombafonalon tud áthaladni a turn-jében.

#### 3.1.11 Bénító spóra

Ha rovar megeszi, akkor pár round-ig le lesz bénulva.

### **3.1.12 Csáprágó bénító spóra**

Ha rovar megeszi, akkor pár round-ig le lesz bénulva a csáprágója, nem fog tudni gombafonalakat elvágni.

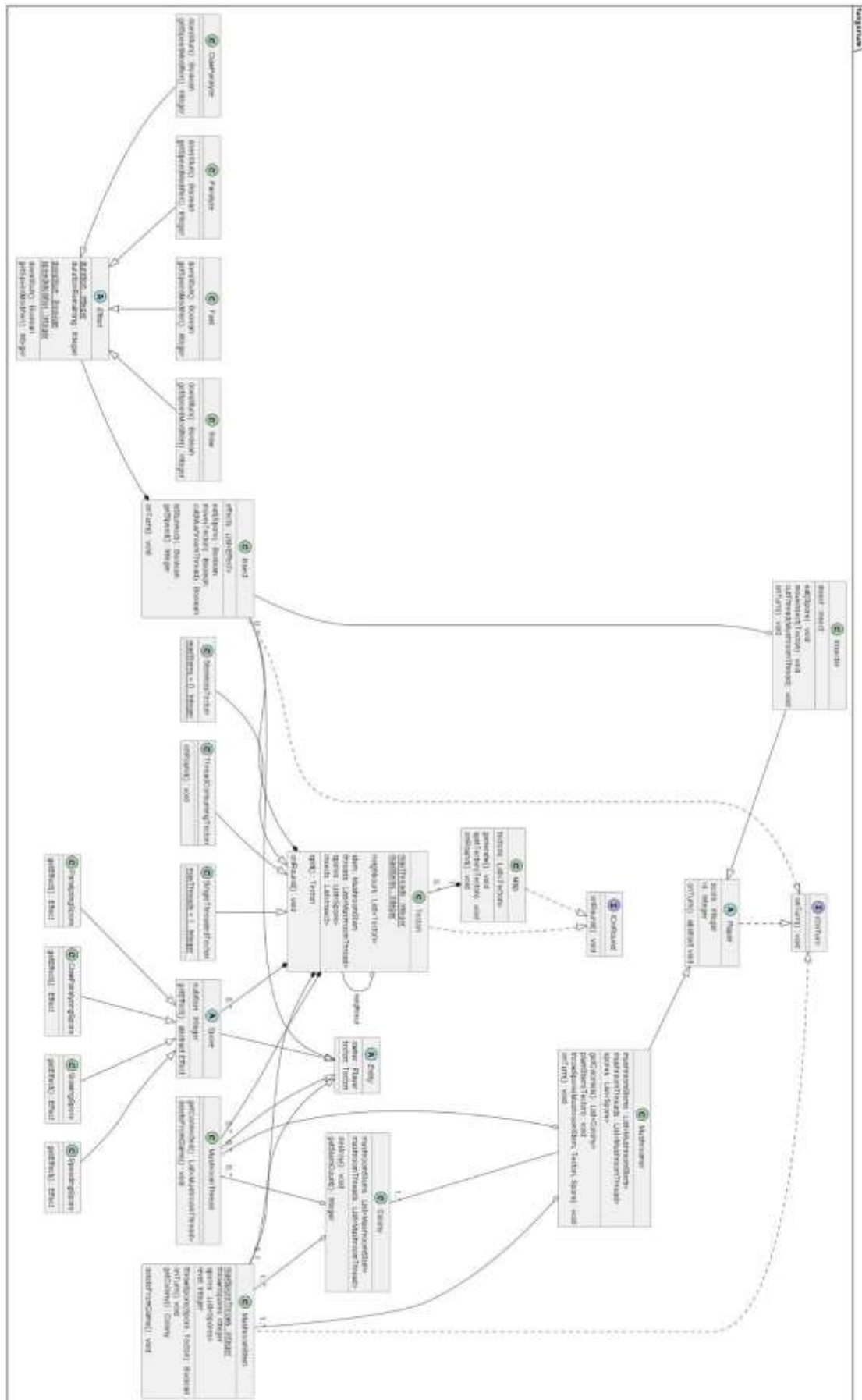
### **3.1.13 Rovar**

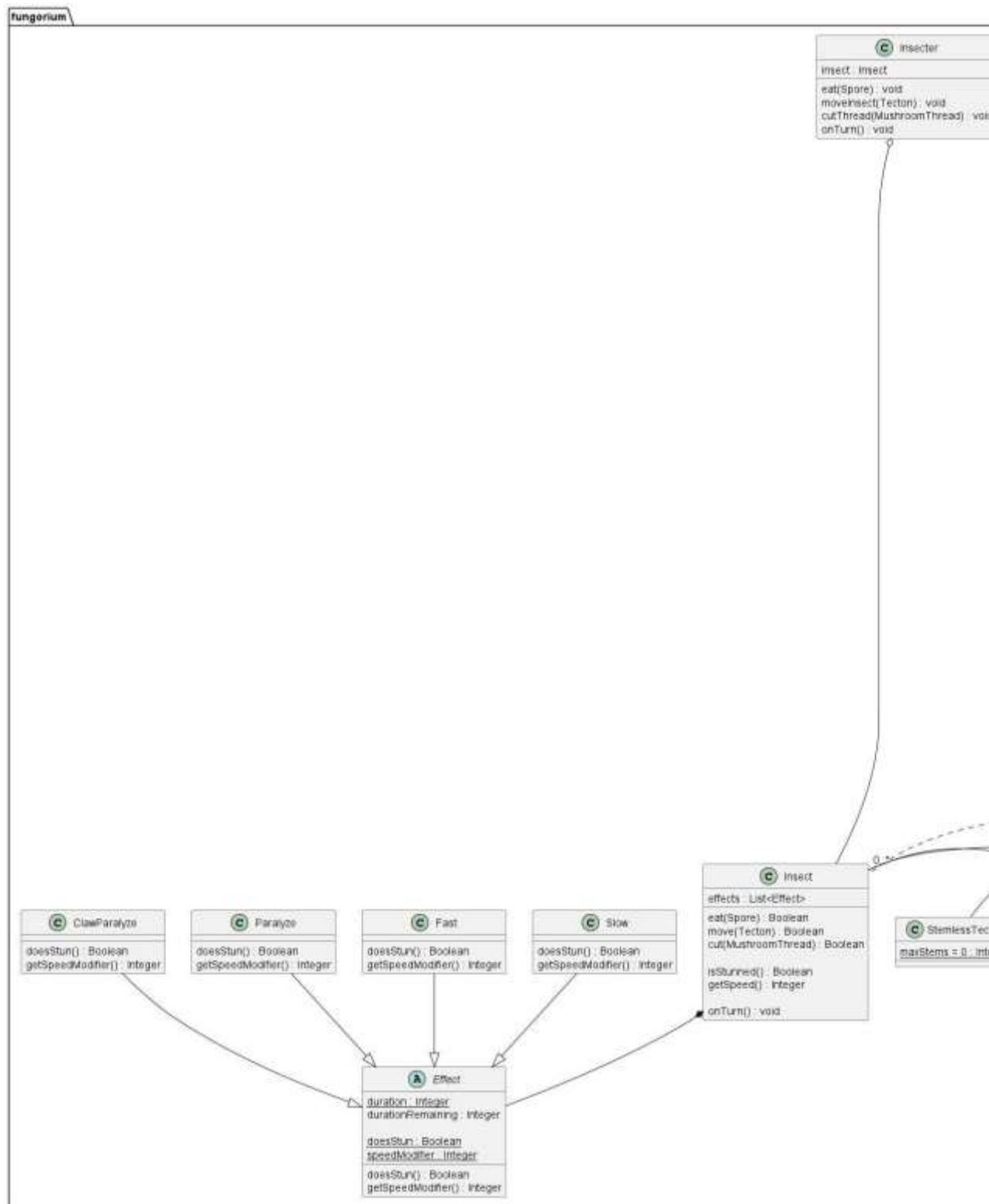
A rovarok a tektonokon vannak, tudnak mozogni, spórát enni, illetve fonalakat elvágni. Tudják melyik tektonon vannak.

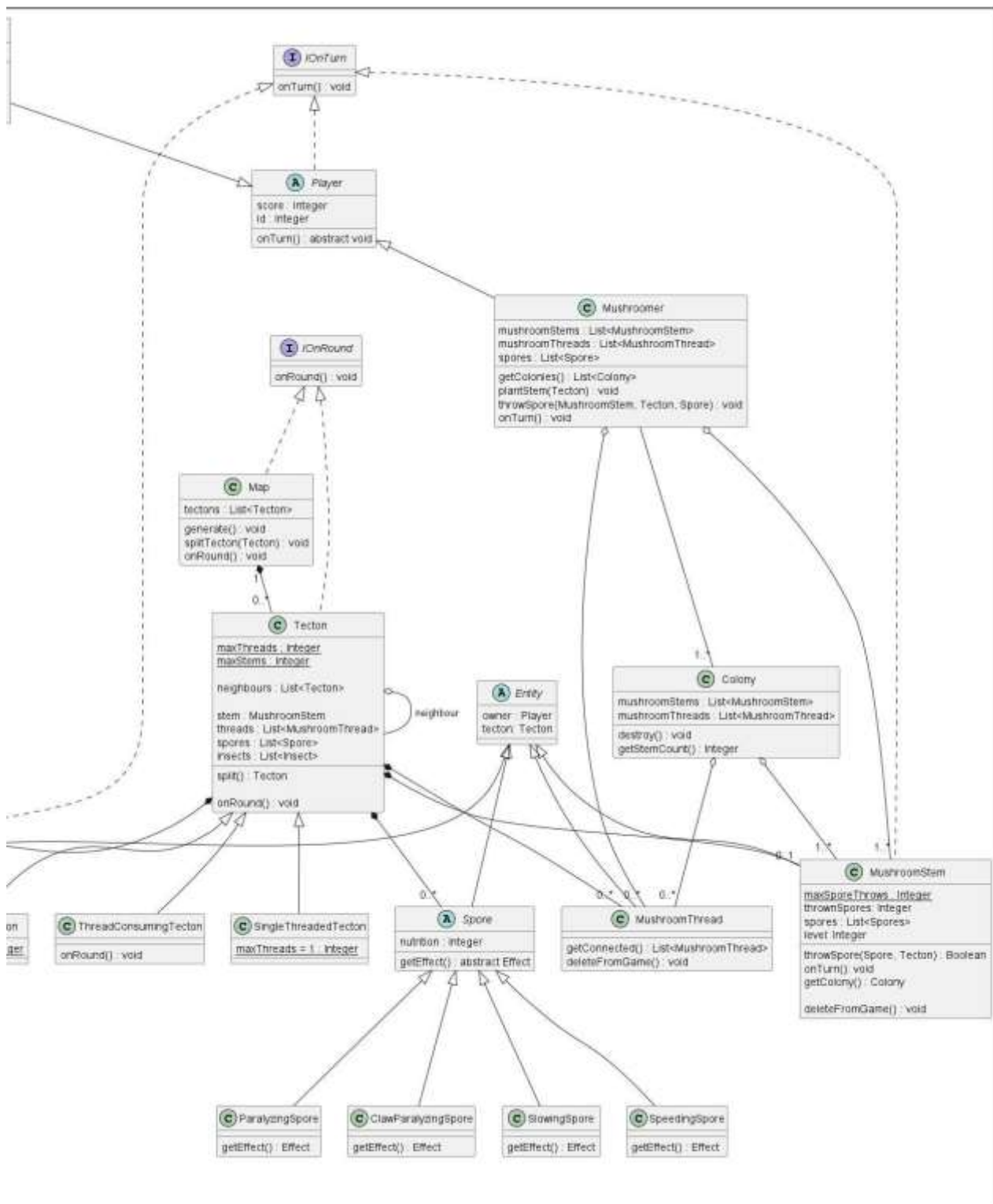
### **3.1.14 Rovarász**

A rovarászok egy rovart irányítanak.

A rovarász játékosok egymással versenyeznek. Az a rovarász nyer, aki a legtöbb tápanyagot fogyasztotta el.







### 3.3 Osztályok leírása

#### 3.1.1 ClawParalyze

- **Felelősség**

Az osztály egy effektus, amely a rovarokra képes hatni. A hatása alatt lévő rovar csáprágója bénított állapotban van, így nem képes gombafonalat elrágni, vagy táplálékot fogyasztani.

- **Ősosztályok**

Effect → ClawParalyze

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **Boolean doesStun():** Megmondja, hogy lefagyasztja-e az effektus a rovar csáprágóját.
- **Integer getSpeedModifier():** Visszaadja, hogy milyen módon manipulálja az effektus a rovar sebességét.

#### 3.1.2 ClawParalyzingSpore

- **Felelősség**

Egyfajta spóra, tektonokon van. Ha egy rovar megeszi, akkor pár körig nem tudja használni a csáprágóját, tehát nem tud enni, vagy gombafonalat elrágni.

- **Ősosztályok**

Entity → Spore → ClawParalyzingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **Effect getEffect():** Ha egy rovar megeszi, akkor a csáprágója bénítva lesz pár körig.



### 3.3.1 Colony

- **Felelősség**

A Colony osztály feladata, hogy egy gombakolónia összefüggő részét kezelhetővé tegye.

- **Össztályok**

Az osztálynak nincs őse.

- **Interfészek**

A Colony osztály nem valósít meg interfészt.

- **Asszociációk**

- **Aggregáció1:** Az összefüggő részben lehetnek gombatestek, így MushroomStem a túloldali szereplő.
- **Aggregáció2:** A Gombakolóniába tartoznak a gombafonalak, így MushroomThread a túloldali szereplő.
- **Asszociáció:** Asszociációban van a Mushroomer osztállyal, hiszen egy gombász játékosnak egyszerre legalább 1 kolóniája van.

- **Attribútumok**

- **List<MushroomStem> mushroomStems:** Az összefüggő részben lévő gombatestek.
- **List<MushroomThread> mushroomThreads:** Az összefüggő részben lévő gombafonalak.

- **Metódusok**

- **void destroy():** A kolóniában található összes gombatestet és gombafonalat eltávolítja a térképről.
- **Integer getStemCount():** Megmondja, hogy hány gombatest található a kolóniában.

### 3.1.3 Effect

- **Felelősség**

Absztrakt osztály. Effektust képvisel, azaz olyan hatásokat, amely alatt a rovarok lehetnek valamilyen spóra megevése után.

- **Össztályok**

Nincs őssztálya.

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **static Integer duration:** Az effektus hatásának az ideje. Statikus tagváltozó.
- **Integer durationRemaining:** Az effektus hatásának idejéből hátralévő idő.
- **static Boolean doesStun:** Tárolja, hogy az effektus bénítja-e a csáprágót. Statikus tagváltozó.
- **static Integer speedModifier:** Tárolja, hogy az effektus mennyire változtatja meg a rovar mozgási sebességét. Statikus tagváltozó.

- **Metódusok**
  - **Boolean doesStun()**: Megmondja, hogy az effektus bénítja-e a csáprágót.
  - **Integer getSpeedModifier()**: Megmondja, hogy az effektus mennyire változtatja meg a rovar mozgási sebességét.

### 3.1.4 Entity

- **Felelősség**

Absztrakt osztály. Minden olyan objektum ebből származik le, amit birtokol egy játékos.

- **Ősosztályok**

Nem származik le egy osztályból sem.

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Player owner**: Az a játékos, aki birtokolja az objektumot
- **Tecton tecton**: Az entitást tartalmazó tecton.

- **Metódusok**

### 3.1.5 Fast

- **Felelősség**

Az osztály egy effektus, amely a rovarokra képes hatni és a gyorsító hatást reprezentálja.

- **Ősosztályok**

Effect → Fast

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **Boolean doesStun():** Megmondja, hogy az effektus bénítja-e a csáprágót.
- **Integer getSpeedModifier():** Megmondja, hogy az effektus mennyire változtatja meg a rovar mozgási sebességét.

### 3.1.6 Insect

- **Felelősség**

Az Insect osztály egy rovarot reprezentál, melyet egy rovarász irányít. Képes fonalakat elvágni, spórékat megenni és mozogni a tektonok között, a fonalakon. Tudja, melyik tektonon van.

- **Ősosztályok**

Entity → Insect

- **Interfészek**

Az Insect osztály az IOnTurn interfészt valósítja meg.

- **Asszociációk**

- **kompozíció:** A rovarok különböző hatások alatt lehetnek, így a túloldali szereplő az Effect osztály.

- **Attribútumok**

- **List<Effect> effects:** Azokat a hatásokat tárolja, amik alatt a rovar éppen van.

- **Metódusok**

- **void eat(Spore Sp):** Megeszik egy spórát, eltávolítva azt a tektonról.
- **void move(Tecton T):** A rovar átmegy egy másik tektonra.
- **void cut(MushroomThread Mt):** A rovar elvág egy fonalat azon tektonon, amelyen tartózkodik.
- **Boolean isStunned():** Megmondja, hogy a rovar csáprágója működik-e.
- **Integer getSpeed():** Megmondja, hogy milyen sebességgel tud mozogni a rovar.
- **void onTurn():** Arra szolgál, hogy a lépés utáni tevékenységeket végrehajtsa.

### 3.1.7 Insecter

- **Felelősség**

A rovarász típusú játékosokat kezelő osztály.

- **Ősosztályok**

Player □ Insecter

- **Interfészek**

Az Insecter osztály nem valósít meg interfészt.

- **Asszociációk**

- **Aggregáció:** Tartalmazza a játékoshoz tartozó Insect objektumot.

- **Attribútumok**

- **Insect insect:** A játékos által irányított rovar objektum.

- **Metódusok**

- **void eat(Spore sp):** A rovar spórát eszik.
- **void moveInsect(Tecton t):** Egy tektonra mozgatja a rovar.
- **void cutThread(MushroomThread mt):** A rovar elvág egy fonalat.
- **void onTurn():** Arra szolgál, hogy a lépés utáni tevékenységeket végrehajtsa.

### 3.1.8 IOnRound

- **Felelősség**

Interfész, azok implementálják, akik valamit csinálnak minden körben.

- **Ősosztályok**

Nincs.

- **Asszociációk**

- **Metódusok**

- **void onRound():** Az a metódus, amit végrehajt minden körnél.

### 3.1.9 IOnTurn

- **Felelősség**

Interfész, azok implementálják, akik valamit csinálnak minden lépésben.

- **Ősosztályok**

Nincs.

- **Asszociációk**

- **Metódusok**

- **void onTurn():** Az a metódus, amit végrehajt minden lépésnél.

### 3.1.10 Map

- **Felelősség**

A tektonokért felel, lényegében a pálya.

- **Ősosztályok**

Az osztálynak nincs ősosztálya.

- **Interfészek**

Az IOnRound interfészt valósítja meg.

- **Asszociációk**

- **aggregáció1**: Tecton a túloldali szereplő, hiszen a térképen tektonok vannak.

- **Attribútumok**

- **List<Tecton> tectons**: A pálya a tektonjait tárolja.

- **Metódusok**

- **void generate()**: Létrehozza a pályát véletlenszerűen. Elhelyezi a szabályoknak megfelelően a tektonokat.
- **void splitTecton(Tecton t)**: Egy tektont 2 részre bont, azaz szétszakít.
- **onRound()**: Az a metódus, amit végrehajt minden körben.

### 3.1.11 Mushroomer

- **Felelősség**

A gombász típusú játékost kezelő osztály.

- **Ősosztályok**

- Player □ Mushroomer

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

- **aggregáció1:** Tartalmazza a játékoshoz tartozó MushroomStem objektumokat.
- **aggregáció2:** Tartalmazza a játékoshoz tartozó MushroomThread objektumokat.
- **asszociáció:** Több gombakolóniája is lehet (Colony).

- **Attribútumok**

- **List<MushroomStem> mushroomStems:** A játékoshoz tartozó összes gombatest.
- **List<MushroomThread> mushroomThreads:** A játékoshoz tartozó összes gombafonál.

- **Metódusok**

- **List<Colony> getColonies():** BFS algoritmus segítségével megkeresi a fonalak mentén összefüggő gombakolóniákat, melyek a játékoshoz tartoznak, és visszaadja ezeket.
- **void plantStem(Tecton t):** Egy tektonra elhelyez egy gombatestet.
- **void throwSpore(MushroomStem, Tecton, Spore):** Egy tektonra spórát dob.
- **void onTurn():** Az a metódus, amit végrehajt minden lépésnél.

### 3.1.12 MushroomStem

- **Felelősség**

A gombatest osztály felelőssége, hogy a gombatesteket kezelje, és ezek tulajdonságainak nyilvántartása.

- **Ősosztályok**

Entity → MushroomStem

- **Interfészek**

Az IOnTurn interfészt valósítja meg.

- **Asszociációk**

- **Attribútumok**

- **static Integer maxSporeThrows:** Rögzíti, hogy hányszor tud spórát dobni a gombatest. Statikus tagváltozó.
- **Integer thrownSpores:** Nyilvántartja, hogy hányszor dobott eddig spórát.
- **Integer level:** Tárolja, hogy hányas szinten van a gombatest.

- **Metódusok**

- **Boolean throwSpore(Spore s, Tecton t):** Spórát dob.
- **Colony getColony():** Megnézi, hogy melyik kolóniában van a gombatest.
- **void onTurn():** Az a metódus, amit végrehajt minden lépésnél.

### 3.1.13 MushroomThread

- **Felelősség**

A tektonokon elhelyezkedő, és azokat összekötő gombafonalakat megvalósító osztály.

- **Ősosztályok**

Entity □ MushroomThread

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **List<MushroomThread> getConnected():** Visszaadja a tektonok mentén összekapcsolt, azonos játékoshoz tartozó fonalak listáját (BFS algoritmus segítségével).

### 3.1.14 Paralyze

- **Felelősség**

Az osztály egy effektus, amely a rovarokra képes hatni és a bénító hatást reprezentálja.

- **Ősosztályok**

Effect → Paralyze

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **Boolean doesStun():** Megmondja, hogy az effektus bénítja-e a csáprágót.
- **Integer getSpeedModifier():** Megmondja, hogy az effektus mennyire változtatja meg a rovar mozgási sebességét.

### 3.1.15 ParalyzingSpore

- **Felelősség**

Egyfajta spóra, tektonokon van. Ha egy rovar megeszi, akkor pár körig nem tud mozogni.

- **Ősosztályok**

Entity → Spore → ParalyzingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **Effect getEffect():** Ha egy rovar megeszi, akkor bénítva lesz pár körig.



### 3.1.16 Player

- **Felelősség**

Absztrakt játékos osztály, ebből származik le a gombász és rovarász osztály.

- **Ősosztályok**

Nincs ősosztálya.

- **Interfészek**

Az IOnTurn interfészt valósítja meg.

- **Asszociációk**

- **Attribútumok**

- **Integer score:** Játékos által eddig elért pontszám.
- **Integer id:** Játékos egyedi azonosítására szolgál.

- **Metódusok**

- **abstract void onTurn():** Az a metódus, amit végrehajt minden lépésnél.

### 3.1.17 SingleThreadedTecton

- **Felelősség**

Az osztály egy tekton fajta, amelyiken csak egyetlen gombafonál helyezkedhet el.

- **Ősosztályok**

Tecton → SingleThreadedTecton

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **static Integer maxThreads=1:** Statikus attribútum, mely rögzíti, hogy csak 1 gombafonál lehet rajta.

- **Metódusok**

### 3.1.18 StemlessTecton

- **Felelősség**

Az osztály egy tekton fajta, amelyiken csak egyetlen gombafonal helyezkedhet el.

- **Össztályok**

Tecton → StemlessTecton

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **static Integer maxStems=0:** Statikus attribútum, mely rögzíti, hogy 0 gombatest lehet rajta.

- **Metódusok**

### 3.1.19 Slow

- **Felelősség**

Az osztály egy effektus, amely a rovarokra képes hatni és a lassító hatást reprezentálja.

- **Össztályok**

Effect → Slow

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **Boolean doesStun():** Megmondja, hogy az effektus bénítja-e a csáprágót.
- **Integer getSpeedModifier():** Megmondja, hogy az effektus mennyire változtatja meg a rovar mozgási sebességét.

### 3.1.20 SlowingSpore

- **Felelősség**

Egyfajta spóra, tektonokon van. Ha egy rovar megeszi, akkor pár körig lassabban tud mozogni.

- **Ősosztályok**

Entity → Spore → SlowingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **Effect getEffect():** Ha egy rovar megeszi, akkor lassítva lesz pár körig.

### 3.1.21 SpeedingSpore

- **Felelősség**

Egyfajta spóra, tektonokon van. Ha egy rovar megeszi, akkor pár körig gyorsabban tud mozogni.

- **Ősosztályok**

Entity → Spore → SpeedingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **Effect getEffect():** Ha egy rovar megeszi, akkor gyorsítva lesz pár körig.

### 3.1.22 Spore

- **Felelősség**

Absztrakt osztály, a többi spóra osztály őse, egy játékban lévő spórát reprezentál. Tektonokon van, ha a tekton szétesik, eltűnik akkor a spórák is meghalnak.

- **Ősosztályok**

Entity → Spore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Integer nutrition:** Az az érték, amennyi pontot ad a rovarnak ha megeszi.

- **Metódusok**

- **abstract Effect getEffect():** Visszaadja, hogy ha egy rovar megeszi, akkor milyen hatással lesz rá.

### 3.1.23 Tecton

- **Felelősség**

A pálya tektonokból áll, ezeken helyezkedik el sok másik objektum, amiket a tektonnak kell számon tartania.

- **Interfészek**

Megvalósítja az IOnRound interfészt.

- **Asszociációk**

- **Kompozíció1:** A tekton tárolja a rajta lévő MushroomStem-et.
- **Kompozíció2:** A tekton tárolja a rajta lévő MushroomThread-eket.
- **Kompozíció3:** A tekton tárolja a rajta lévő Spore-okat.
- **Kompozíció4:** A tekton tárolja a rajta lévő Insect-eket.
- **Aggregáció:** A tekton tárolja a saját magával szomszédos tektonokat is.

- **Attribútumok**

- **MushroomStem stem:** A tektonon lévő gombatest.
- **List<MushroomThread> threads:** A tektonon lévő gombafonalakat tárolja.
- **List<Spore> spores:** A tektonon lévő spórákat tárolja.
- **List<Insect> insects:** A tektonon lévő rovarokat tárolja.
- **List<Tecton> neighbors:** tárolja a tekton szomszédjait.
- **Static Integer MaxThreads:** A tektonon lévő gombafonalak maximális számát adja meg, a legtöbb tektonon végtelen, de van olyan amin 1.
- **Static Integer MaxStems:** A tektonon lévő gombafonalak maximális számát adja meg, a legtöbb tektonon 1, de van olyan amin 0.

- **Metódusok**

- **Tecton split():** A tekton két darabra törik.
- **void onRound():** Az a metódus, ami minden körben lefut.

### 3.1.24 ThreadConsumingTecton

- **Felelősség**

Az osztály egy tekton fajta, amelyiken nem tud gombafonal megmaradni.

- **Ősosztályok**

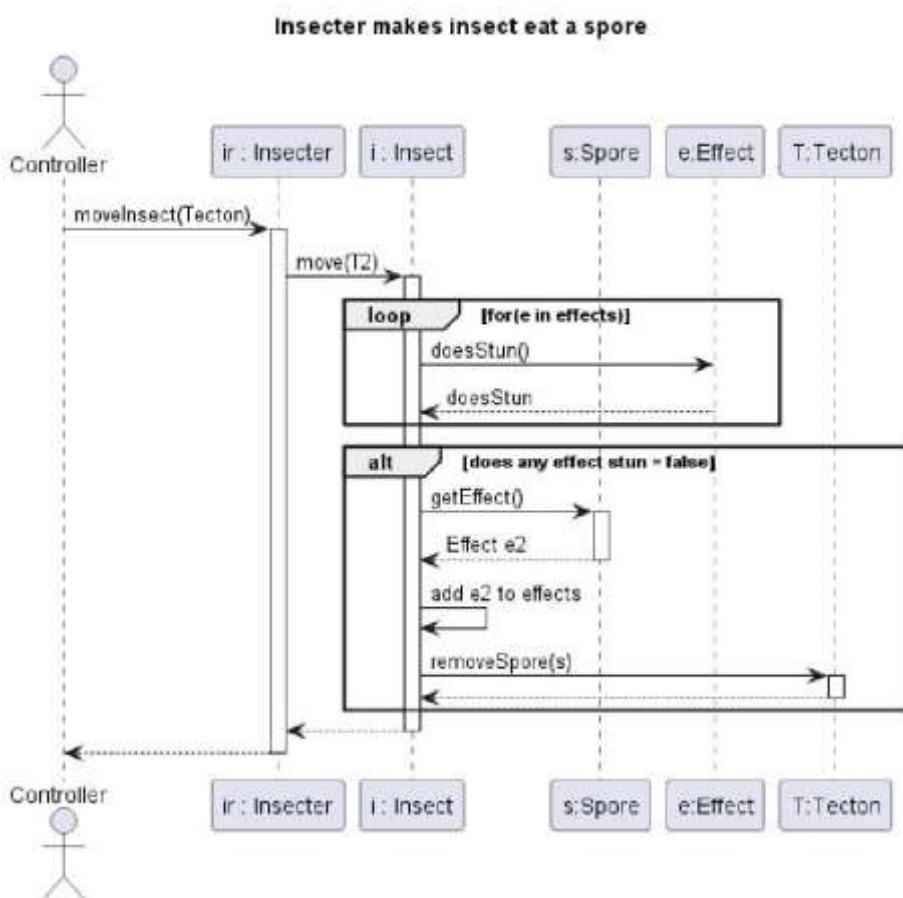
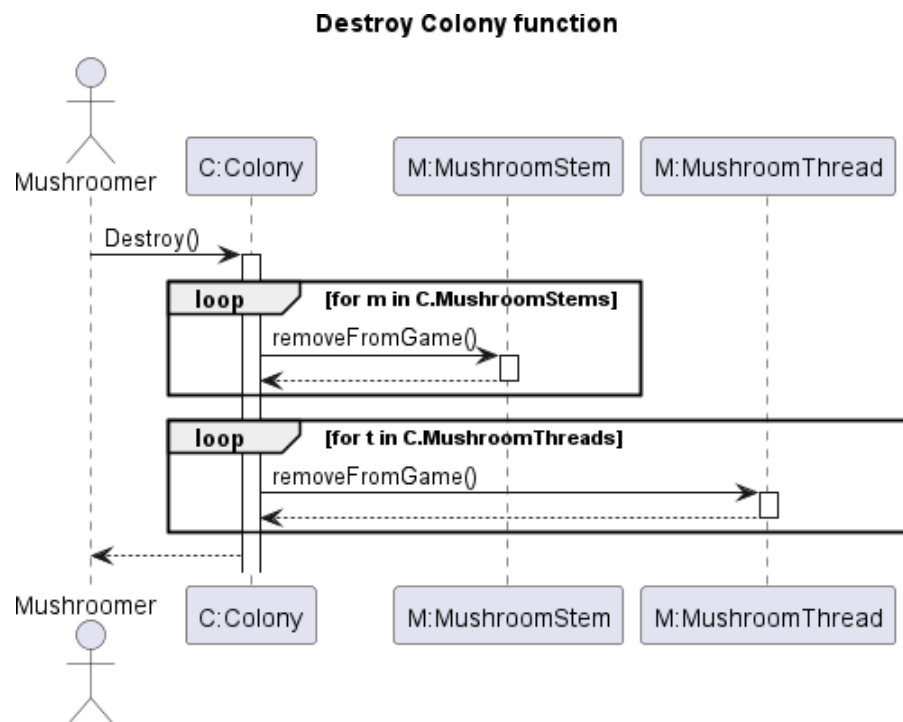
Tecton → ThreadConsumingTecton

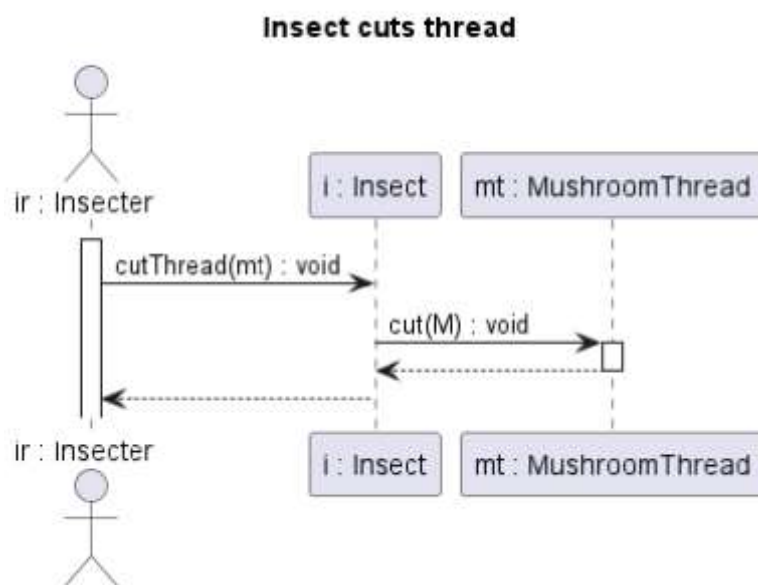
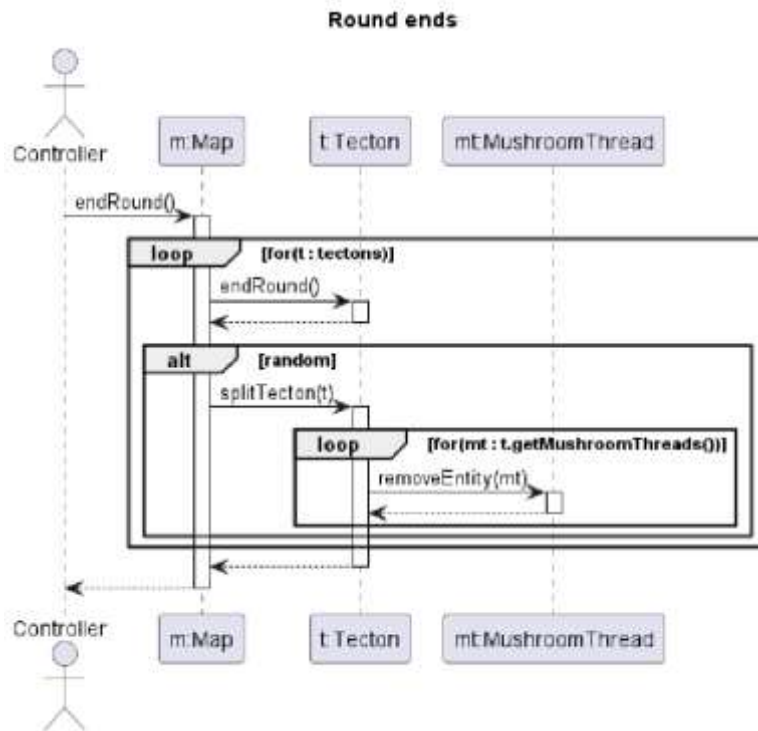
- **Interfészek**

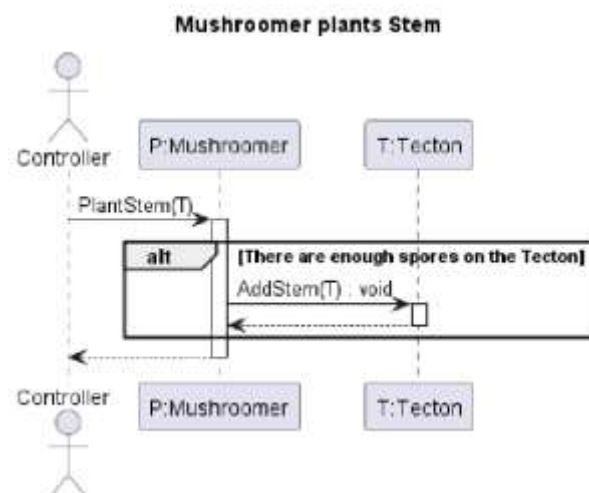
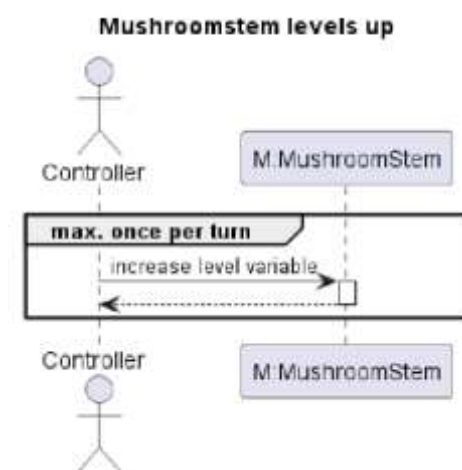
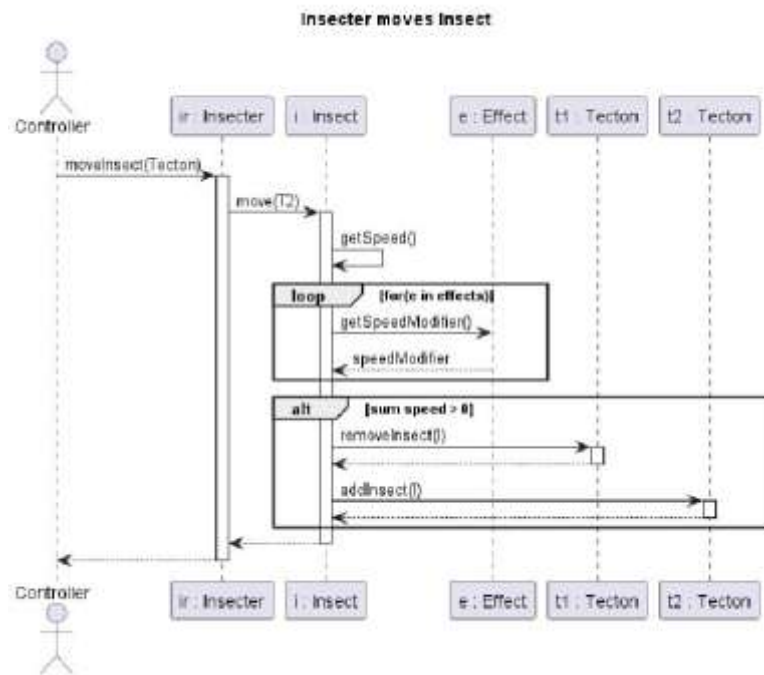
Az osztály nem valósít meg interfészt.

- **Asszociációk**
- **Attribútumok**
- **Metódusok**

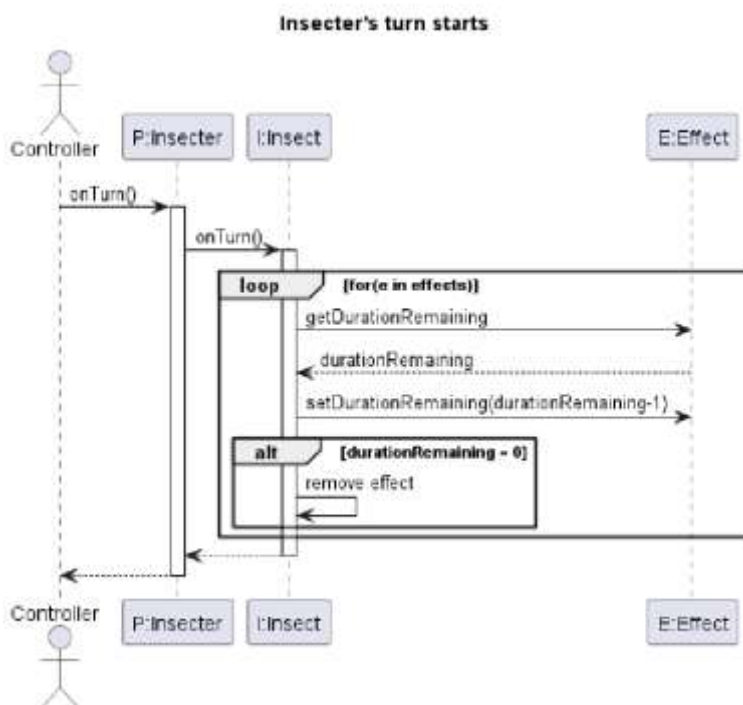
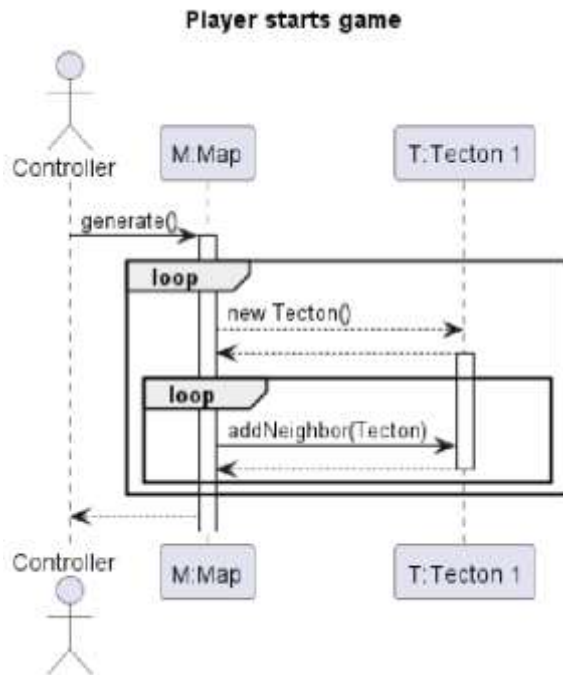
### 3.4 Szekvencia diagramok

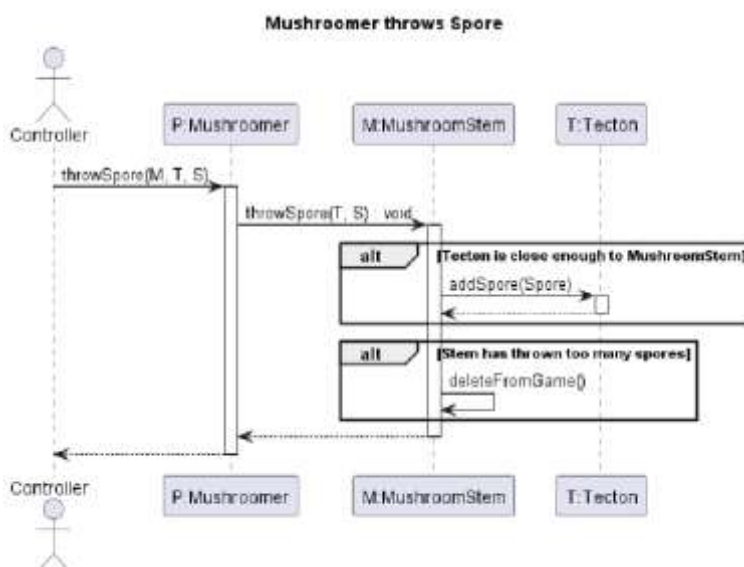
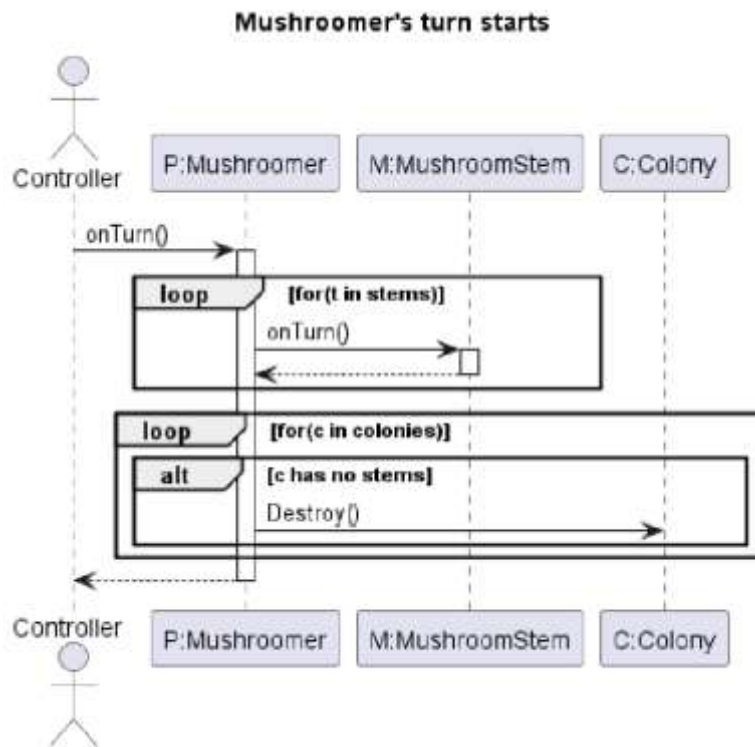








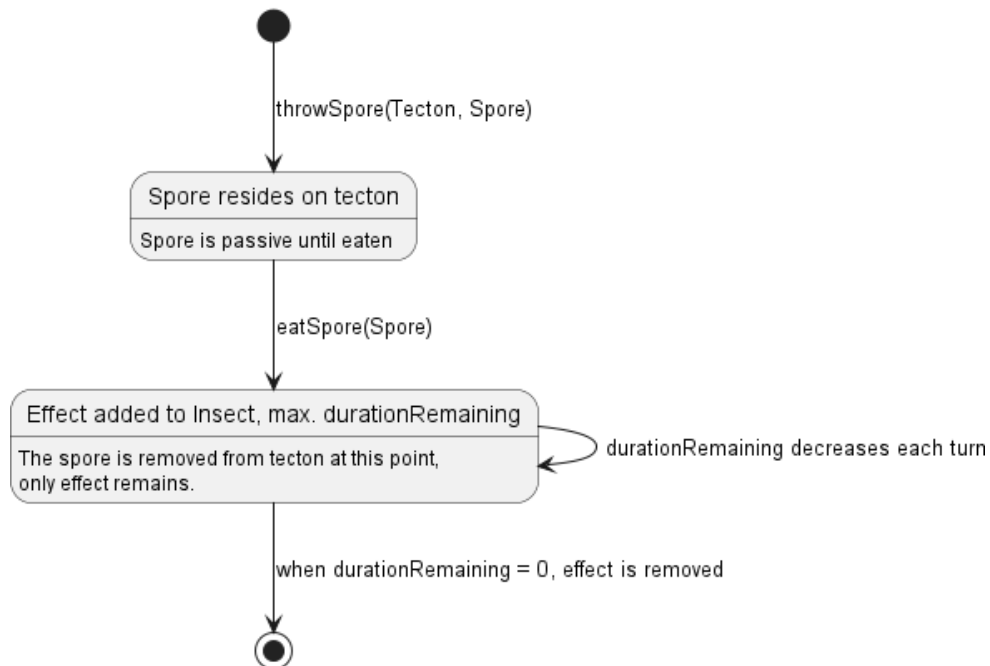




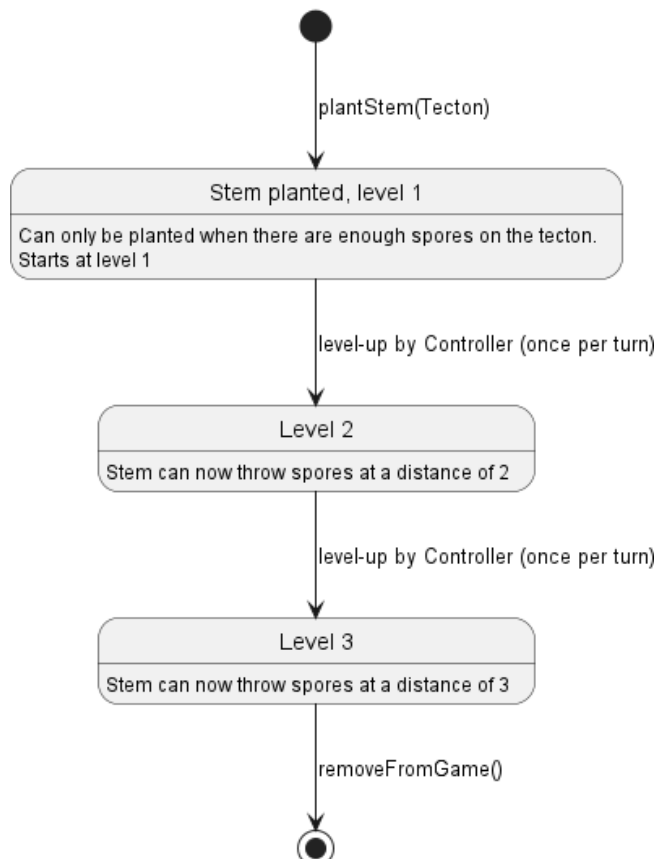


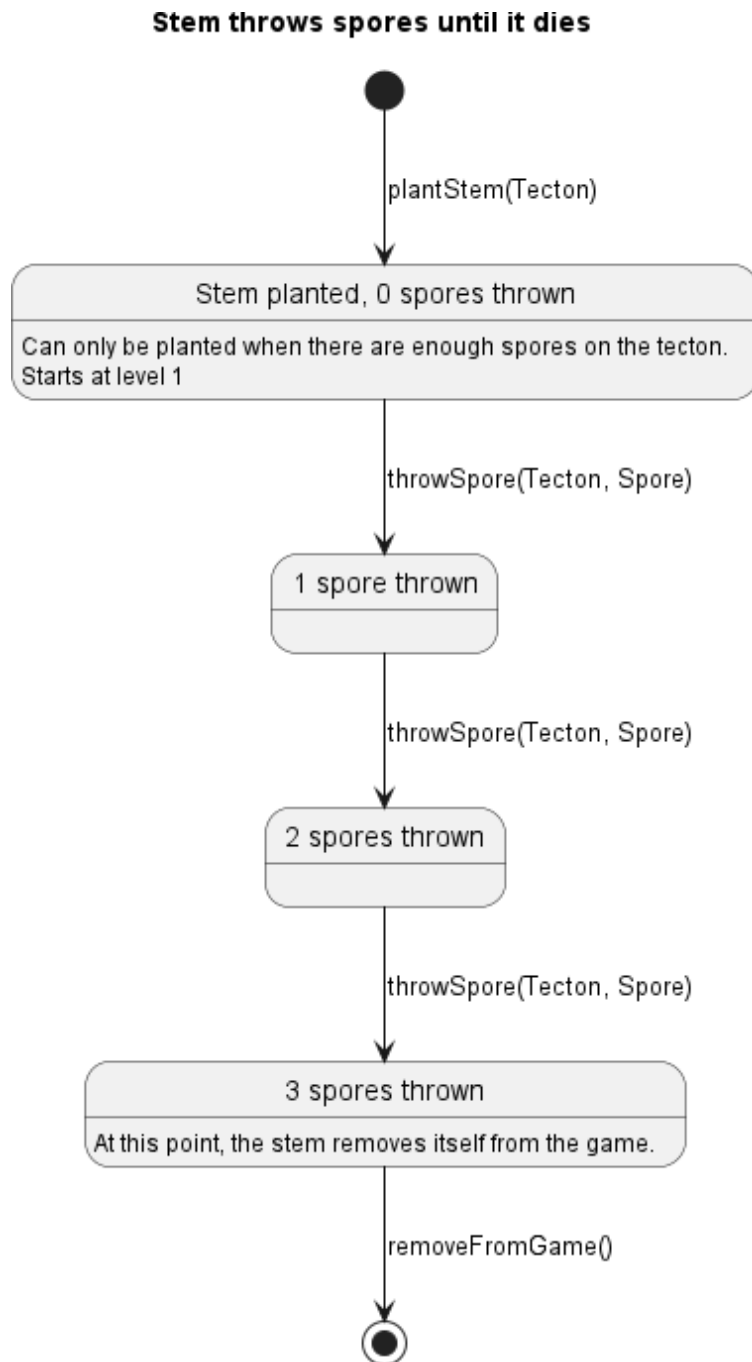
### 3.5 State-chartok

#### Spore is thrown, then eaten



#### Stem is upgraded by Player in their turns





### 3.6 Napló

Kezdet	Időtartam	Résztevők	Leírás
2025.02.24. 14:00	1 óra	Dúcz Galambos Horváth Végvári	Értekezlet. Döntés: Dúcz elkészíti a Játékos osztályt, Galambos elkészíti a Rovar osztályt, Horváth elkészíti a gombatest osztályt, Végvári elkészíti a tekton osztályt, Bohus elkészíti a spóra osztályt.
2025.02.24. 17:00	2 óra	Végvári	Tevékenység: Objektum katalógus: pálya, tekton, diagramok
2025.02.25. 10:00	2 óra	Végvári	Tevékenység: Diagramok készítése
2025.02.25 13:00	2 óra	Horváth	Tevékenység: Gombatest osztály Gombakolónia osztály
2025.02.25 20:00	1 óra	Dúcz	Tevékenység: IPlayer interfész, ShroomPlayer, InsectPlayer
2025.02.26 11:00	1 óra	Dúcz	Tevékenység: Thread osztály
2025.02.26 9:00	1 óra	Bohus	Tevékenység: Spóra osztályok megírása
2025.02.26 12:00	1 óra	Dúcz Bohus Galambos	Értekezlet: Döntés: Osztálydiagram átbeszélése, átírása
2025.02.26 17:00	4 óra	Végvári	Tevékenységek: Osztálydiagramok, osztályok leírása

2025.02.26 19:00	2.5 óra	Dúcz	Tevékenység: Osztálydiagramok, egyéb hibajavítások
2025.02.27 12:00	1 óra	Végvári	Tevékenységek: Osztálydiagramok
2025.02.27 13:00	2,5 óra	Horváth	Tevékenység: Dokumentum frissítése az osztálydiagramok szerint
2025.02.27 17:00	1 óra	Galambos	Tevékenység: Osztálydiagram írása és osztályok írása
2025.02.27 20:00	3 óra	Galambos	Tevékenység: Osztálydiagram, osztályok írása, átnézése
2025.02.28 10:30	2 óra	Galambos	Tevékenység: Osztálydiagram írása, szekvenciadiagram írása
2025.02.28 14:00	2 óra	Végvári Dúcz	Tevékenység: Osztálydiagramok, szekvencia diagramok
2025.02.28 17:00	2 óra	Bohus	Tevékenység: Diagramok másolása, diagramok írása
2025.02.28 16:00	2 óra	Dúcz	Tevékenység: Szekvencia diagramok, osztálydiagram
2025.02.28 20:00	1 óra	Végvári	Tevékenység: Osztálydiagramok, szekvencia diagramok

## 3. Analízis modell kidolgozása

totoro

2025.03.01 11:00	3 óra	Horváth	Tevékenység: Dokumentum írása az osztálydiagramok alapján Állapotdiagram tervezése
------------------	-------	---------	---



			Osztálydiagramok javítása
2025.03.01 11:00	1 óra	Bohus	Tevékenység: diagramok írása, javítása
2025.03.01 11:00	1 óra	Dúcz	Tevékenység: diagramok javítása
2025.03.01 12:00	1.5 óra	Dúcz	Tevékenység: state diagramok

# Tab 1

## 4. Analízis modell (II. változat)

23 – totoro

Konzulens:

Haragos Gergő Viktor

### Csapattagok

**Bohus János**

Dúcz Ákos

Galambos Csaba

Horváth

Avarka

Végvári Péter

**XYOVFZ**

GC1RTE

BVBQKN

OODMXV

AW3JH1

**janos.bohus1@gmail.com**

akos.ducz@gmail.com

galambos.csaba05@gmail.com

avarkahorvath@gmail.com

vegvaripeter21@gmail.co

m

2025.03.05

## 4. Analízis modell kidolgozása

### 4.1 *Objektum katalógus*

#### 4.1.1 Pálya

Itt helyezkedik el a játék összes eleme. Tektonokból áll, a tektonok között pedig víz van. Felelőssége a pályán lévő tektonokat számon tartani.

#### 4.1.2 Tekton

A tektonok a pálya alapelemei, egy tektonon egy gombatest, gombafonalak, illetve rovarok lehetnek. Egy tekton szomszédos lehet más tektonokkal. Nyilvántartja tehát, hogy mik van rajta, és milyen szomszédjai vannak. Minden kör végén ketté válhatnak. Több fajta tekton létezik, különböző tulajdonságokkal.

#### 4.1.3 Fonalelnyelő tekton

Olyan tektonon, ahonnan a fonalak pár kör után felszívódnak.

#### 4.1.4 Gombamentes tekton

Olyan tekton, amin nem tud gombatest nőni.

#### 4.1.5 Egyfonalas tekton

Olyan tekton, amin csak egy fonál lehet.

#### 4.1.6 Gombatest

A gombász egy tektonon tudja elhelyezni, ha van ott fonal és megfelelő mennyiségű spóra található ott. Energiát termel minden turn-re. A termelt energiát spórák kilövésére tudja költeni, szomszédos tektonra és ha elég spóra van egy tektonon akkor ott nő egy új gombatest. Minél nagyobb szintű, annál több energiát termel a gombatest, és annál messzebbre tudja kilögni spóráit. Adott számú spóra kilövés után elpusztul.

#### 4.1.7 Fonál

A tektonokon helyezkedik el, és szomszédos tektonokat köt össze. Minden gombásznak saját fonaltípusa van. Két szomszédos tekton akkor van összekötve egy adott fonaltípus által, ha mindkettőn van adott típusú fonal. Ha egy tektonon van spóra akkor arra gyorsabban nő gombafonal.

#### 4.1.8 Gombakolónia

Egy gombafaj összefüggő gombatestjei és fonalai alkotják.

#### 4.1.9 Gyorsító spóra

Ha rovar megeszi, akkor több gombafonalon tud áthaladni a turn-jében.

#### 4.1.10 Lassító spóra

Ha rovar megeszi, akkor kevesebb gombafonalon tud áthaladni a turn-jében.

#### 4.1.11 Bénító spóra

Ha rovar megeszi, akkor pár round-ig le lesz bénulva.

#### **4.1.12 Csáprágó bénító spóra**

Ha rovar megeszi, akkor pár round-ig le lesz bénulva a csáprágója, nem fog tudni gombafonalakat elvágni.

#### **4.1.13 Rovar**

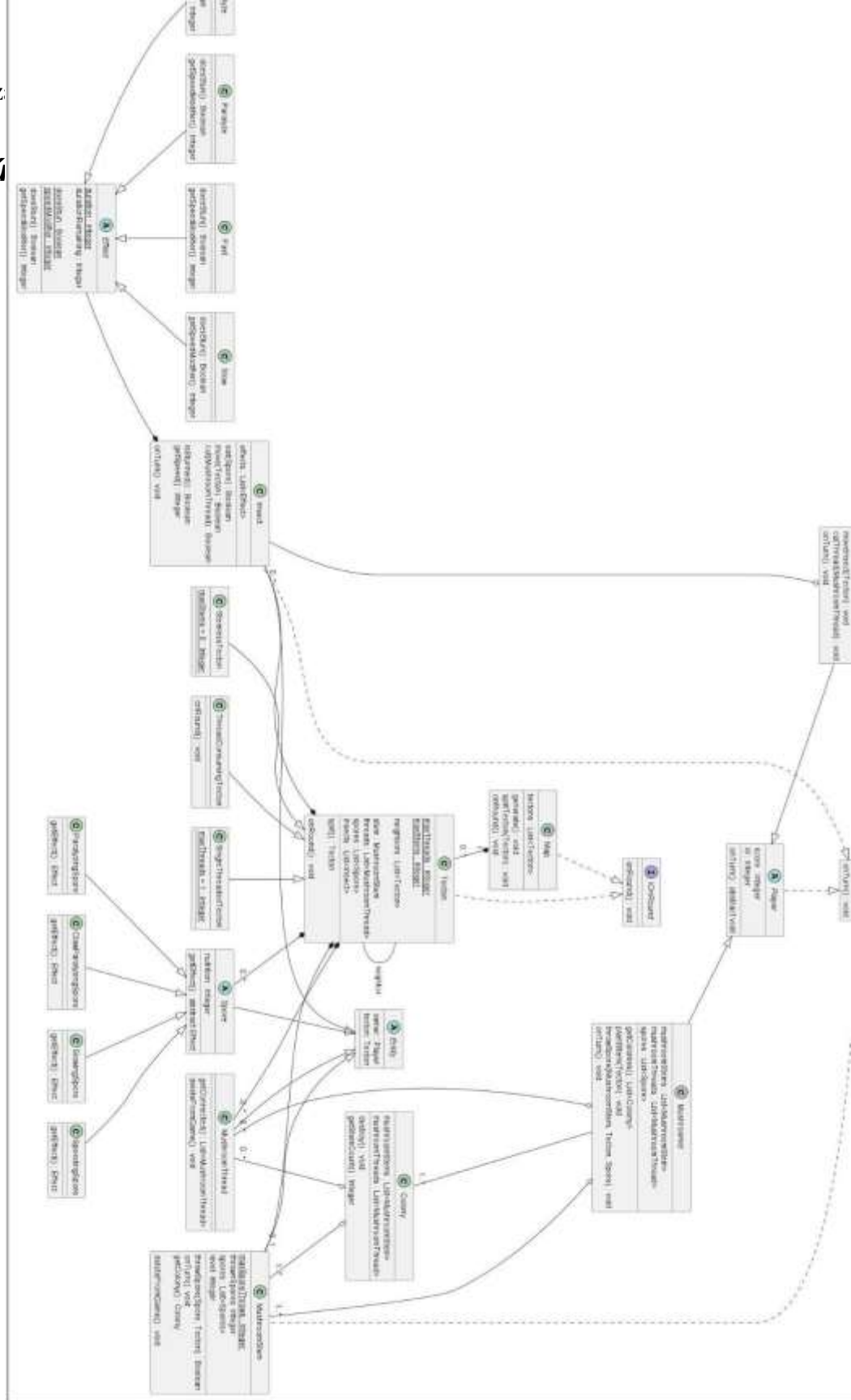
A rovarok a tektonokon vannak, tudnak mozogni, spórát enni, illetve fonalakat elvágni. Tudják melyik tektonon vannak.

#### **4.1.14 Rovarász**

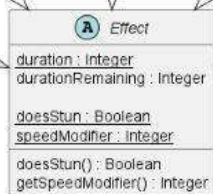
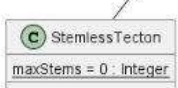
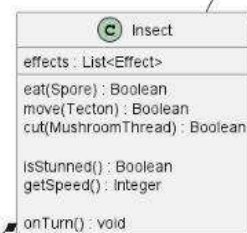
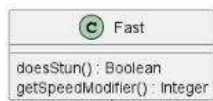
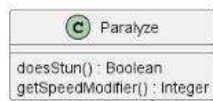
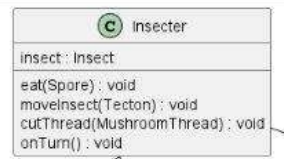
A rovarászok egy rovar irányítanak.

A rovarász játékosok egymással versenyeznek. Az a rovarász nyer, aki a legtöbb tápanyagot fogyasztotta el.

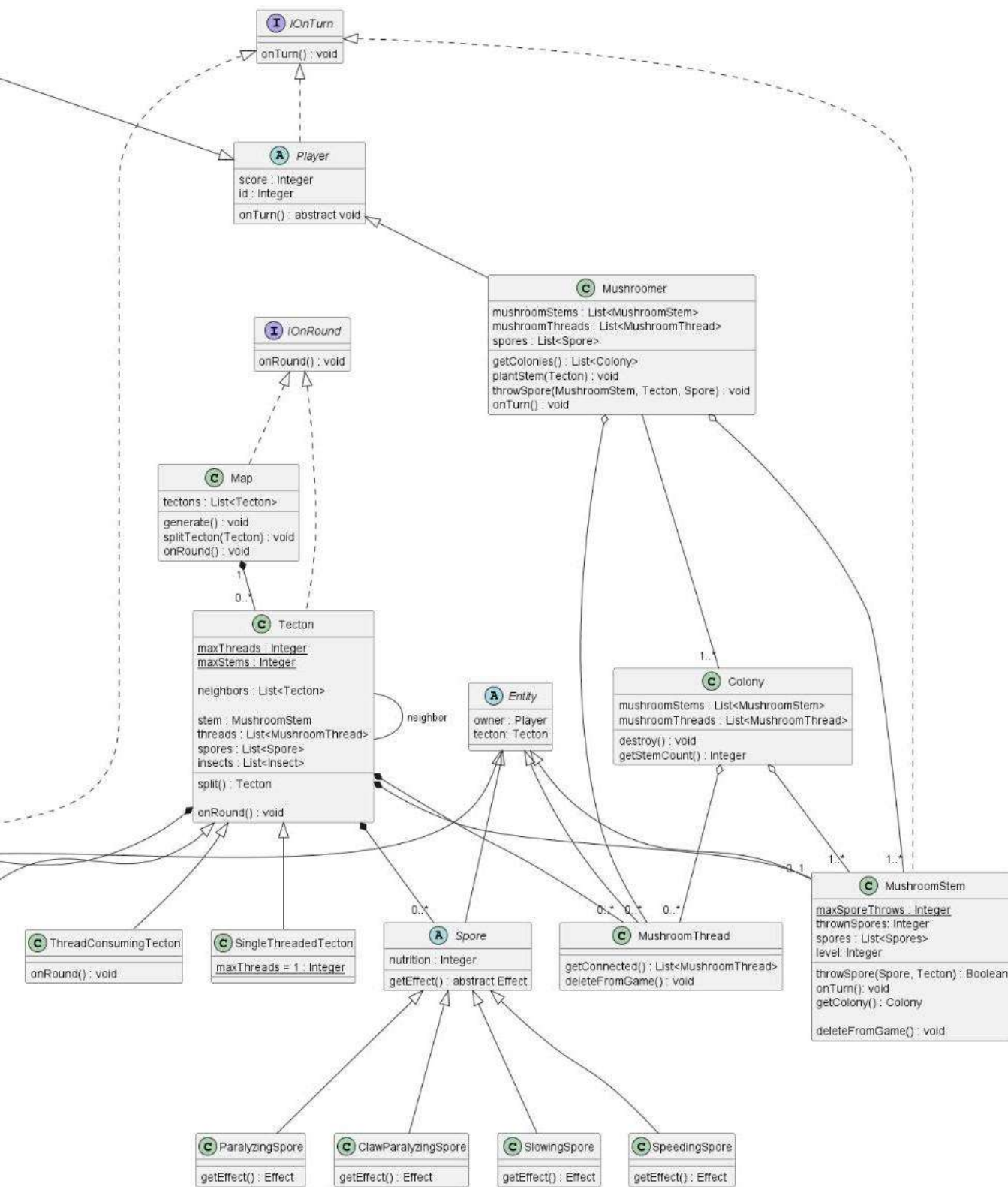
## 4.2 Statikus struktúra



fungorium



0





## 4.3 Osztályok leírása

### 3.1.1 ClawParalyze

- **Felelősség**

Az osztály egy effektus, amely a rovarokra képes hatni. A hatása alatt lévő rovar csáprágója bénított állapotban van, így nem képes gombafonalat elrágni, vagy táplálékot fogyasztani.

- **Ősosztályok**

Effect → ClawParalyze

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **Boolean doesStun():** Megmondja, hogy lefagyasztja-e az effektus a rovar csáprágóját.
- **Integer getSpeedModifier():** Visszaadja, hogy milyen módon manipulálja az effektus a rovar sebességét.

### 3.1.2 ClawParalyzingSpore

- **Felelősség**

Egyfajta spóra, tektonokon van. Ha egy rovar megeszi, akkor pár körig nem tudja használni a csáprágóját, tehát nem tud enni, vagy gombafonalat elrágni.

- **Ősosztályok**

Entity → Spore → ClawParalyzingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **Effect getEffect():** Ha egy rovar megeszi, akkor a csáprágója bénítva lesz pár körig.

### 4.3.1 Colony

- **Felelősség**

A Colony osztály feladata, hogy egy gombakolónia összefüggő részét kezelhetővé tegye.

- **Össztályok**

Az osztálynak nincs őse.

- **Interfészek**

A Colony osztály nem valósít meg interfészt.

- **Asszociációk**

- **Aggregáció1:** Az összefüggő részben lehetnek gombatestek, így MushroomStem a túloldali szereplő.
- **Aggregáció2:** A Gombakolóniába tartoznak a gombafonalak, így MushroomThread a túloldali szereplő.
- **Asszociáció:** Asszociációban van a Mushroomer osztállyal, hiszen egy gombász játékosnak egyszerre legalább 1 kolóniája van.

- **Attribútumok**

- **List<MushroomStem> mushroomStems:** Az összefüggő részben lévő gombatestek.
- **List<MushroomThread> mushroomThreads:** Az összefüggő részben lévő gombafonalak.

- **Metódusok**

- **void destroy():** A kolóniában található összes gombatestet és gombafonalat eltávolítja a térképről.
- **Integer getStemCount():** Megmondja, hogy hány gombatest található a kolóniában.

### 3.1.3 Effect

- **Felelősség**

Absztrakt osztály. Effektust képvisel, azaz olyan hatásokat, amely alatt a rovarok lehetnek valamilyen spóra megevése után.

- **Össztályok**

Nincs őssztálya.

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **static Integer duration:** Az effektus hatásának az ideje. Statikus tagváltozó.
- **Integer durationRemaining:** Az effektus hatásának idejéből hátralévő idő.
- **static Boolean doesStun:** Tárolja, hogy az effektus bénítja-e a csáprágót. Statikus tagváltozó.
- **static Integer speedModifier:** Tárolja, hogy az effektus mennyire változtatja meg a rovar mozgási sebességét. Statikus tagváltozó.

- **Metódusok**

- **Boolean doesStun():** Megmondja, hogy az effektus bénítja-e a csáprágót.
- **Integer getSpeedModifier():** Megmondja, hogy az effektus mennyire változtatja meg a rovar mozgási sebességét.

### 3.1.4 Entity

- **Felelősség**

Absztrakt osztály. Minden olyan objektum ebből származik le, amit birtokol egy játékos.

- **Ősosztályok**

Nem származik le egy osztályból sem.

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Player owner:** Az a játékos, aki birtokolja az objektumot
- **Tecton tecton:** Az entitást tartalmazó tecton.

- **Metódusok**

### 3.1.5 Fast

- **Felelősség**

Az osztály egy effektus, amely a rovarokra képes hatni és a gyorsító hatást reprezentálja.

- **Ősosztályok**

Effect → Fast

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **Boolean doesStun():** Megmondja, hogy az effektus bénítja-e a csáprágót.
- **Integer getSpeedModifier():** Megmondja, hogy az effektus mennyire változtatja meg a rovar mozgási sebességét.

### 3.1.6 Insect

- **Felelősség**

Az Insect osztály egy rovarot reprezentál, melyet egy rovarász irányít. Képes fonalakat elvágni, spórékat megenni és mozogni a tektonok között, a fonalakon. Tudja, melyik tektonon van.

- **Ősosztályok**

Entity → Insect

- **Interfészek**

Az Insect osztály az IOnTurn interfészt valósítja meg.

- **Asszociációk**

- **kompozíció:** A rovarok különböző hatások alatt lehetnek, így a túloldali szereplő az Effect osztály.

- **Attribútumok**

- **List<Effect> effects:** Azokat a hatásokat tárolja, amik alatt a rovar éppen van.

- **Metódusok**

- **void eat(Spore Sp):** Megeszik egy spórát, eltávolítva azt a tektonról.
- **void move(Tecton T):** A rovar átmegy egy másik tektonra.
- **void cut(MushroomThread Mt):** A rovar elvág egy fonalat azon tektonon, amelyen tartózkodik.
- **Boolean isStunned():** Megmondja, hogy a rovar csáprágója működik-e.
- **Integer getSpeed():** Megmondja, hogy milyen sebességgel tud mozogni a rovar.
- **void onTurn():** Arra szolgál, hogy a lépés utáni tevékenységeket végrehajtsa.

### 3.1.7 Insecter

- **Felelősség**

A rovarász típusú játékosokat kezelő osztály.

- **Ősosztályok**

Player □ Insecter

- **Interfészek**

Az Insecter osztály nem valósít meg interfészt.

- **Asszociációk**

- **Aggregáció:** Tartalmazza a játékoshoz tartozó Insect objektumot.

- **Attribútumok**

- **Insect insect:** A játékos által irányított rovar objektum.

- **Metódusok**

- **void eat(Spore sp):** A rovar spórát eszik.
- **void moveInsect(Tecton t):** Egy tektonra mozgatja a rovar.
- **void cutThread(MushroomThread mt):** A rovar elvág egy fonalat.
- **void onTurn():** Arra szolgál, hogy a lépés utáni tevékenységeket végrehajtsa.

### 3.1.8 IOnRound

- **Felelősség**

Interfész, azok implementálják, akik valamit csinálnak minden körben.

- **Ősosztályok**

Nincs.

- **Asszociációk**

- **Metódusok**

- **void onRound():** Az a metódus, amit végrehajt minden körnél.

### 3.1.9 IOnTurn

- **Felelősség**

Interfész, azok implementálják, akik valamit csinálnak minden lépésben.

- **Ősosztályok**

Nincs.

- **Asszociációk**

- **Metódusok**

- **void onTurn():** Az a metódus, amit végrehajt minden lépésnél.

### 3.1.10 Map

- **Felelősség**

A tektonokért felel, lényegében a pálya.

- **Ősosztályok**

Az osztálynak nincs ősosztálya.

- **Interfészek**

Az IOnRound interfészt valósítja meg.

- **Asszociációk**

- **aggregáció1**: Tecton a túloldali szereplő, hiszen a térképen tektonok vannak.

- **Attribútumok**

- **List<Tecton> tectons**: A pálya a tektonjait tárolja.

- **Metódusok**

- **void generate()**: Létrehozza a pályát véletlenszerűen. Elhelyezi a szabályoknak megfelelően a tektonokat.
- **void splitTecton(Tecton t)**: Egy tektont 2 részre bont, azaz szétszakít.
- **onRound()**: Az a metódus, amit végrehajt minden körben.

### 3.1.11 Mushroomer

- **Felelősség**

A gombász típusú játékost kezelő osztály.

- **Ősosztályok**

- Player □ Mushroomer

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

- **aggregáció1:** Tartalmazza a játékoshoz tartozó MushroomStem objektumokat.
- **aggregáció2:** Tartalmazza a játékoshoz tartozó MushroomThread objektumokat.
- **asszociáció:** Több gombakolóniája is lehet (Colony).

- **Attribútumok**

- **List<MushroomStem> mushroomStems:** A játékoshoz tartozó összes gombatest.
- **List<MushroomThread> mushroomThreads:** A játékoshoz tartozó összes gombafonál.

- **Metódusok**

- **List<Colony> getColonies():** BFS algoritmus segítségével megkeresi a fonalak mentén összefüggő gombakolóniákat, melyek a játékoshoz tartoznak, és visszaadja ezeket.
- **void plantStem(Tecton t):** Egy tektonra elhelyez egy gombatestet.
- **void throwSpore(MushroomStem, Tecton, Spore):** Egy tektonra spórát dob.
- **void onTurn():** Az a metódus, amit végrehajt minden lépésnél.

### 3.1.12 MushroomStem

- **Felelősség**

A gombatest osztály felelőssége, hogy a gombatesteket kezelje, és ezek tulajdonságainak nyilvántartása.

- **Ősosztályok**

Entity → MushroomStem

- **Interfészek**

Az IOnTurn interfészt valósítja meg.

- **Asszociációk**

- **Attribútumok**

- **static Integer maxSporeThrows:** Rögzíti, hogy hányszor tud spórát dobni a gombatest. Statikus tagváltozó.
- **Integer thrownSpores:** Nyilvántartja, hogy hányszor dobott eddig spórát.
- **Integer level:** Tárolja, hogy hányas szinten van a gombatest.

- **Metódusok**

- **Boolean throwSpore(Spore s, Tecton t):** Spórát dob.
- **Colony getColony():** Megnézi, hogy melyik kolóniában van a gombatest.
- **void onTurn():** Az a metódus, amit végrehajt minden lépésnél.

### 3.1.13 MushroomThread

- **Felelősség**

A tektonokon elhelyezkedő, és azokat összekötő gombafonalakat megvalósító osztály.

- **Ősosztályok**

Entity □ MushroomThread

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **List<MushroomThread> getConnected():** Visszaadja a tektonok mentén összekapcsolt, azonos játékoshoz tartozó fonalak listáját (BFS algoritmus segítségével).



### 3.1.14 Paralyze

- **Felelősség**

Az osztály egy effektus, amely a rovarokra képes hatni és a bénító hatást reprezentálja.

- **Ősosztályok**

Effect → Paralyze

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **Boolean doesStun():** Megmondja, hogy az effektus bénítja-e a csáprágót.
- **Integer getSpeedModifier():** Megmondja, hogy az effektus mennyire változtatja meg a rovar mozgási sebességét.

### 3.1.15 ParalyzingSpore

- **Felelősség**

Egyfajta spóra, tektonokon van. Ha egy rovar megeszi, akkor pár körig nem tud mozogni.

- **Ősosztályok**

Entity → Spore → ParalyzingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **Effect getEffect():** Ha egy rovar megeszi, akkor bénítva lesz pár körig.

### 3.1.16 Player

- **Felelősség**

Absztrakt játékos osztály, ebből származik le a gombász és rovarász osztály.

- **Ősosztályok**

Nincs őosztálya.

- **Interfészek**

Az IOnTurn interfészt valósítja meg.

- **Asszociációk**

- **Attribútumok**

- **Integer score:** Játékos által eddig elért pontszám.
- **Integer id:** Játékos egyedi azonosítására szolgál.

- **Metódusok**

- **abstract void onTurn():** Az a metódus, amit végrehajt minden lépésnél.

### 3.1.17 SingleThreadedTecton

- **Felelősség**

Az osztály egy tekton fajta, amelyiken csak egyetlen gombafonál helyezkedhet el.

- **Ősosztályok**

Tecton → SingleThreadedTecton

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **static Integer maxThreads=1:** Statikus attribútum, mely rögzíti, hogy csak 1 gombafonál lehet rajta.

- **Metódusok**

### 3.1.18 StemlessTecton

- **Felelősség**

Az osztály egy tekton fajta, amelyiken csak egyetlen gombafonal helyezkedhet el.

- **Össztályok**

Tecton → StemlessTecton

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **static Integer maxStems=0**: Statikus attribútum, mely rögzíti, hogy 0 gombatest lehet rajta.

- **Metódusok**

### 3.1.19 Slow

- **Felelősség**

Az osztály egy effektus, amely a rovarokra képes hatni és a lassító hatást reprezentálja.

- **Össztályok**

Effect → Slow

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

- **Boolean doesStun()**: Megmondja, hogy az effektus bénítja-e a csáprágót.
- **Integer getSpeedModifier()**: Megmondja, hogy az effektus mennyire változtatja meg a rovar mozgási sebességét.

### 3.1.20 SlowingSpore

- **Felelősség**

Egyfajta spóra, tektonokon van. Ha egy rovar megeszi, akkor pár körig lassabban tud mozogni.

- **Ősosztályok**

Entity → Spore → SlowingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **Effect getEffect():** Ha egy rovar megeszi, akkor lassítva lesz pár körig.

### 3.1.21 SpeedingSpore

- **Felelősség**

Egyfajta spóra, tektonokon van. Ha egy rovar megeszi, akkor pár körig gyorsabban tud mozogni.

- **Ősosztályok**

Entity → Spore → SpeedingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **Effect getEffect():** Ha egy rovar megeszi, akkor gyorsítva lesz pár körig.

### 3.1.22 Spore

- **Felelősség**

Absztrakt osztály, a többi spóra osztály őse, egy játékban lévő spórát reprezentál. Tektonokon van, ha a tekton szétesik, eltűnik akkor a spórák is meghalnak.

- **Ősosztályok**

Entity → Spore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Integer nutrition:** Az az érték, amennyi pontot ad a rovarnak ha megeszi.

- **Metódusok**

- **abstract Effect getEffect():** Visszaadja, hogy ha egy rovar megeszi, akkor milyen hatással lesz rá.

### 3.1.23 Tecton

- **Felelősség**

A pálya tektonokból áll, ezeken helyezkedik el sok másik objektum, amiket a tektonnak kell számon tartania.

- **Interfészek**

Megvalósítja az IOnRound interfészt.

- **Asszociációk**

- **Kompozíció1:** A tekton tárolja a rajta lévő MushroomStem-et.
- **Kompozíció2:** A tekton tárolja a rajta lévő MushroomThread-eket.
- **Kompozíció3:** A tekton tárolja a rajta lévő Spore-okat.
- **Kompozíció4:** A tekton tárolja a rajta lévő Insect-eket.
- **Asszociáció:** A tekton tárolja a saját magával szomszédos tektonokat is.

- **Attribútumok**

- **MushroomStem stem:** A tektonon lévő gombatest.
- **List<MushroomThread> threads:** A tektonon lévő gombafonalakat tárolja.
- **List<Spore> spores:** A tektonon lévő spórákat tárolja.
- **List<Insect> insects:** A tektonon lévő rovarokat tárolja.
- **List<Tecton> neighbors:** a tekton szomszédjai.
- **Static Integer MaxThreads:** A tektonon lévő gombafonalak maximális számát adja meg, a legtöbb tektonon végtelen, de van olyan amin 1.
- **Static Integer MaxStems:** A tektonon lévő gombafonalak maximális számát adja meg, a legtöbb tektonon 1, de van olyan amin 0.

- **Metódusok**

- **Tecton split():** A tekton két darabra törik.
- **void onRound():** Az a metódus, ami minden körben lefut.

### 3.1.24 ThreadConsumingTecton

- **Felelősség**

Az osztály egy tekton fajta, amelyiken nem tud gombafonal megmaradni.

- **Ősosztályok**

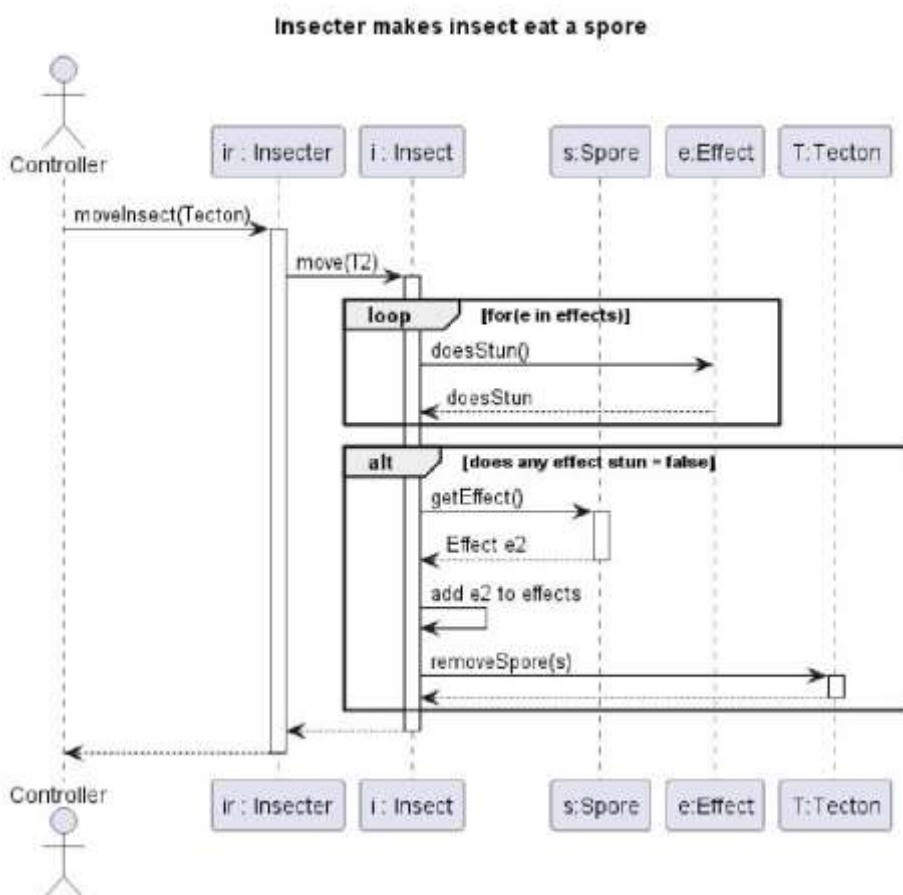
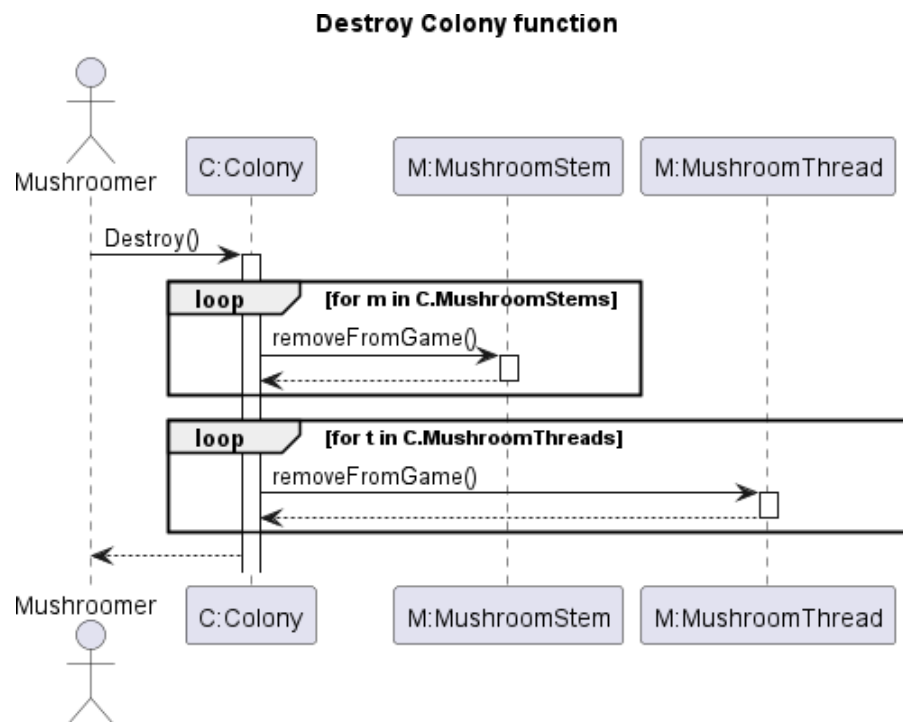
Tecton → ThreadConsumingTecton

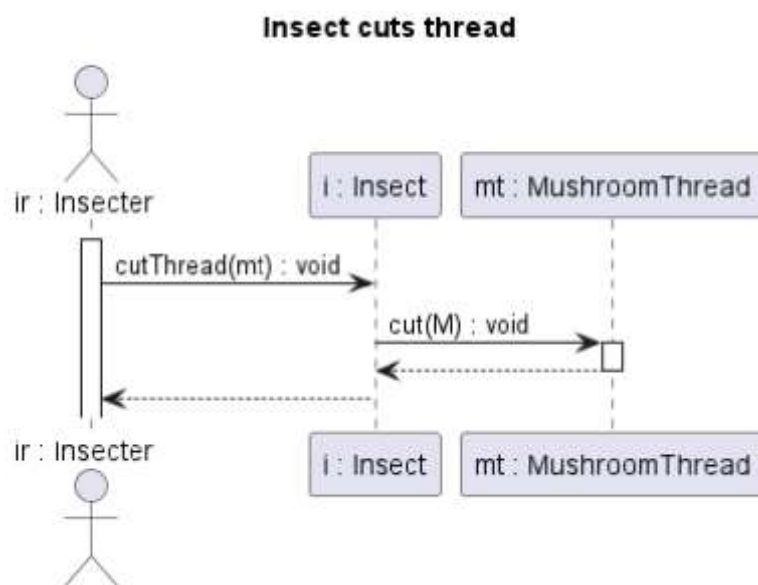
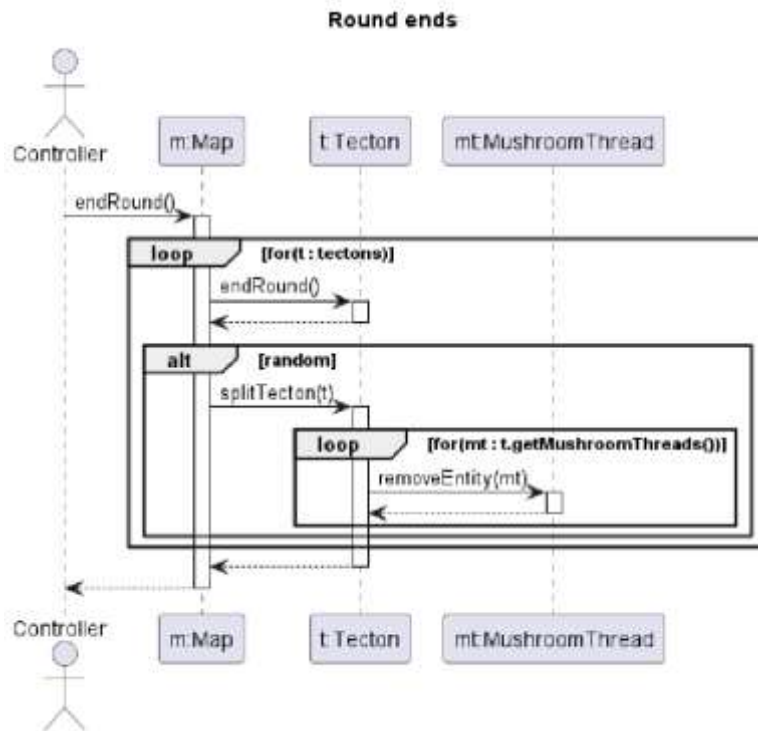
- **Interfészek**

Az osztály nem valósít meg interfészt.

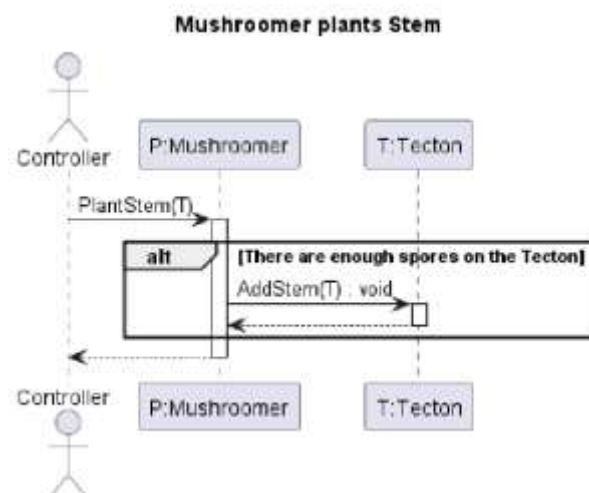
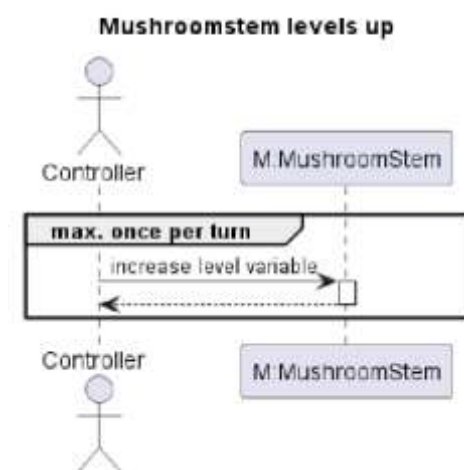
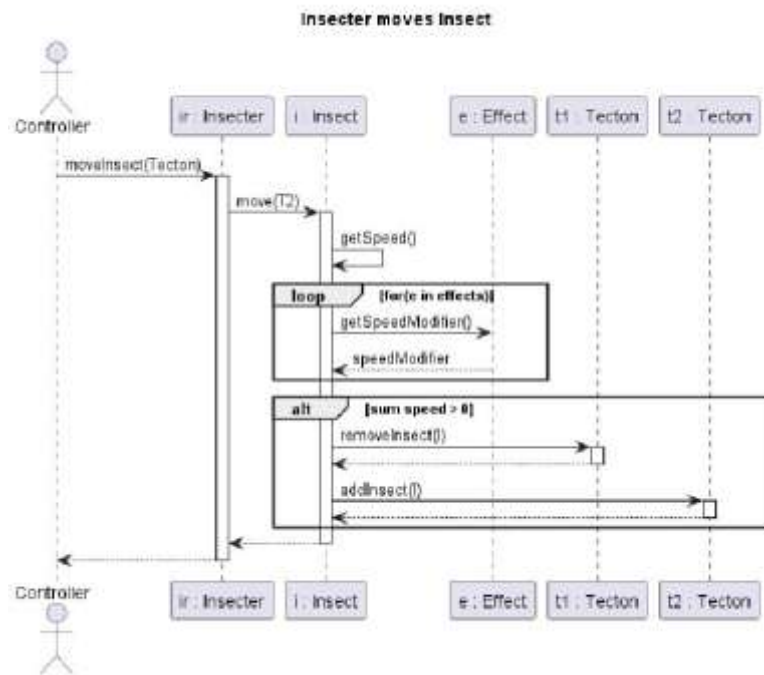
- **Asszociációk**
- **Attribútumok**
- **Metódusok**

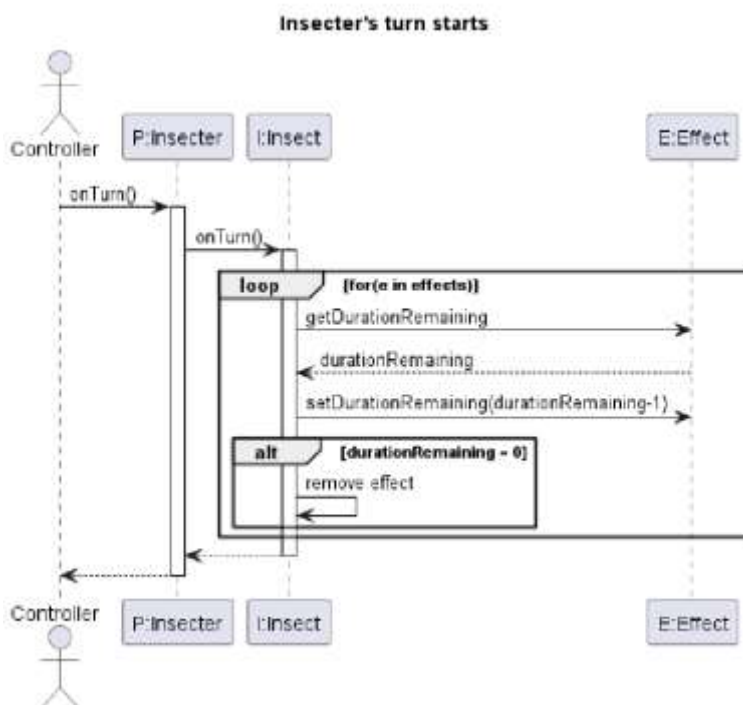
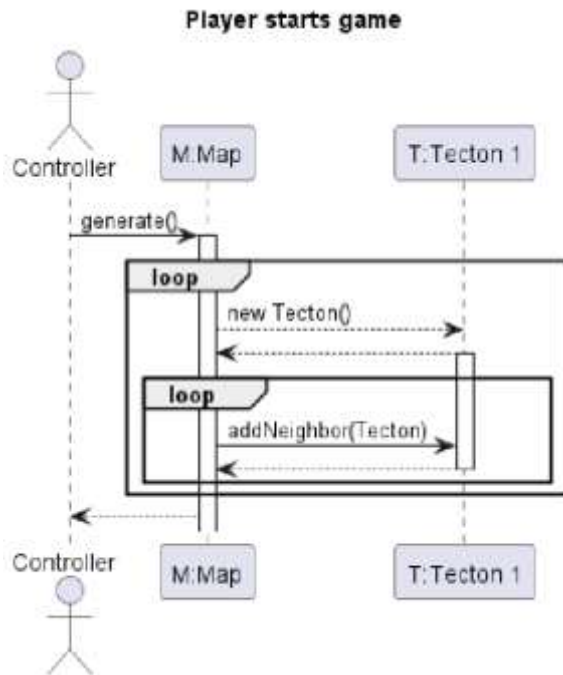
## 4.4 Szekvencia diagramok

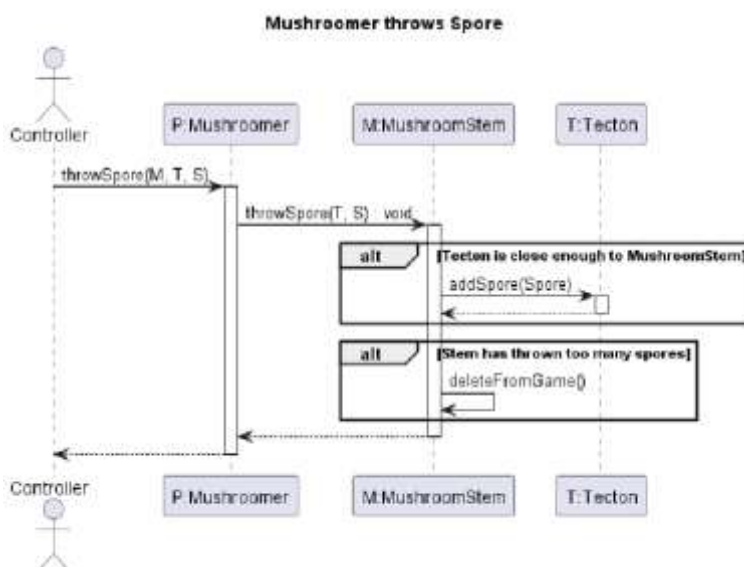
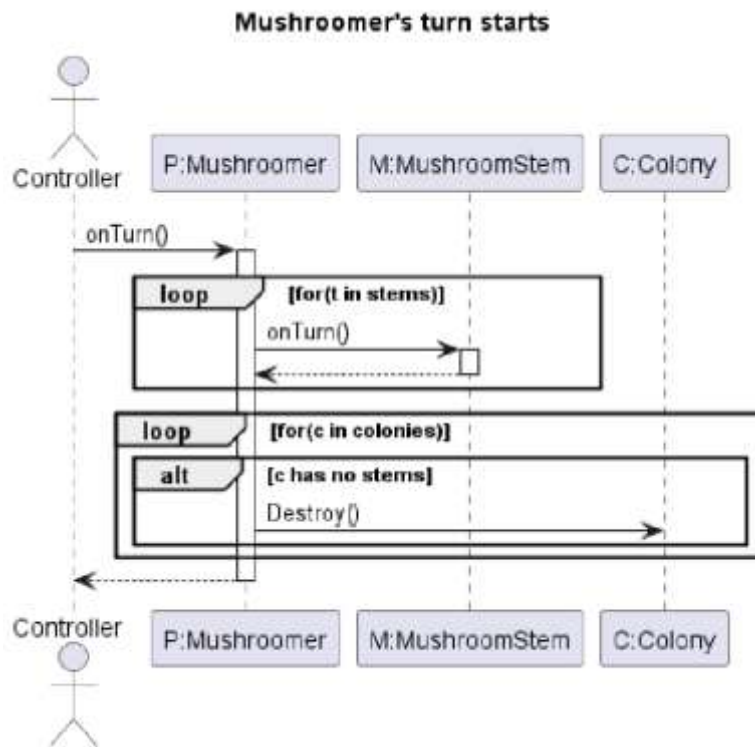






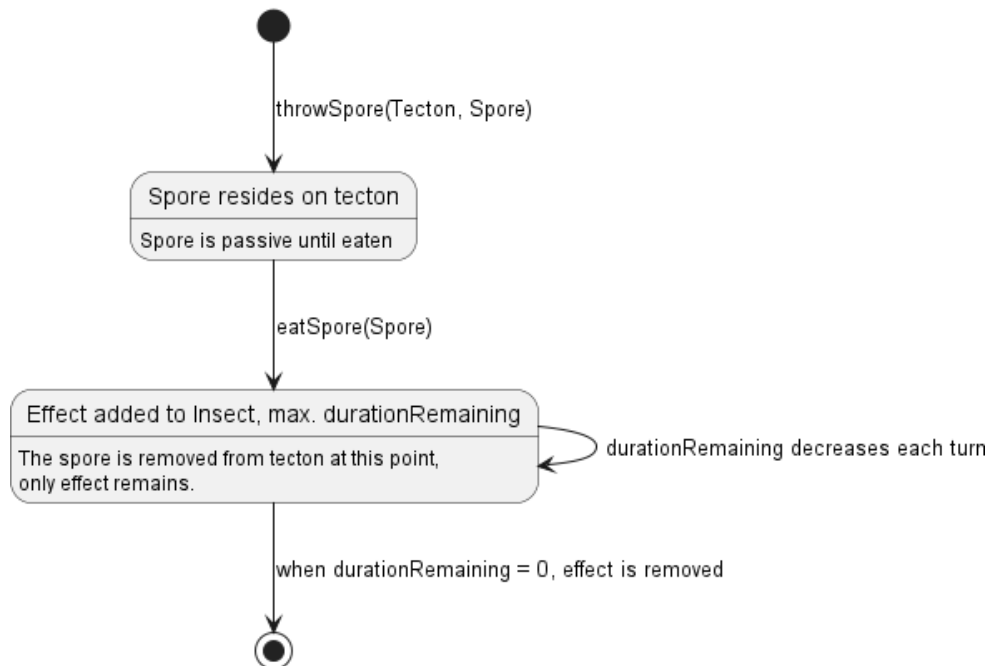




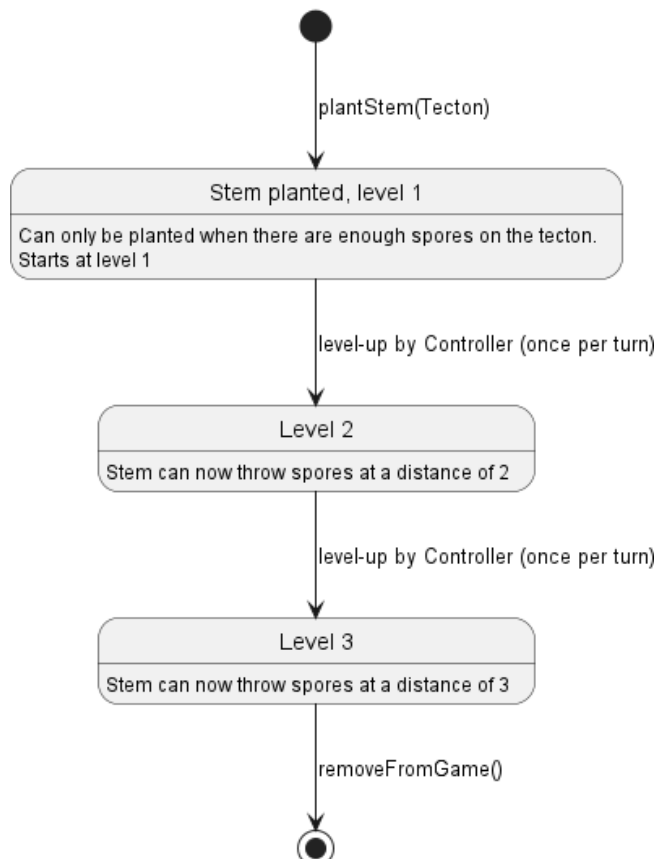


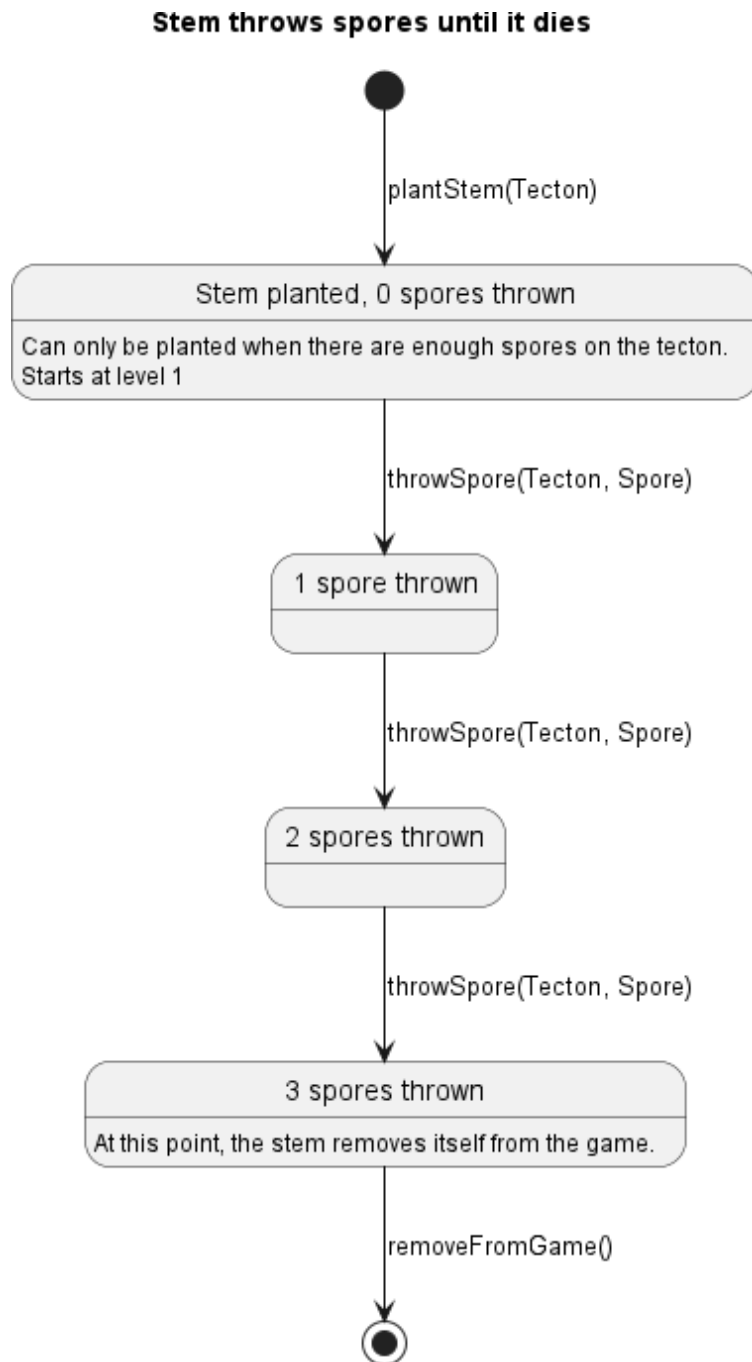
## 4.5 State-chartok

### Spore is thrown, then eaten



### Stem is upgraded by Player in their turns





## 4.6 Napló

Kezdet	Időtartam	Résztevők	Leírás
2025.03.06	1 óra	Dúcz Galambos Horváth Végvári Bohus	Értekezlet. Döntés: Az osztálydiagramon nem változtatunk sokat, a felmerülő hibákat Horváth javítja ki.
2025.03.06 17:00	2 óra	Végvári	Tevékenység: Objektum katalógus frissítése.
2025.03.07 17:00	2 óra	Dúcz	Tevékenység: Osztálydiagram átnézése.
2025.03.07 20:00	2 óra	Horváth	Tevékenység: Szekvenciadiagramok átnézése.
2025.03.07 20:00	1 óra	Galambos	Tevékenység: Állapotdiagramok átnézése.
2025.03.08 16:00	1 óra	Galambos	Tevékenység: Hibák javítása a dokumentumban
2025.03.09 17:00	1 óra	Horváth	Tevékenység: Dokumentum leírások átnézése, javítás

## 5. Szkeleton tervezése

23 – totoro

Konzulens:

Haragos Gergő Viktor

### Csapattagok

<b>Bohus János</b>	<b>XYOVFZ</b>	<b>janos.bohus1@gmail.com</b>	
Dúcz Ákos	GC1RTE	akos.ducz@gmail.com	
Galambos Csaba	BVBQKN	galambos.csaba05@gmail.com	
Horváth Avarka Végvári	OODMXV AW3JH1		a
Péter			v
			a
			r
			k
			a
			h

o  
r  
v  
a  
t  
h  
@  
g  
m  
a  
i  
l  
.  
c

o  
m  
v  
e  
g  
v  
a  
r  
i  
p  
e  
t  
e  
r

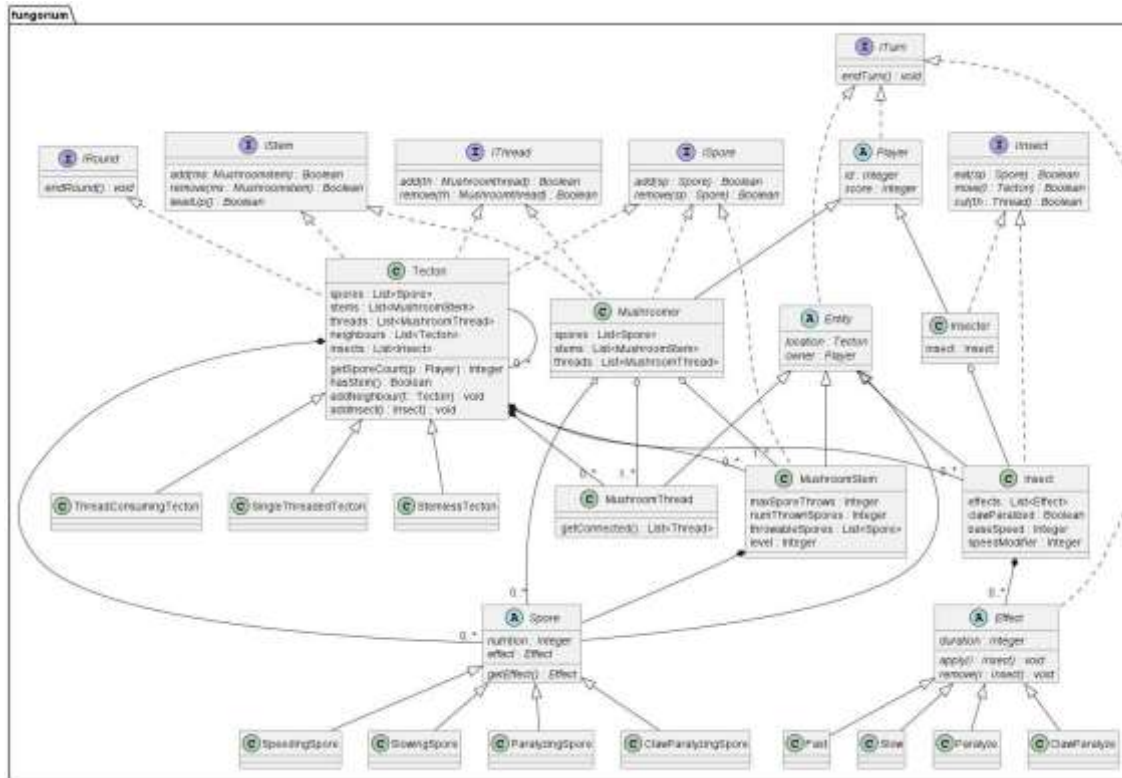
2  
1  
@  
g  
m  
a  
i  
l  
.  
c  
o  
m

2025.03.10



## 5. Szkeleton tervezése

### Módosított modell:



### Változtatások:

- ITurn és IRound interfészek endTurn(), endRound() metódusokkal
- Tacton interfészekkel (ISpore, IStem, IThread) és metódusokkal
- Effect absztrakt osztály apply() és remove() metódusokkal
- Insect-ben paralyzed, clawParalyzed állapotok és baseSpeed, speedModifier
- MushroomStem-hez throwableSpores lista és levelUp() metódus
- Colony kivétele

## **5.1 A szkeleton modell valóságos use-case-ei**

### Spóra

Gombász gyorsító spórát dob tektonra

Gombász lassító spórát dob tektonra

Gombász bénító spórát dob tektonra

Gombász csáprágó bénító spórát dob tektonra

### Gombatest

Gombász gombatestet növeszt spórából tektonra

Gombász gombatest szintjét növeli

### Gombafonal

Gombász gombafonolatot növeszt tektonra

Rovarász rovára egy gombász gombafonalát elvágja

### Rovar

Rovarász rovára tektonra mozog

Rovarász rovára gyorsító spórát eszik

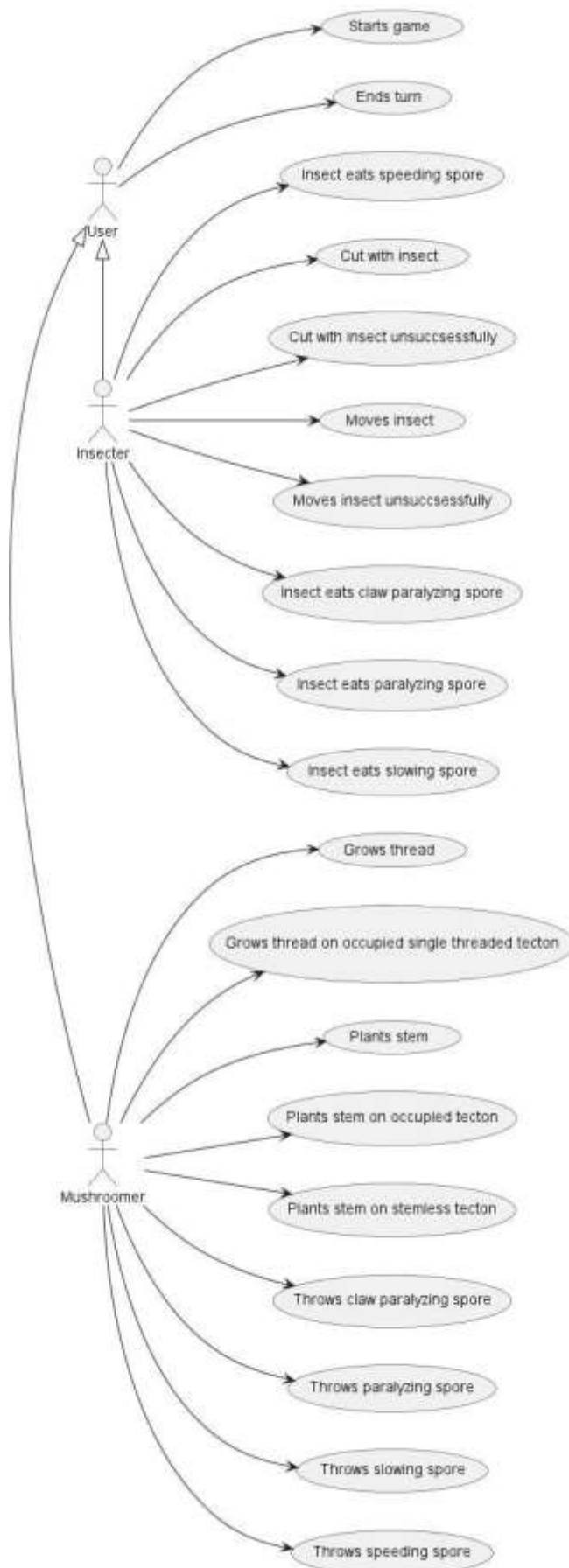
Rovarász rovára lassító spórát eszik

Rovarász rovára bénító spórát eszik

Rovarász rovára csáprágót bénító spórát eszik

Rovarász rovára gombafonolatot vág el

### 5.1.1 Use-case diagram



### 5.1.2 Use-case leírások

<b>Use-case neve</b>	Start Game
<b>Rövid leírás</b>	Játékos elkezd a kört
<b>Aktorok</b>	User
<b>Forgatókönyv</b>	Játékos elkezd a kört, a pálya inicializálódik

<b>Use-case neve</b>	End turn
<b>Rövid leírás</b>	Játékos befejezi a körét
<b>Aktorok</b>	User
<b>Forgatókönyv</b>	Játékos befejezi a körét, továbbadja a következő játékosnak

<b>Use-case neve</b>	Insect eats claw paralyzing spore
<b>Rövid leírás</b>	Rovar spórát eszik
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar megeszi a spórát, ennek következtében nem fog tudni elválni fonalakat

<b>Use-case neve</b>	Moves insect
<b>Rövid leírás</b>	Rovar mozgatása
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovarász mozgatja a rovar egy elérhető távolságban lévő tektonra

<b>Use-case neve</b>	Moves insect unsuccessfully
<b>Rövid leírás</b>	Rovar sikertelen mozgatása
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovarász egy olyan tektont választ ki ami túl messze van, ennek következtében nem fog mozogni

<b>Use-case neve</b>	Insect eats paralyzing spore
<b>Rövid leírás</b>	Rovar kábító spórát eszik
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar kábító spórát eszik, ennek következtében nem fog tudni pár körig mozogni, enni

<b>Use-case neve</b>	Insect eats slowing spore
<b>Rövid leírás</b>	Rovar lassító spórát eszik
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar lassító spórát eszik, ennek következtében következő körökben kevesebbet tud mozogni

<b>Use-case neve</b>	Insect eats speeding spore
<b>Rövid leírás</b>	Rovar gyorsító spórát eszik
<b>Aktorok</b>	Insecter

<b>Forgatókönyv</b>	Rovar gyorsító spórát eszik, ennek következtében következő pár körben többet tud mozogni
---------------------	--

<b>Use-case neve</b>	Cut with insect
<b>Rövid leírás</b>	Rovar elvágja a gombafonalat
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar elvágja a fonalat és az eltűnik a pályáról.

<b>Use-case neve</b>	Cut with insect but can't because stun
<b>Rövid leírás</b>	Rovarnak nem sikerült a vágás
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar megpróbálja elvágni a fonalat, de nem sikerül neki, mert kábító hatás van rajta

<b>Use-case neve</b>	Cut with insect but can't because claw stun
<b>Rövid leírás</b>	Rovarnak nem sikerült a vágás
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar megpróbálja elvágni a fonalat, de nem sikerül neki, mert a vágóján van kábító hatás

<b>Use-case neve</b>	Plants stem
<b>Rövid leírás</b>	Gomba nő
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Ha elég spóra van egy tektonon akkor tud rajta gombatest nőni

<b>Use-case neve</b>	Plants stem on stemless tecton
<b>Rövid leírás</b>	Gombát próbálunk növesztetni, de nem sikerül
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Gombát próbálunk növesztetni egy olyan tektonra amire nem lehet, ezért nem sikerül

<b>Use-case neve</b>	Plants stem on occupied tecton
<b>Rövid leírás</b>	Gombát próbálunk növesztetni, de nem sikerül
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Gombát próbálunk növesztetni egy olyan tektonra amin már van másik gomba, ezért nem sikerül

<b>Use-case neve</b>	Throws claw paralyzing spore
<b>Rövid leírás</b>	Spóra dobása
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Csáprágó kábító spóra dobása, kiválasztunk egy tektont és rá dobjuk a spórát

<b>Use-case neve</b>	Throws paralyzing spore
<b>Rövid leírás</b>	Spóra dobása
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Kábító spóra dobása, kiválasztunk egy tektont és rá dobjuk a spórát

<b>Use-case neve</b>	Throws slowing spore
<b>Rövid leírás</b>	Spóra dobása

<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Lassító spóra dobása, kiválasztunk egy tektont és rá dobjuk a spórát

<b>Use-case neve</b>	Throws speeding spore
<b>Rövid leírás</b>	Spóra dobása
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Gyorsító spóra dobása, kiválasztunk egy tektont és rá dobjuk a spórát

<b>Use-case neve</b>	Throws claw paralyzing spore but can't
<b>Rövid leírás</b>	Spóra dobása olyan tektonra ami nem lehetséges
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Csáprágó kábító spóra dobása, kiválasztunk egy olyan tektont ami túl távol van és nem érjük el, majd megpróbáljuk rádobni a spórát, de nem sikerült neki

<b>Use-case neve</b>	Throws paralyzing spore but can't
<b>Rövid leírás</b>	Spóra dobása olyan tektonra ami nem lehetséges
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Kábító spóra dobása, kiválasztunk egy olyan tektont ami túl távol van és nem érjük el, majd megpróbáljuk rádobni a spórát, de nem sikerült neki

<b>Use-case neve</b>	Throws slowing spore but can't
<b>Rövid leírás</b>	Spóra dobása olyan tektonra ami nem lehetséges
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Lassító spóra dobása, kiválasztunk egy olyan tektont ami túl távol van és nem érjük el, majd megpróbáljuk rádobni a spórát, de nem sikerült neki

<b>Use-case neve</b>	Throws speeding spore but can't
<b>Rövid leírás</b>	Spóra dobása olyan tektonra ami nem lehetséges
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Gyorsító spóra dobása, kiválasztunk egy olyan tektont ami túl távol van és nem érjük el, majd megpróbáljuk rádobni a spórát, de nem sikerült neki

<b>Use-case neve</b>	Grows thread
<b>Rövid leírás</b>	Fonal növesztése
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Kiválasztjuk, hogy milyen irányba akarunk gombafonalat növesztetni

<b>Use-case neve</b>	Grows thread on occupied single threaded tecton
<b>Rövid leírás</b>	Fonal növesztése olyan tektonra, amire nem már lehet



<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Fonal növesztése olyan tektonra, amire nem már lehet

## **5.2 A szkeleton kezelői felületének terve, dialógusok**

A menü úgy működik, hogy mindig kiírja a műveleteket, és mellé egy azonosítót, amit inputként vár. pl Gomba kiválasztása : 1 vagy Kör vége : 2, így lehet választani mit szeretnénk csinálni. Más gomb lenyomására hibaüzenet érkezik. Ha jó inputot adunk, meg akkor kiíródik, hogy mi történt. pl.: Lassító spóra dobása a 10. tektonra. Ez alapvetően minden műveletre igaz.

Játék elején, van lehetőségünk új játékot kezdeni, ha erre nyomunk akkor utána egyesével kéri a konzol a játékosok nevét.

Minden kör elején kiírja, hogy melyik játékos következik.

Gombász köre:

Konzol megkérdezi, hogy el akarja-e kezdeni a kört, azonosító megadásával elkezdődik.

Kör elején automatikusan ellenőrzük, hogy van-e elég spóra az egyik tektonon, ahhoz, hogy gombatest nőjön rajta. Ha van akkor konzolon kiíródik, hogy gomba nőtt, és, hogy melyik tektonra.

Konzol kiír egy menüt ami a lehetséges műveleteket tartalmazza. Mindig csak azok vannak amiket el tudunk végezni. A menü pontjai: Gomba kiválasztása és kör vége.

Ha a gomba kiválasztására nyomunk, akkor felugranak a lehetséges gombák, ezek közül választunk egyet.

Ezután kiválaszthatjuk, hogy spórát akarunk dobni vagy gombafonalat növesztetni.

Spóra dobásánál kiírja a lehetséges spóra fajtákat, és lehetséges tektonokat.

Gombafonal növesztésénél pedig a lehetséges tektonokat, hogy milyen irányban fog növekedni akarjuk, hogy terjeszkedjen a fonal.

Ha választottunk mindig visszadob a kezdeti menüre(Gomba kiválasztása, kör vége) és addig ismétljük ameddig köv végét nem választjuk, ha arra nyomunk, akkor lezajlik pár művelet:

Ha elég spórát dobott a gomba akkor szintet lép. Ha ez bekövetkezik, akkor kiírjuk a konzolra.

Kiírja, hogy merre nőttek fonalak. Ezután, a következő játékos következik.

Rovarász köre következik:

Konzol megkérdezi, hogy el akarja-e kezdeni a kört, azonosító megadásával elkezdődik.

Konzol kiír egy menüt, hogy mit tud tenni a játékos. Ez csak a lehetséges dolgokat írja ki, változik. Ezek lehetnek a mozgás, spóra evés, gombafonal evés, és kör vége opció. Ezen kívül még azt is kiírja hányat tud még mozogni, és milyen effect van érvényben a rovaron.

Rovarász kiválasztja, hogy melyiket szeretné csinálni:

Ha spóra evést választja akkor, ha több spóra van kiírja, hogy melyiket tudjuk megenni.

Kiválasztjuk, majd a konzol visszajelzi, hogy milyen hatással volt ránk, újra megjelenik a menü, már frissítve az effectel. Ha csak egy spóra van rajta akkor automatikusan azt eszi meg.

Ha a gombafonal evést választja akkor elvágja a spórát.

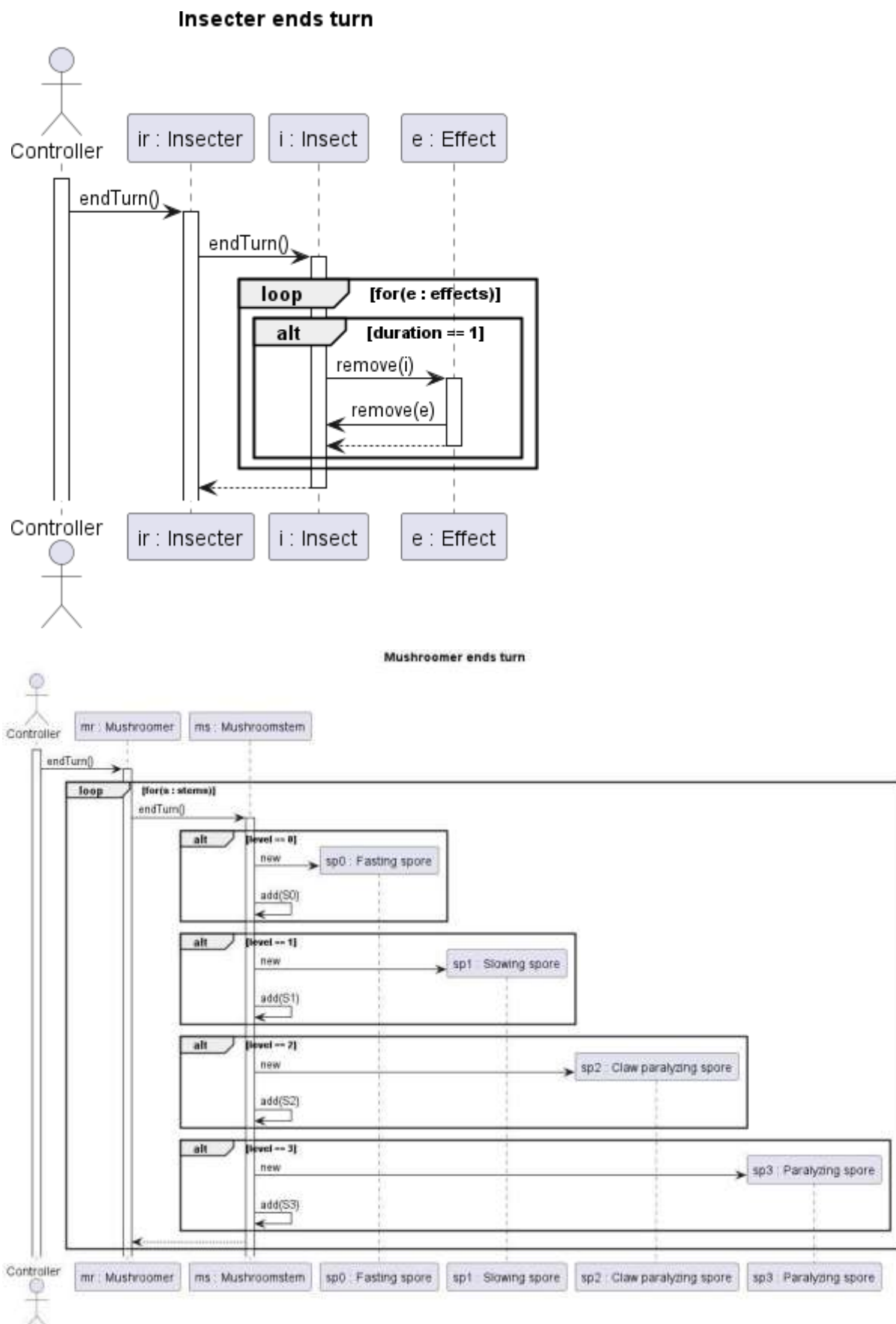
Ha mozgást választjuk akkor kiírja, a szomszédos tektonokat, majd választhatunk, hogy hova szeretnék mozogni.

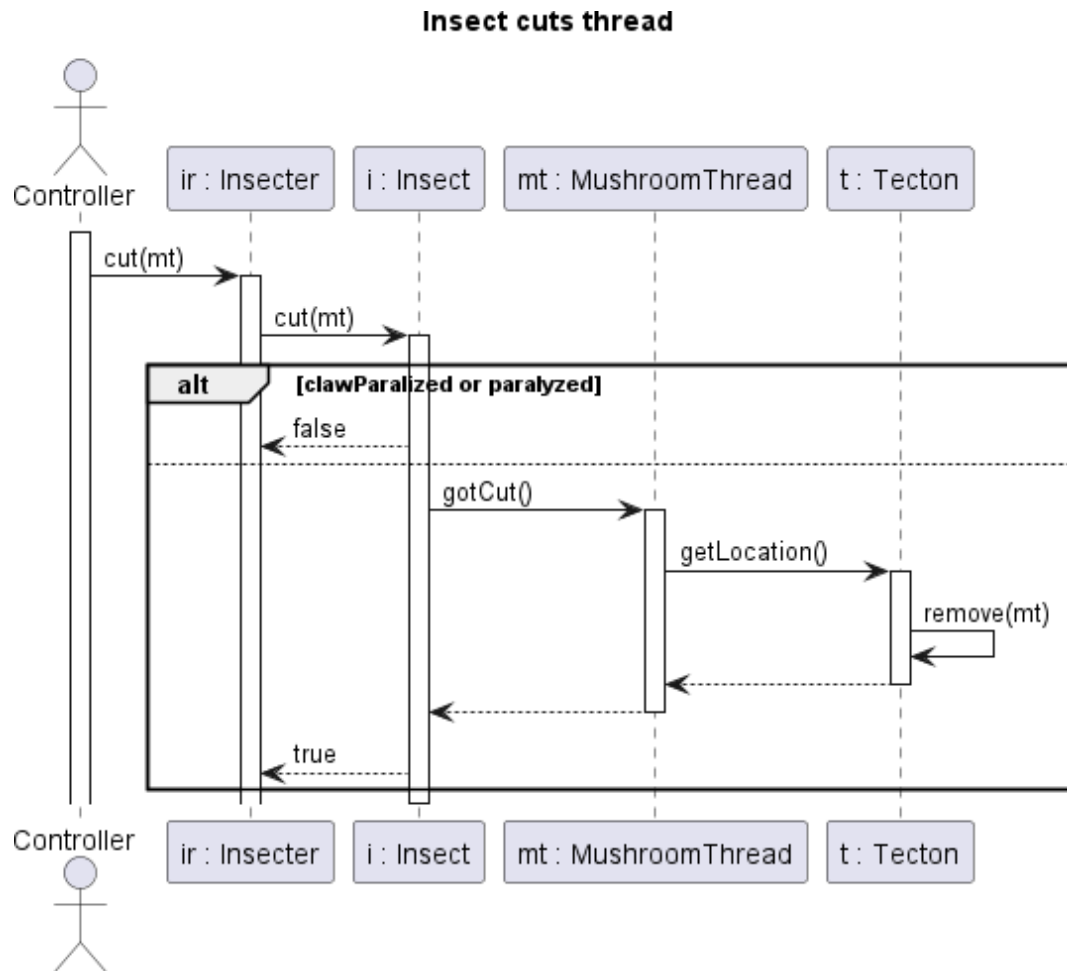
Minden művelet után újra megjelenik a menü, ez addig ismétlődik ameddig el nem fogynak a lépéseink, vagy tovább nem adjuk a kört.

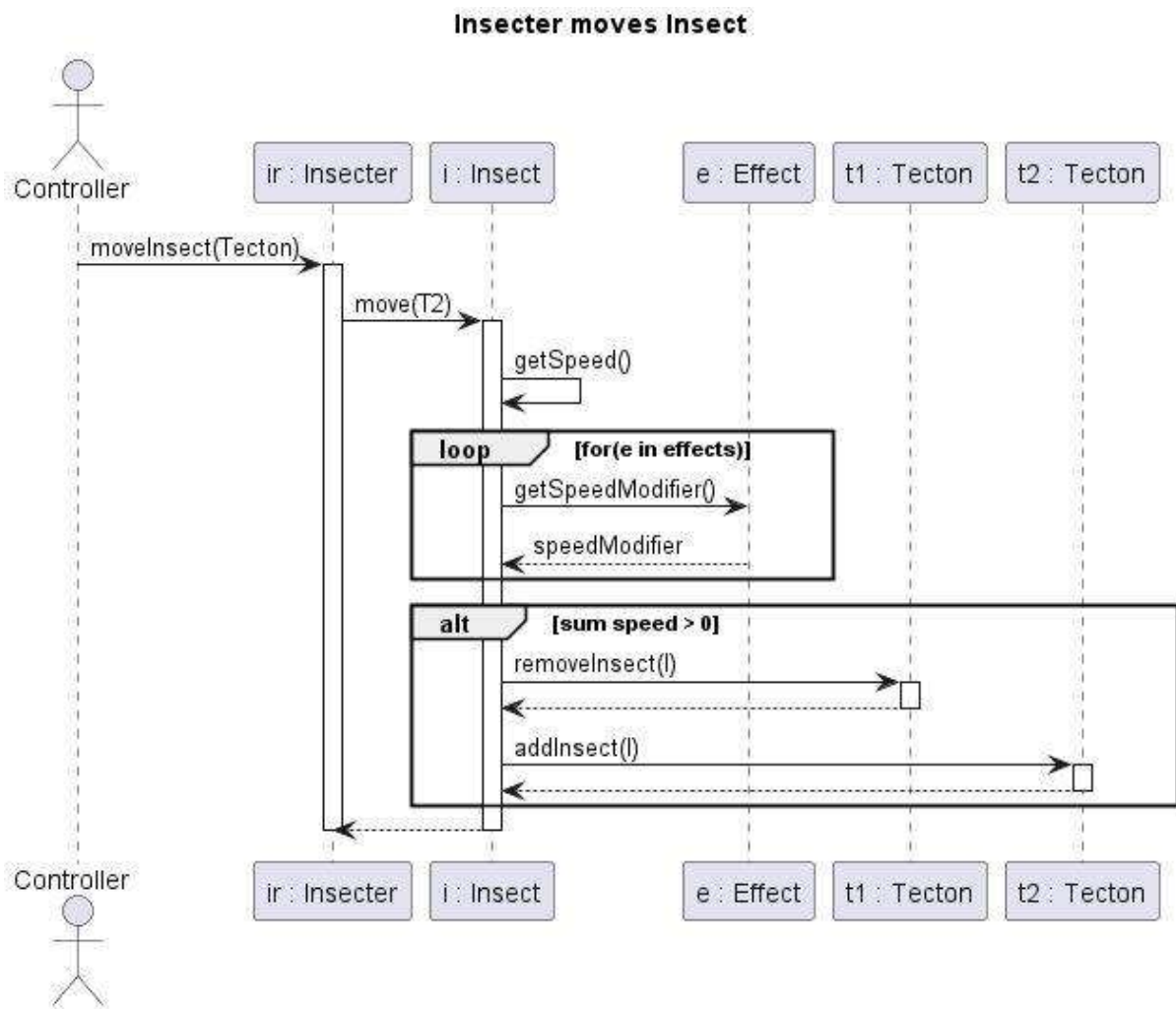
Ha a kör vége azonosítóját írjuk be akkor a továbbadjuk a kört, a következő játékos következik.

Játék végén kiírja a pontokat, sorrendbe, hogy ki hogyan végzett.

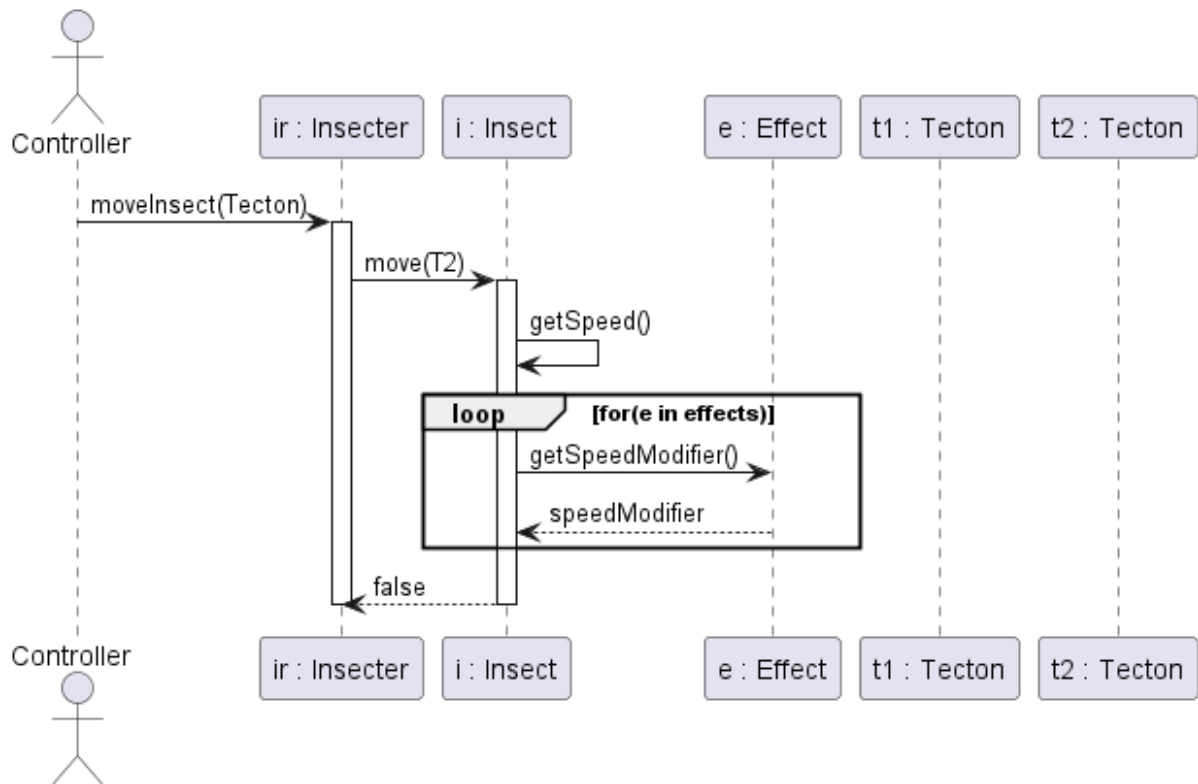
### 5.3 Szekvencia diagramok a belső működésre



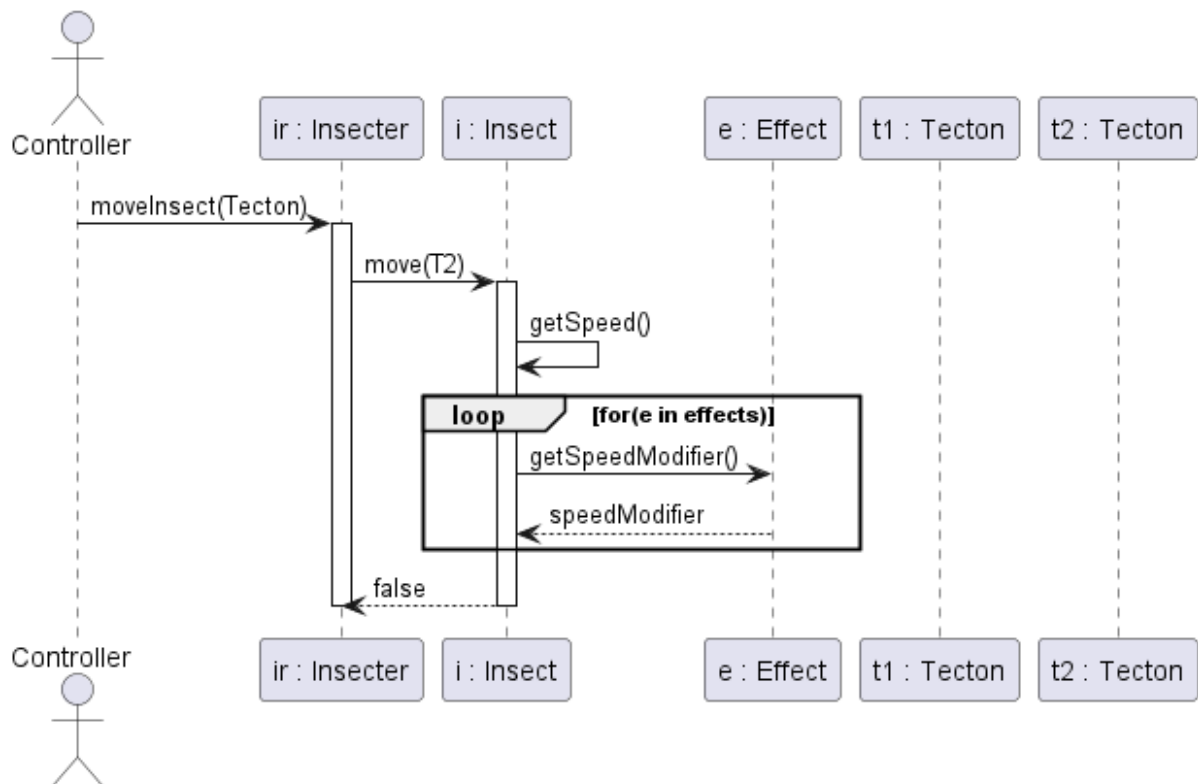


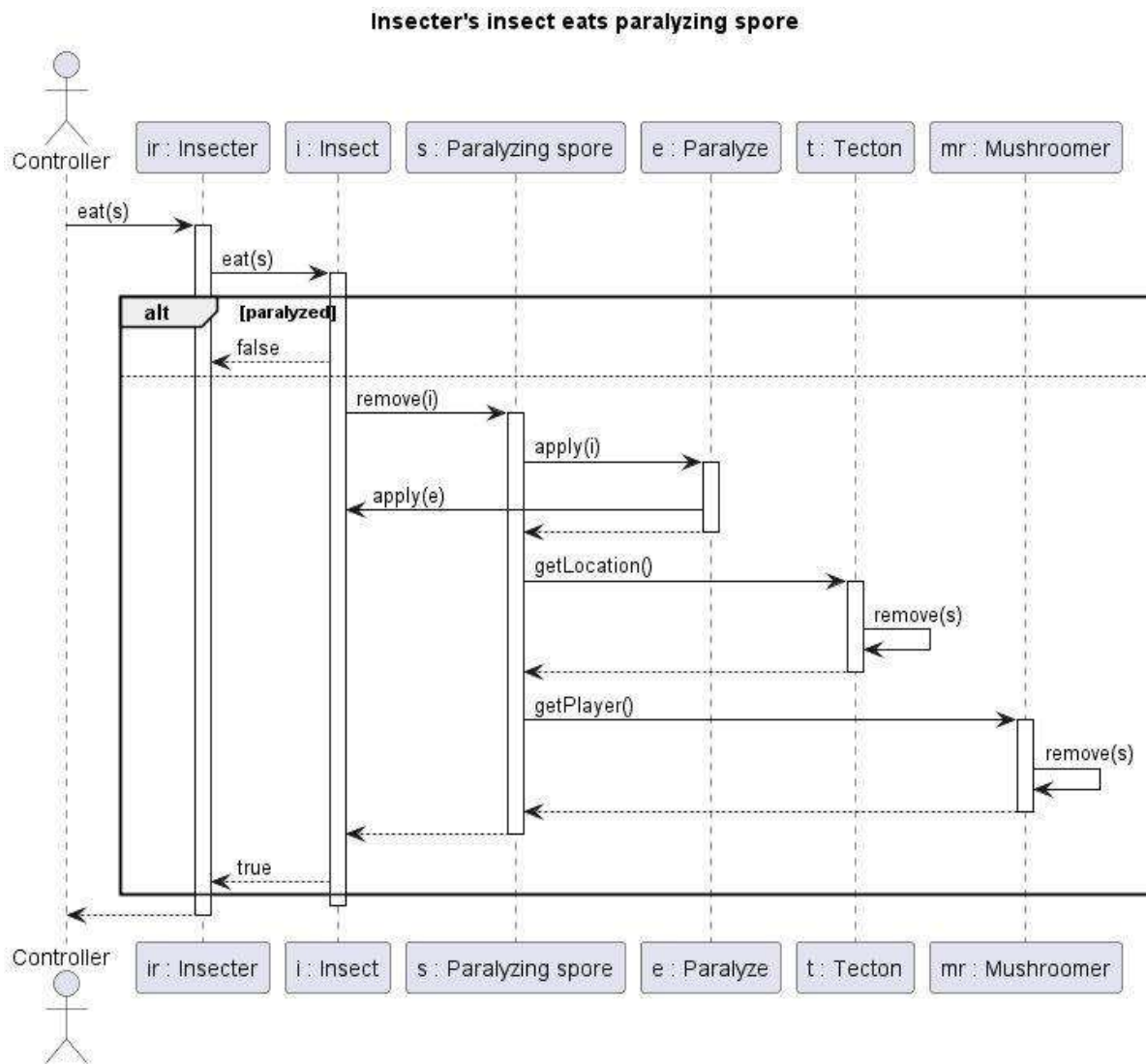


## Insecter moves insect unsuccessfully

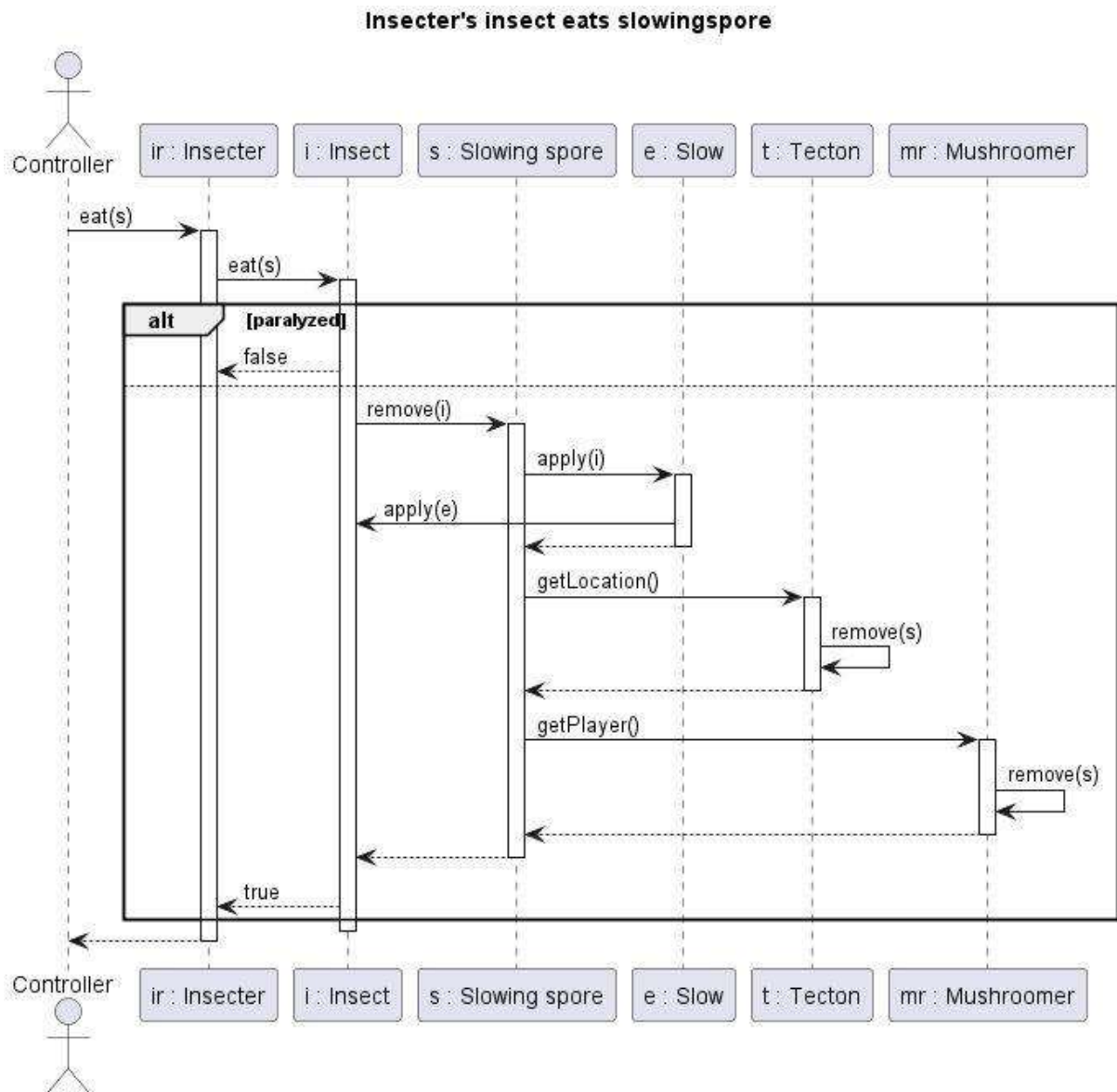


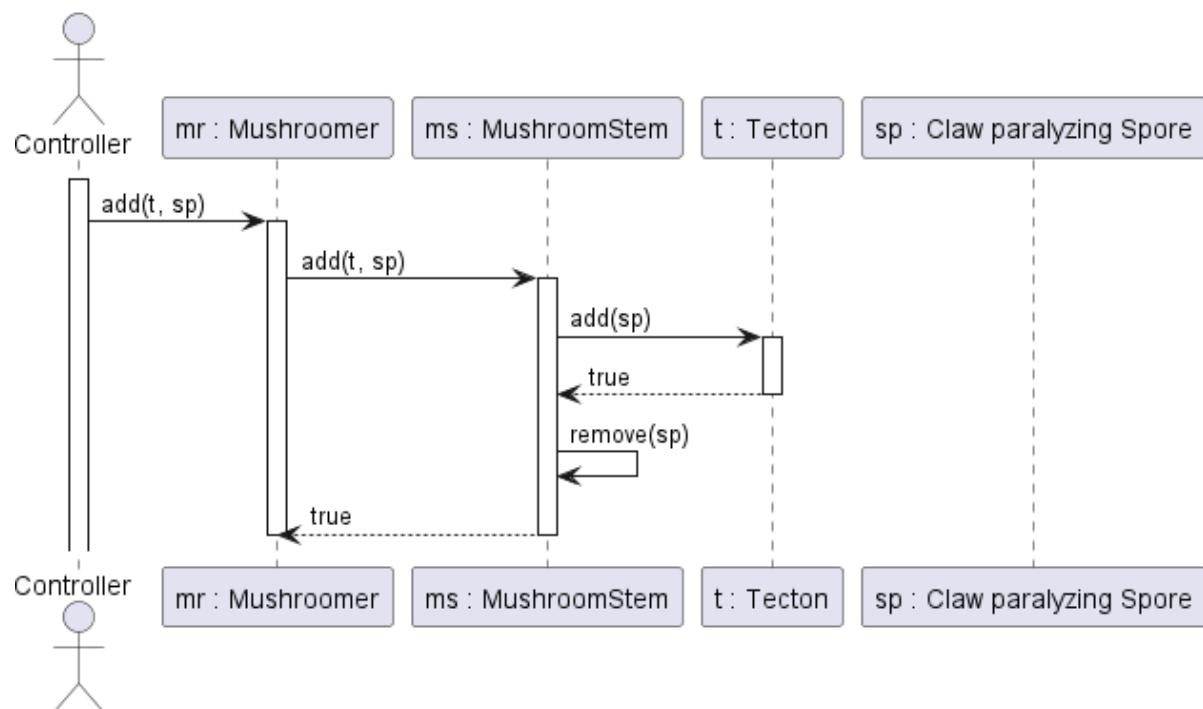
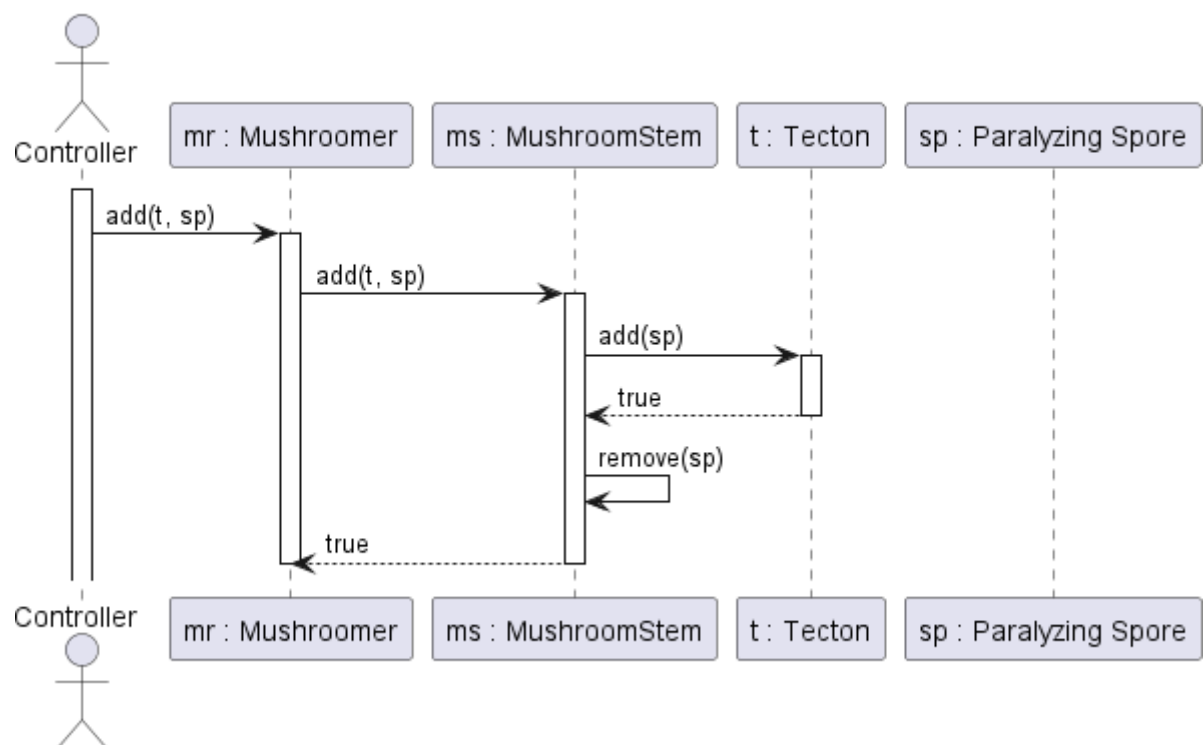
## Insecter moves insect unsuccessfully

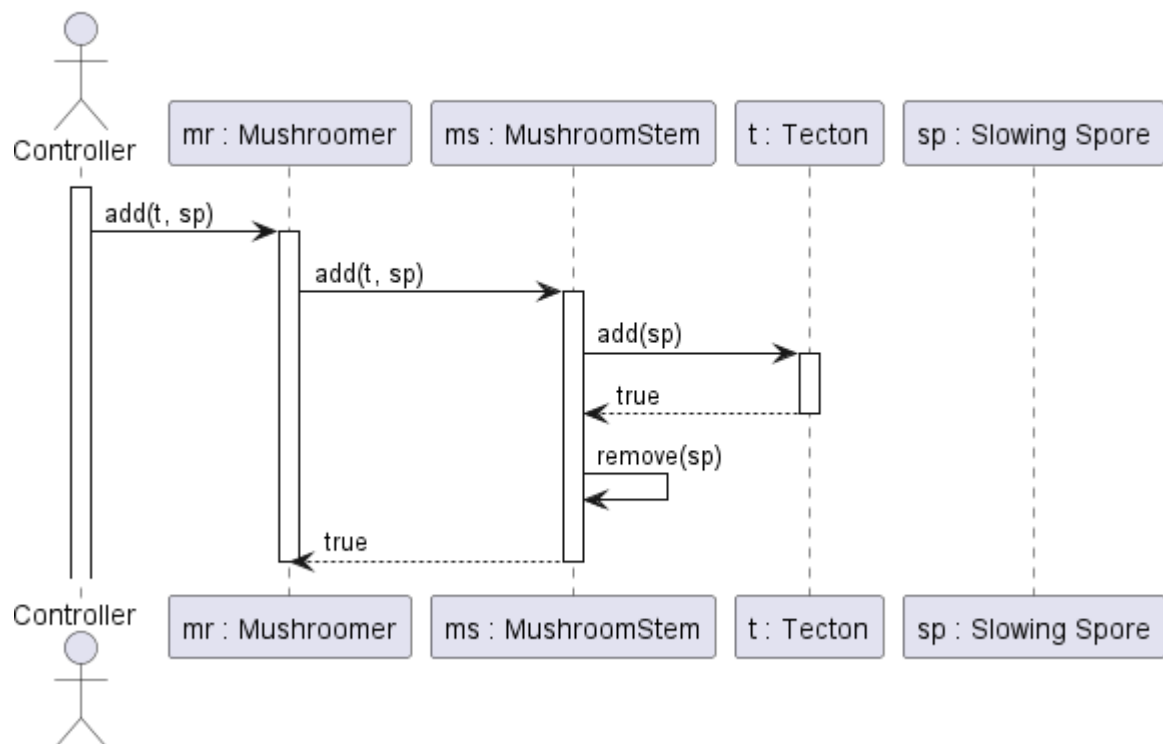
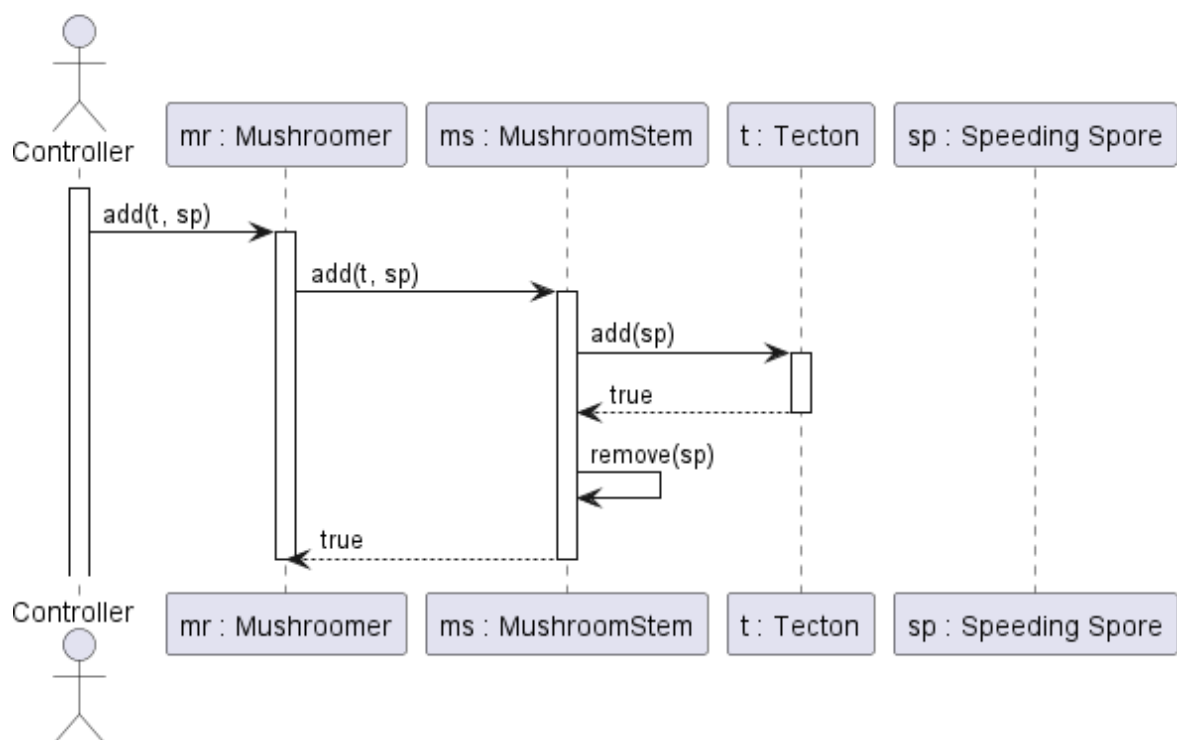


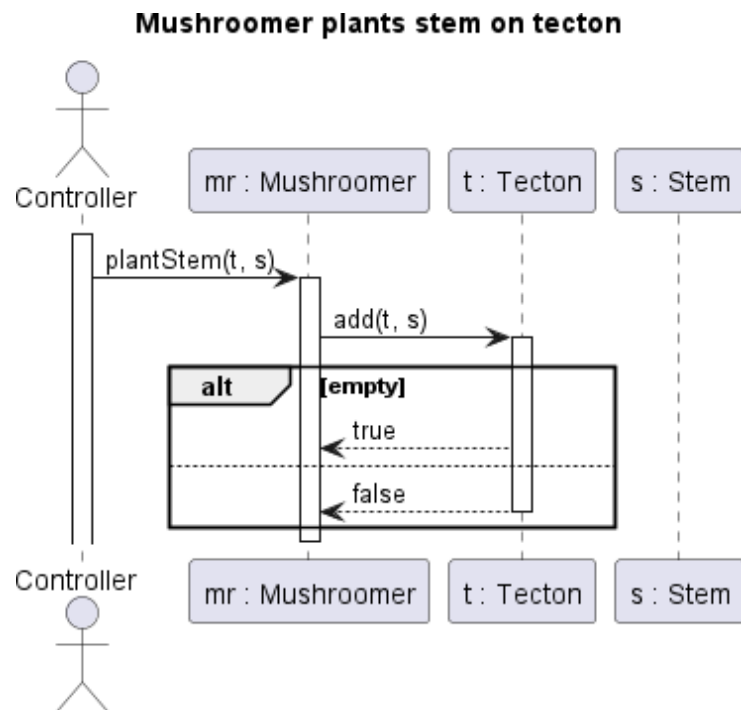


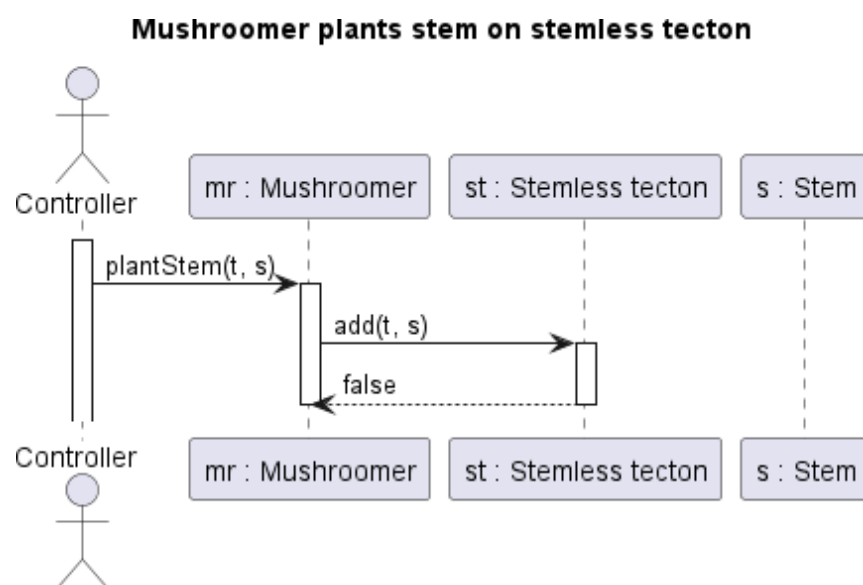
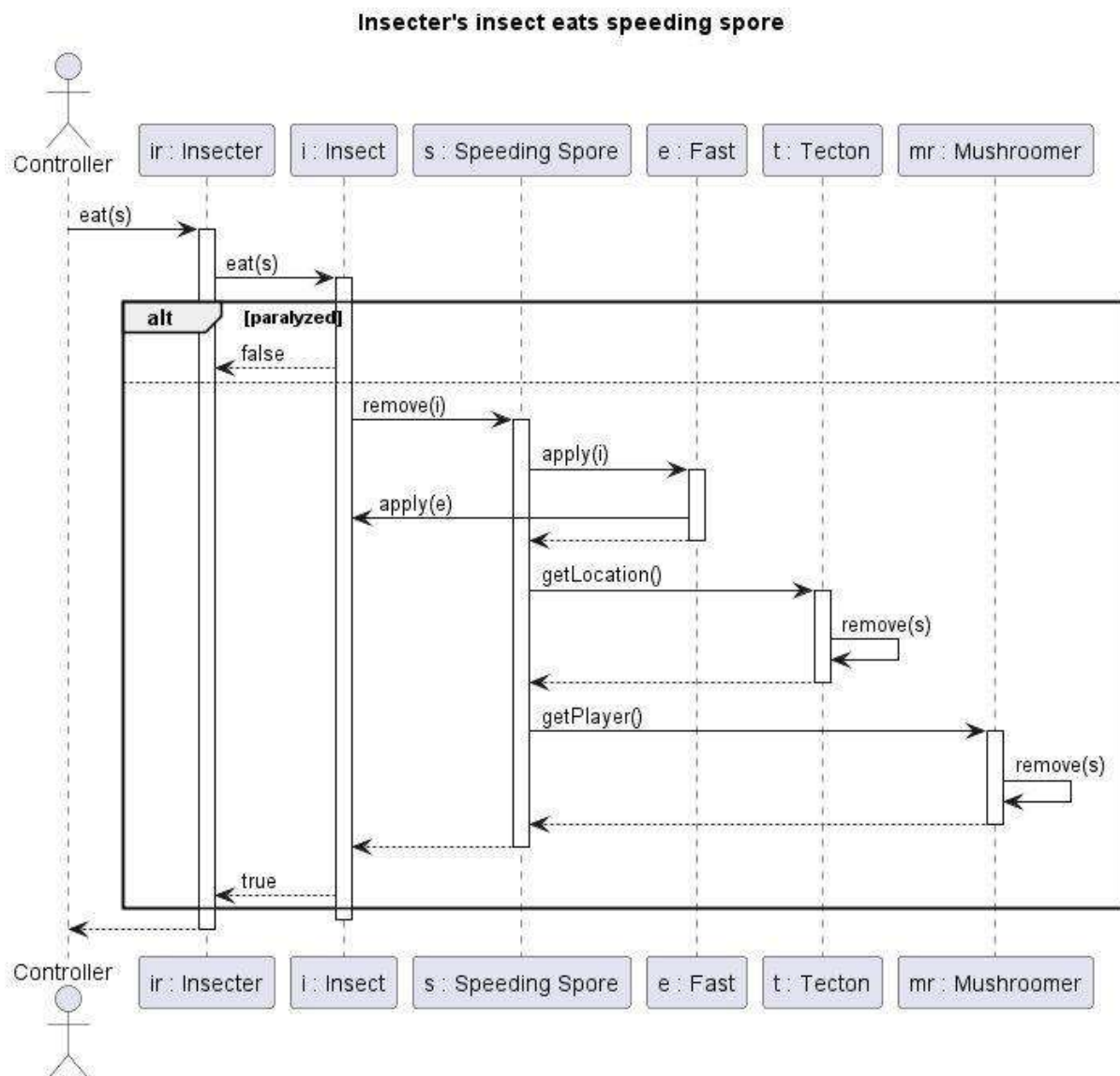


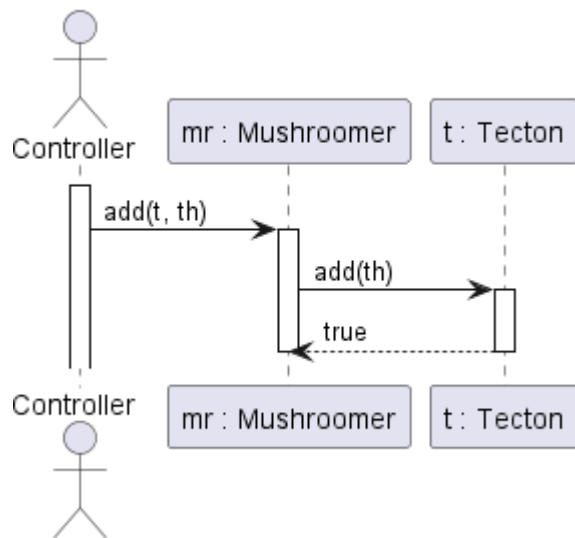
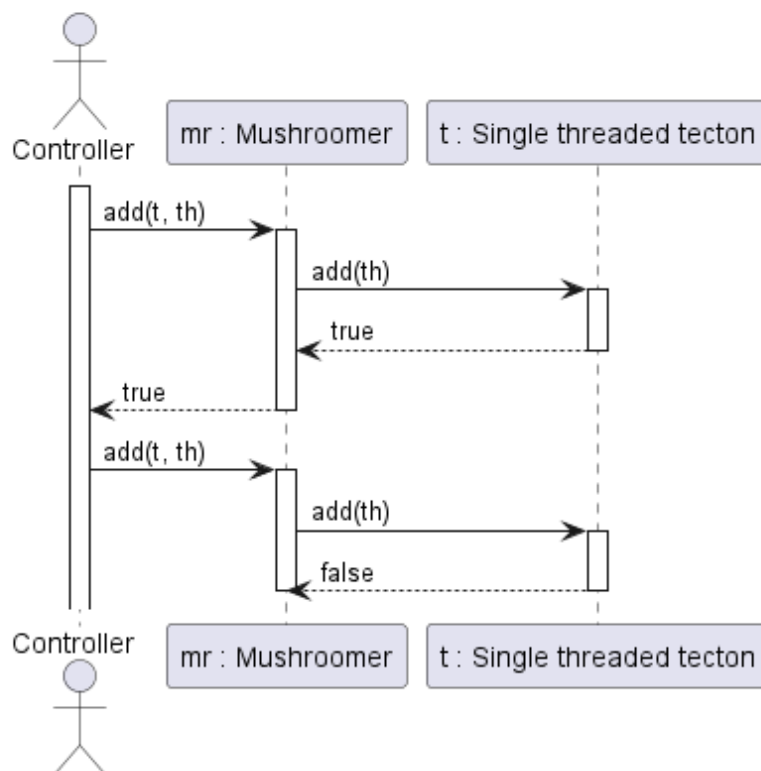


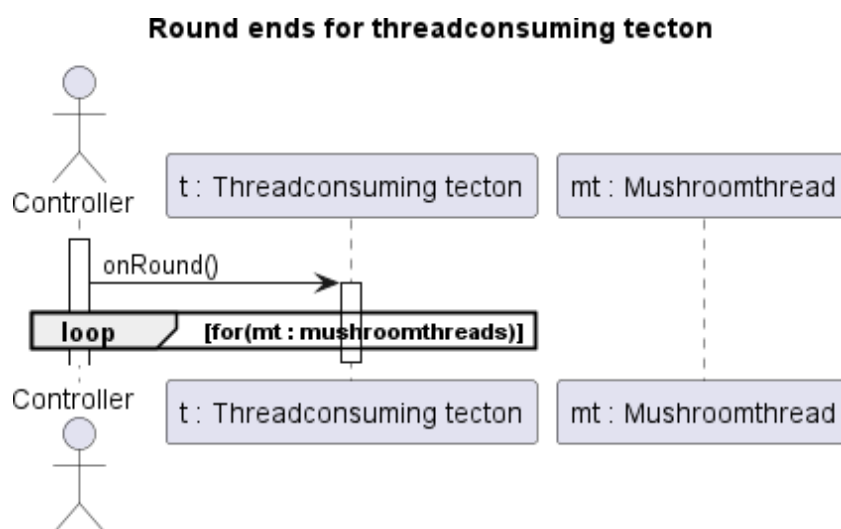
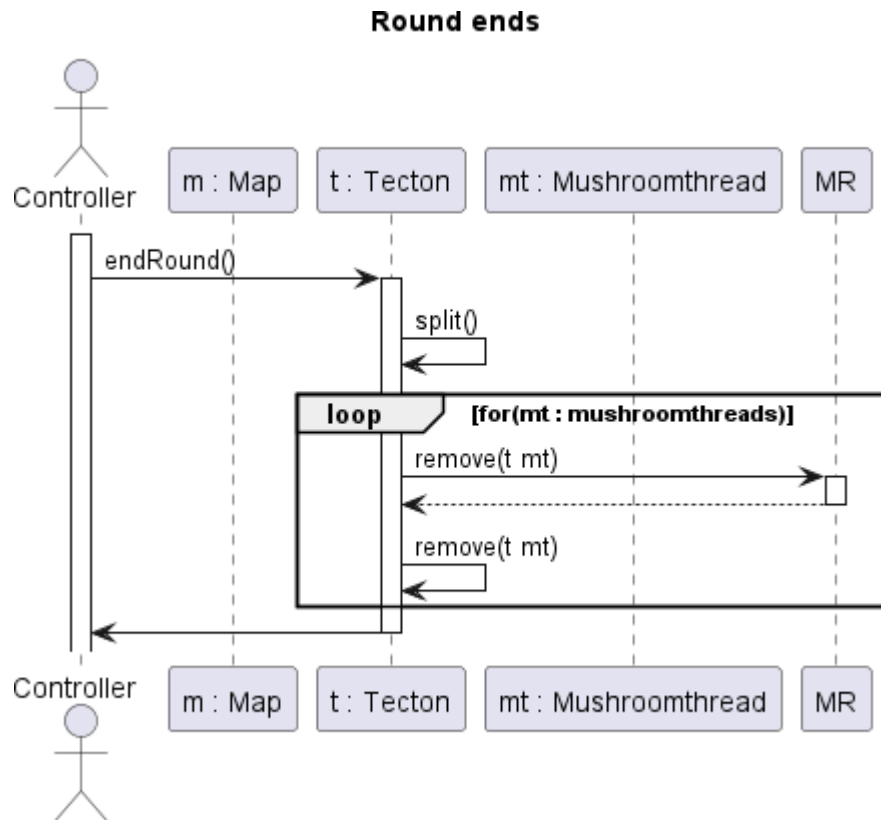
**Mushroomer throws claw paralyzing spore from stem to tecton****Mushroomer throws paralyzing spore from stem to tecton**

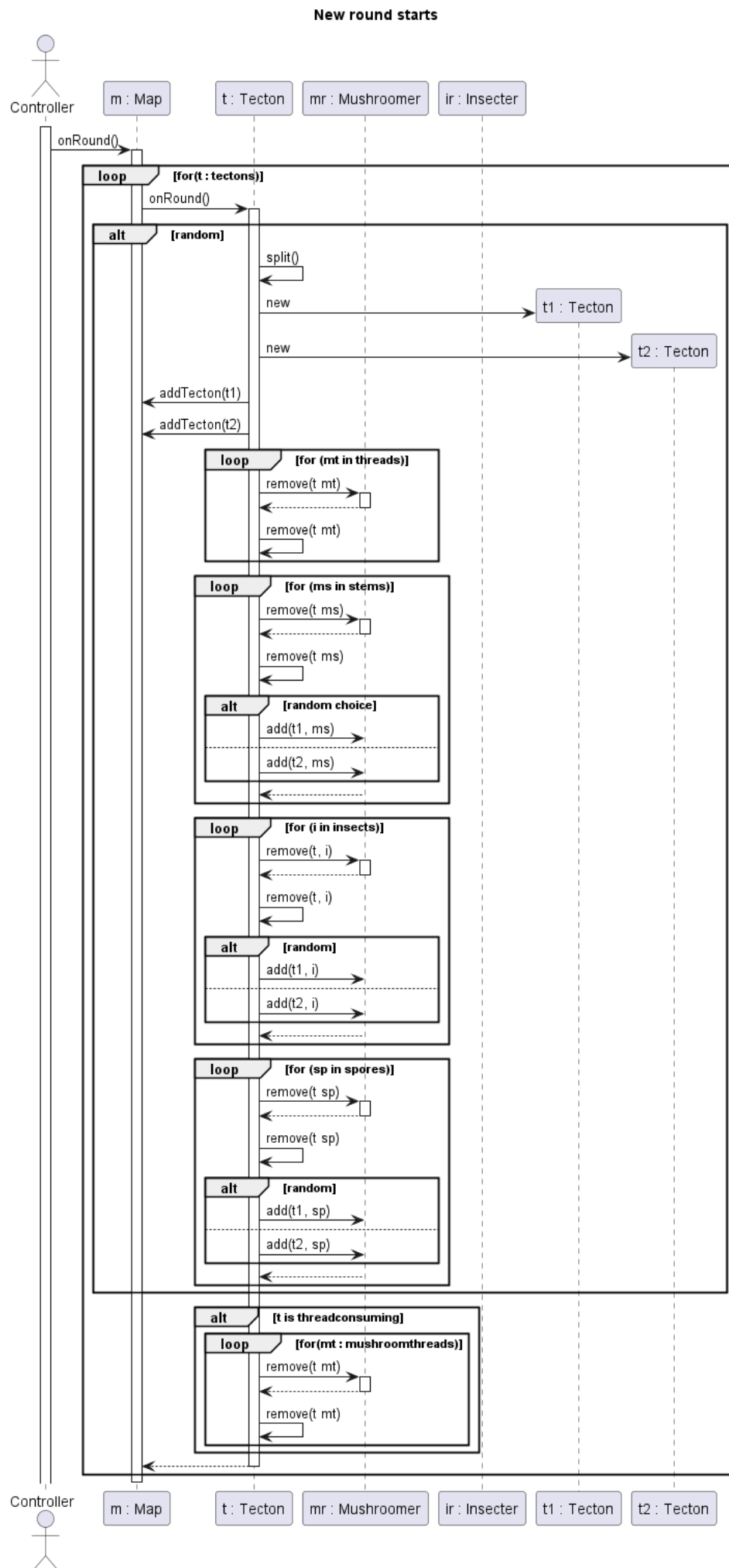
**Mushroomer throws slowing spore from stem to tecton****Mushroomer throws speeding spore from stem to tecton**





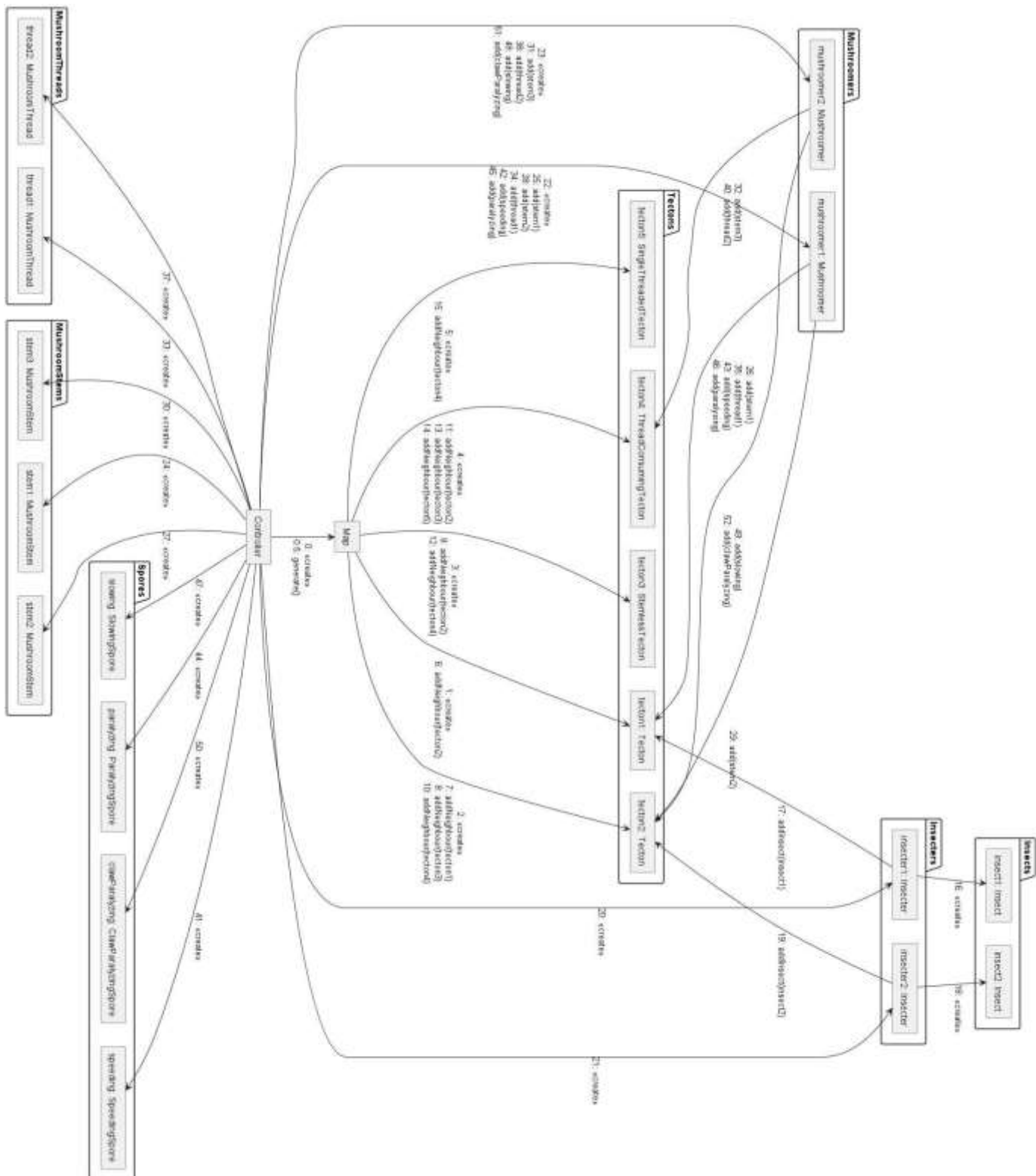
**Mushroomer grows thread on a tecton****Mushroomer grows thread on an occupied single threaded tecton**

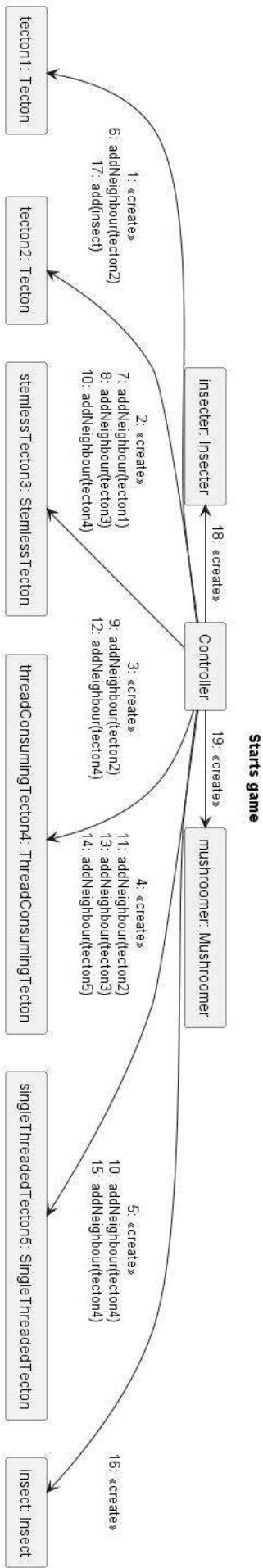






## 5.4 Kommunikációs diagramok







## 5.5 Napló

Kezdet	Időtartam	Résztevők	Leírás
2025.03.10. 21:30	1 óra	Bohus Galambos Horváth Végvári	Értekezlet. Döntés: Bohus elkészíti a kezelői felület tervét, Galambos a szekvenciadiagramok első felét, Horváth a kommunikációs diagramokat, Végvári a szekvenciadiagramok második felét.
2025.03.11. 10:15	2 óra	Végvári	Tevékenységek: Osztály diagram frissítése, use case diagram
2025.03.11 14:00	2 óra	Horváth	Tevékenységek: Kommunikációs diagram elkészítése, megtervezése
2025.03.11 20:00	2 óra	Végvári	Tevékenységek: szekvencia diagramok készítése
2025.03.12 12:00	1 óra	Bohus Galambos Horváth	Értekezlet. Döntés: Módosítottunk a feladat eloszlásán, osztálydiagramon minimális változások
2025.03.12 16:00	1 óra	Bohus	Tevékenység: 5.2-es megírása
2025.03.12 20:00	1 óra	Bohus	Tevékenység: 5.2-es javítása, use casek elkezdése
2025.03.12 20:00	2 óra	Végvári	Tevékenységek: szekvencia diagramok készítése
2025.03.13 17:00	2 óra	Horváth	Tevékenység: Kommunikációs diagram írása

2025.03.13 16:00	3,5 óra	Dúc	Tevékenység: Komm. diagramok, egyéb hibák javítása, tervek átnézése
------------------	---------	-----	---

2025.03.14 10:00	1 óra	Bohus Galambos Horváth Végvári	Értekezlet. Döntés: Módosítottunk a feladat elosztásán.
2025.03.14 11:00	2 óra	Galambos	Tevékenységek: szekvencia diagramok készítése
2025.03.14 12:00	1 óra	Horváth	Tevékenység: Kommunikációs diagram elkészítése
2025.03.14 16:00	2 óra	Végvári	Tevékenység: szekvencia diagramok elkészítése
2025.03.15 10:00	1 óra	Bohus Galambos Horváth Végvári	Értekezlet. Döntés: Tisztáztuk a use-case- eket
2025.03.15 10:00	1 óra	Galambos	Tevékenységek: Use-case, szekvencia diagramok készítése
2025.03.15 12:00	1 óra	Bohus	Tevékenységek: Use-case leírások
2025.03.15 13:30	3 óra	Horváth	Tevékenységek: Kommunikációs diagramm kiegészítése
2025.03.15 13:20	1 óra	Galambos	Tevékenységek: Use-case, szekvencia diagramok készítése
2025.03.15 20:00	1.5 óra	Dúcz	Tevékenységek: Komm. diagram, hibák javítása
2025.03.15 22:00	2 óra	Végvári	Tevékenység: szekvencia diagramok elkészítése

## 6. Szkeleton tervezése

23 – totoro

Konzulens:

Haragos Gergő Viktor

### Csapattagok

Bohus János  
Ducz Ákos  
Galambos Csaba  
Horváth Avarka  
Végvári Péter

XYOVFZ	janos.bohus1@gmail.com
GC1RTE	akos.ducz@gmail.com
BVBQKN	galambos.csaba05@gmail.com
OODMXV	avarkahorvath@gmail.com
AW3JH1	vegvaripeter21@gmail.com

## 6. Szkeleton beadás

### 6.1 Fordítási és futtatási útmutató

#### 6.1.1 Fájllista

Fájl neve	Méret	Keletkezés ideje	Tartalom
Debug.java	1KB	2025.03.22	Függvény amit a kiíráshoz használunk
Entity.java	1KB	2025.03.19	Objektumok ebből származnak le
IRound.java	1KB	2025.03.19	Interface, round végén mit kell csinálni
ITurn.java	1KB	2025.03.19	Interface, kör végén mit kell csinálni
Player.java	1KB	2025.03.19	Ősosztály a játékosoknak
ClawParalyzeEffect.java	1KB	2025.03.19	Kábító effect
Effect.java	1KB	2025.03.19	Effect osztályok ősosztálya, ezt fogja majd a rovar megkapni
FastEffect.java	1KB	2025.03.19	Gyorsító effect
ParalyzeEffect.java	1KB	2025.03.19	Kábító effect
SlowEffect.java	1KB	2025.03.19	Lassító effect
IInsect.java	1KB	2025.03.19	Insecter, és Insect osztály interface
Insect.java	6KB	2025.03.19	Rovarhoz szükséges függvények
Insecter.java	3KB	2025.03.19	Rovar irányításához szükséges függvények
ISpore.java	1KB	2025.03.19	Spóra osztályok interface
Spore.java	1KB	2025.03.19	Spóra osztályok ősosztálya
ClawParalyzing.java	1KB	2025.03.19	Kábító spóra
SlowingSpore.java	1KB	2025.03.19	Lassító spóra
SpeedingSpore.java	1KB	2025.03.19	Gyorsító spóra
IStem.java	1KB	2025.03.19	Tecton és Mushroomer interface
IThread.java	1KB	2025.03.19	Tecton és Mushroomer interface
Mushroomer.java	7KB	2025.03.19	Gombász játékos osztálya
MushroomStem.java	4KB	2025.03.19	Gombatest
MushroomThread.java	1KB	2025.03.19	Gombafonal
SingleThreadedTecton.java	1KB	2025.03.19	Olyan tecton amelyikre csak egy fajta gombafon mehet



## 6. Szkeleton beadás

totoro

SteamlessTecton.java	1KB	2025.03.19	Olyan tecton amire nem nőhet gomba
Tecton.java	8KB	2025.03.19	Alap tecton osztálya és a többi fajta tecton őssosztálya

ThreadConsumingTecton.java	1KB	2025.03.19	Olyan tecton amin könnyebben tűnnek el a gombafonalak
Main.java	11KB	2025.03.19	Tesztek megírása, program futtatása
Map.java	2KB	2025.03.19	Pálya, tektonokból áll

### 6.1.2 Fordítás

A program fordítása a Visual Studio Code-ban, Extension Pack for Java kiegészítővel történik, automatikusan futtatáskor. A fordításhoz OpenJDK 17 vagy annál újabb szükséges.

### 6.1.3 Futtatás

A futtatás szintén VSCode-ban zajlik, felül a Run->Start Debugging(F5) gombbal lehet elindítani a programot, ami az integrált terminálba fog futni, ahol majd a bementét is várja.

## 6.2 Értékelés

Tag neve	Tag neptun	Munka százalékban
Bohus János	XYOVFZ	20%
Dúcz Ákos	GC1RTE	20%
Galambos Csaba	BVBQKN	20%
Horváth Avarka	OODMXV	20%
Végvári Péter	AW3JH1	20%

### 6.3 Napló

Kezdet	Időtartam	Résztevők	Leírás
2025.03.19. 12:00	2 óra	Bohus Dúcz Galambos Horváth Végvári	Értekezlet. Döntés: Dúcz, Végvári elkezdi a kódrészletek megírását Bohus, Galambos, Horváth elkészíti a dokumentummal kapcsolatos teendőket, kódok kiegészítését, a kommenteket
2025.03.19. 15:00	5 óra	Végvári	Tevékenységek: Osztályok létrehozása
2025.03.21 17:00	1 óra	Dúcz	Tevékenység: Tesztek készítése
2025.03.22 18:00	1.5 óra	Dúcz	Tevékenység: Tesztek írása, hibák javítása
2025.03.22. 22:00	2 óra	Bohus	Tevékenység: Rovar függvényeinek írása
2025.03.22. 22:00	2 óra	Horváth	Tevékenység: Rovar függvényeinek írása, kód kommentelés
2025.03.23. 16:00	3 óra	Galambos	Tevékenység: Kommentek, függvények írása
2025.03.23. 21:00	1 óra	Bohus	Tevékenység: Dokumentum írása
2025.03.23 21:00	1 óra	Dúcz	Tevékenység: Hibák javítása

## 7. Prototípus koncepciója

23 – totoro

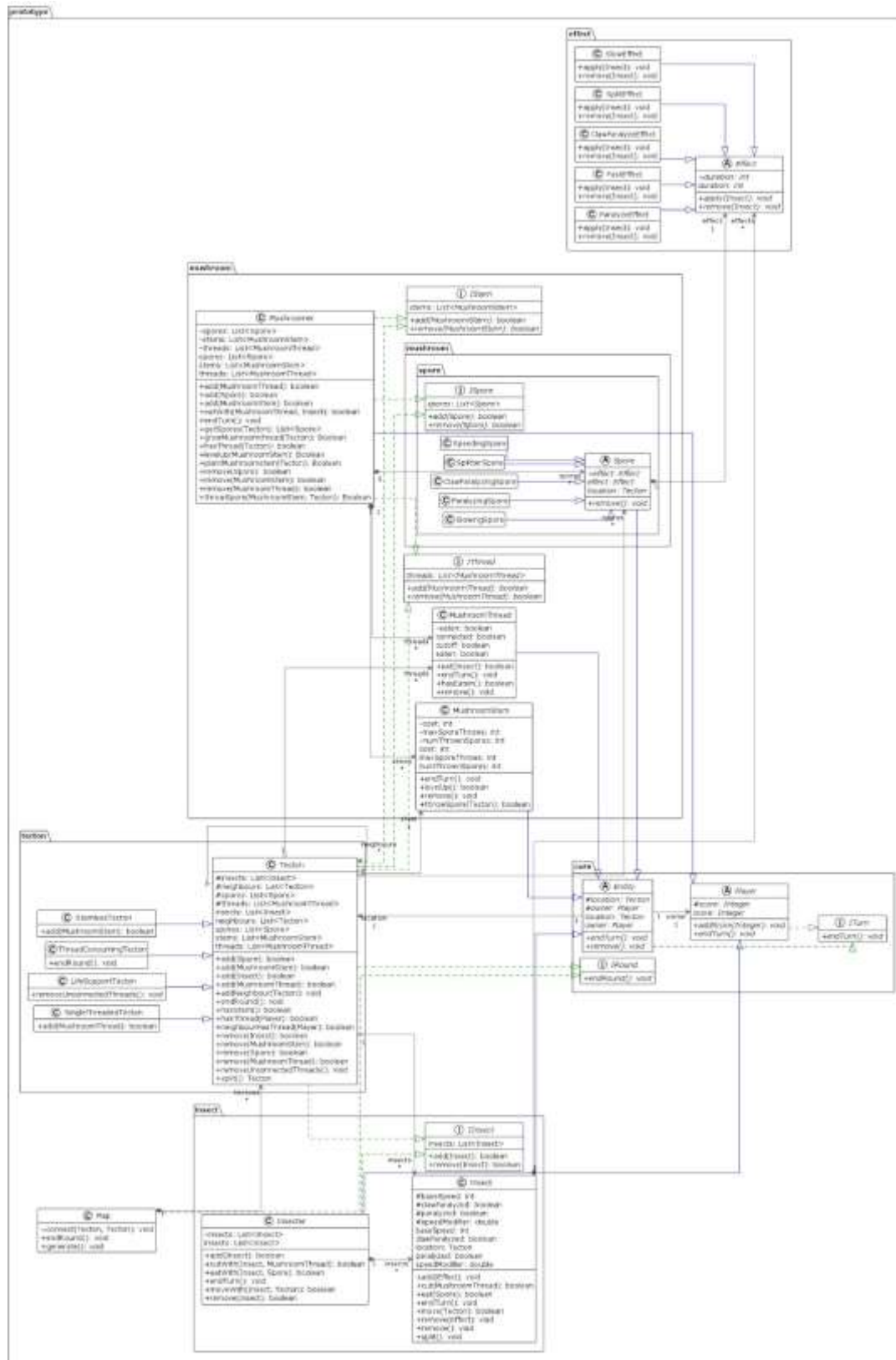
Konzulens:

Haragos Gergő Viktor

### Csapattagok

Bohus János  
Ducz Ákos  
Galambos Csaba  
Horváth Avarka  
Végvári Péter

XYOVFZ	janos.bohus1@gmail.com
GC1RTE	akos.ducz@gmail.com
BVBQKN	galambos.csaba05@gmail.com
ODMXV	avarkahorvath@gmail.com
AW3JH1	vegvaripeter21@gmail.com



## 7.0.2 Új vagy megváltozó metódusok

### LifeSupportTecton

- Ősosztály: Tecton.
- Megvalósított interfacek: IStem, IThread, ISpore, Insect.
- void removeUnconnectedThreads(): az őt tecton osztályban meghívjuk ezt mindent kör végén, LifeSupportTecton-nál viszont ez a metódus üres lesz, nem fog csinálni semmit.

### SplitterSpore

- Ősosztály: Spore.
- SplitEffect getEffect(): SplitEffect-et fogja visszaadni ha megeszi egy rovar.

### SplitEffect:

- Ősosztály: Effect.
- void apply(Insect): ketté osztja azt a rovar ami megette, ilyenkor az eredeti rovar helyére létrejön a rovarásznak még egy rovára.
- void remove(Insect): ilyenkor nem történik semmi.

### MushroomThread:

- boolean eat(Insect): egy lebénult rovarat megeszik, ilyenkor egy gombatestet növeszthetünk a gombafonalra a gombász. A művelet sikerességét adja vissza.
- boolean hasEaten(): visszaadja, hogy evett-e meg rovar.
- void setEaten(boolean): beállítja, hogy evett-e.
- void setCutOff(boolean): beállítja, hogy el lett-e vágva.
- void endTurn(): elvágástól számítva, megadott kör után távolítódik csak el a játékból.

### Insect:

- boolean add(Insect)
- boolean remove(Insect)
- List<Insect> getInsects()

### Insecter:

- boolean add(Insect): rovarászhoz új rovar ad, a művelet sikerességét adja vissza.
- boolean remove(Insect): rovarász egyik rovarját kitörli, a művelet sikerességét adja vissza.
- boolean moveWith(Insect, Tecton): rovarász egyik rovarával egy tectonra mozog, a művelet sikerességét adja vissza.
- boolean eatWith(Insect, Tecton): rovarász egyik rovarával egy spórát eszik, a művelet sikerességét adja vissza.
- boolean cutWith(Insect, Tecton): rovarász egyik rovarával gombafonalat vág el, a művelet sikerességét adja vissza.

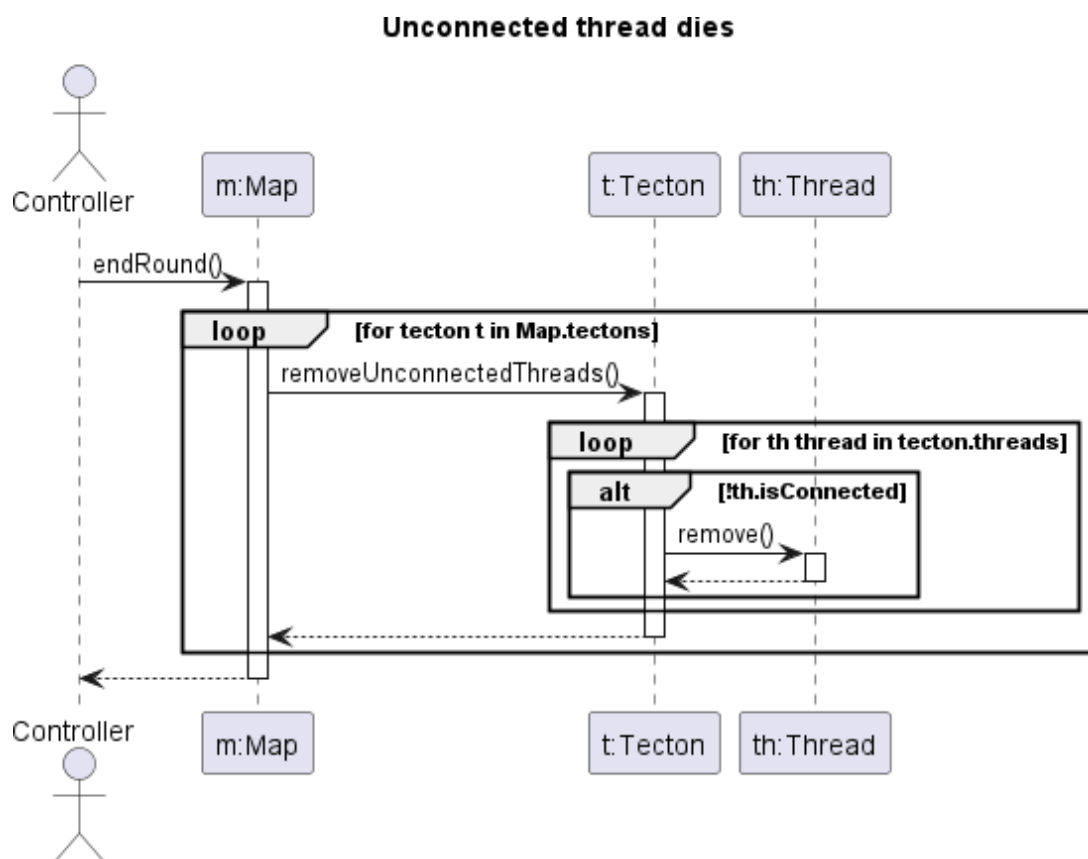
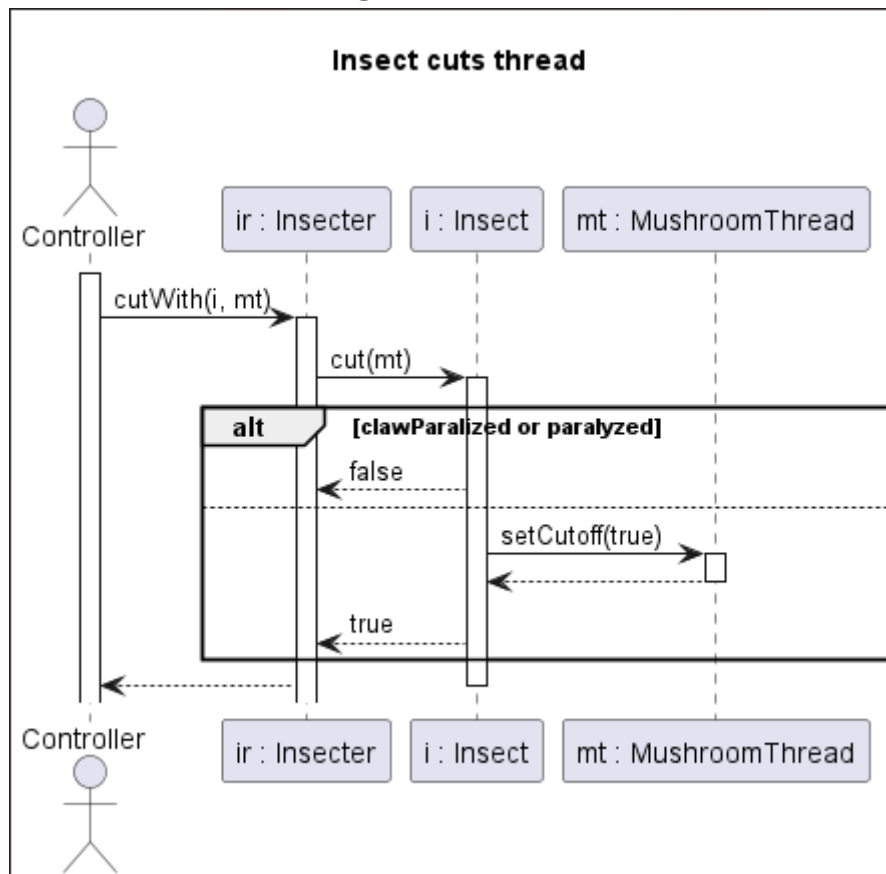
### Insect:

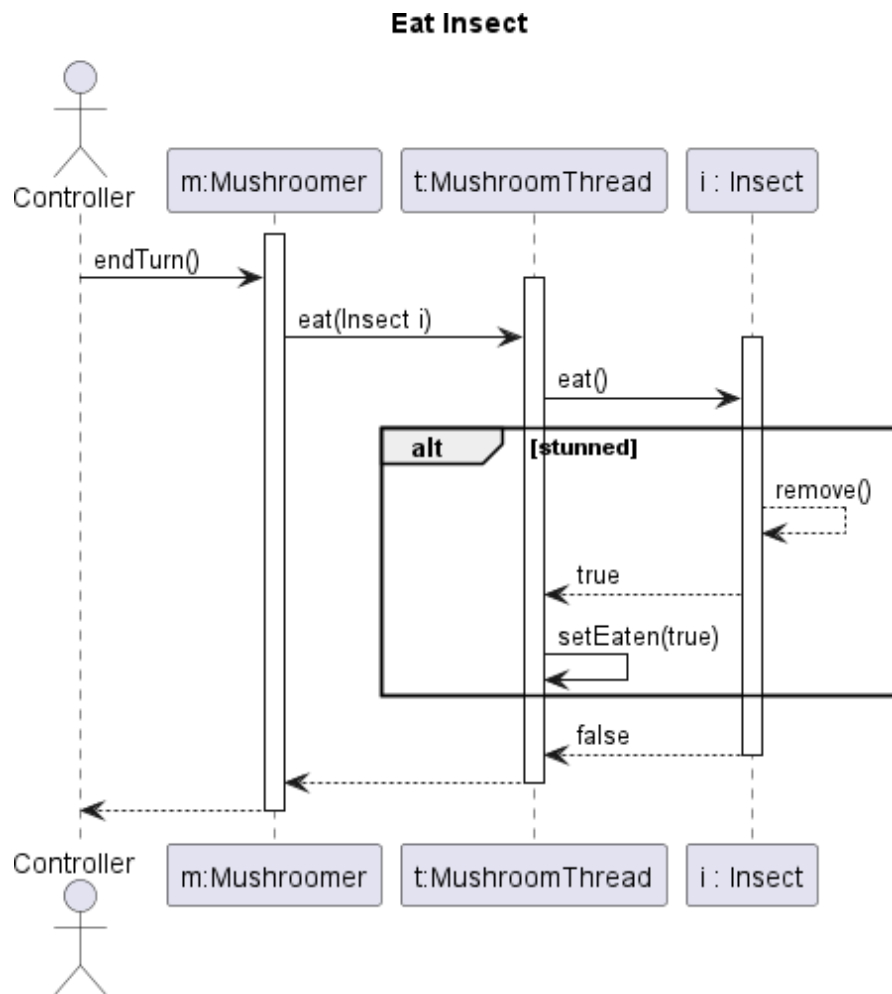
- void split(): kettéosztódik a rovar, ilyenkor a rovarász kap egy újabb rovar ugyanarra a tektonra ahol a jelenleg kettéosztódó rovar van.
- bool eat(): megnézi le van-e stunolva, és ha igen, akkor removeolja magát.

### Mushroomer:

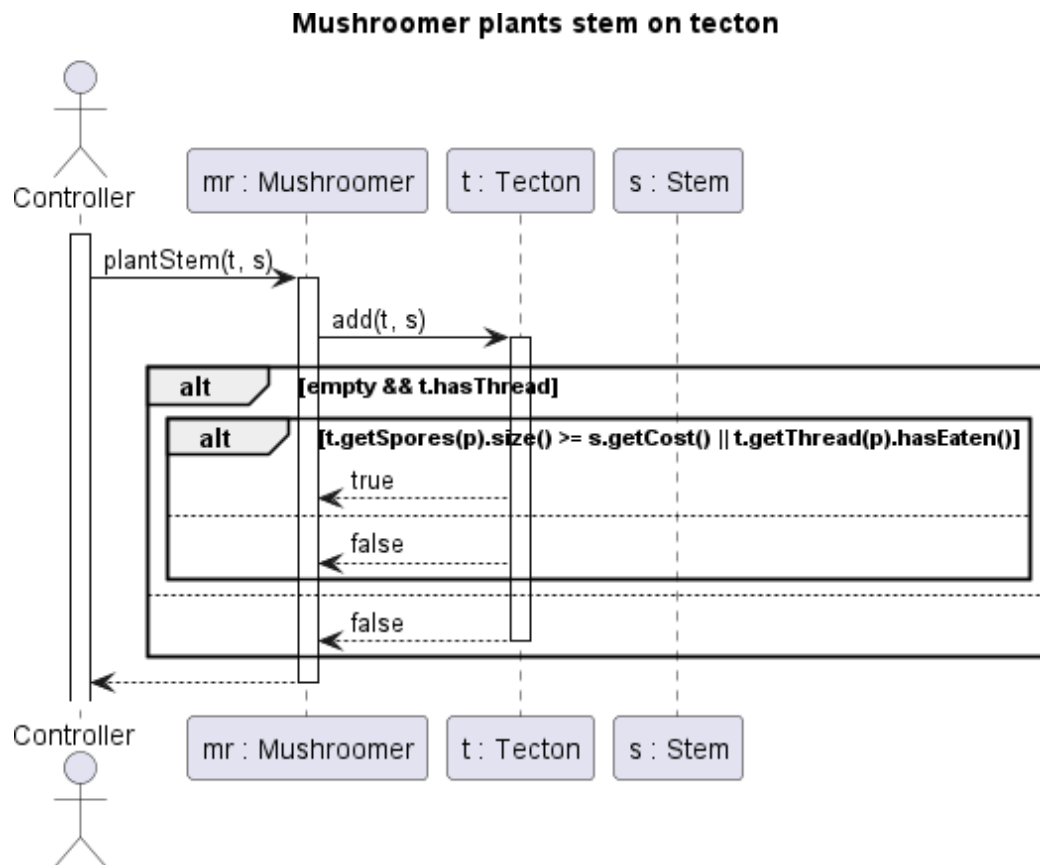
- bool plantMushroomstem(): A gombász most már nem csak kellő mennyiségű spóra esetén tud gombatestet növeszteni, hanem akkor is, ha a tektonon lévő fonal evett.

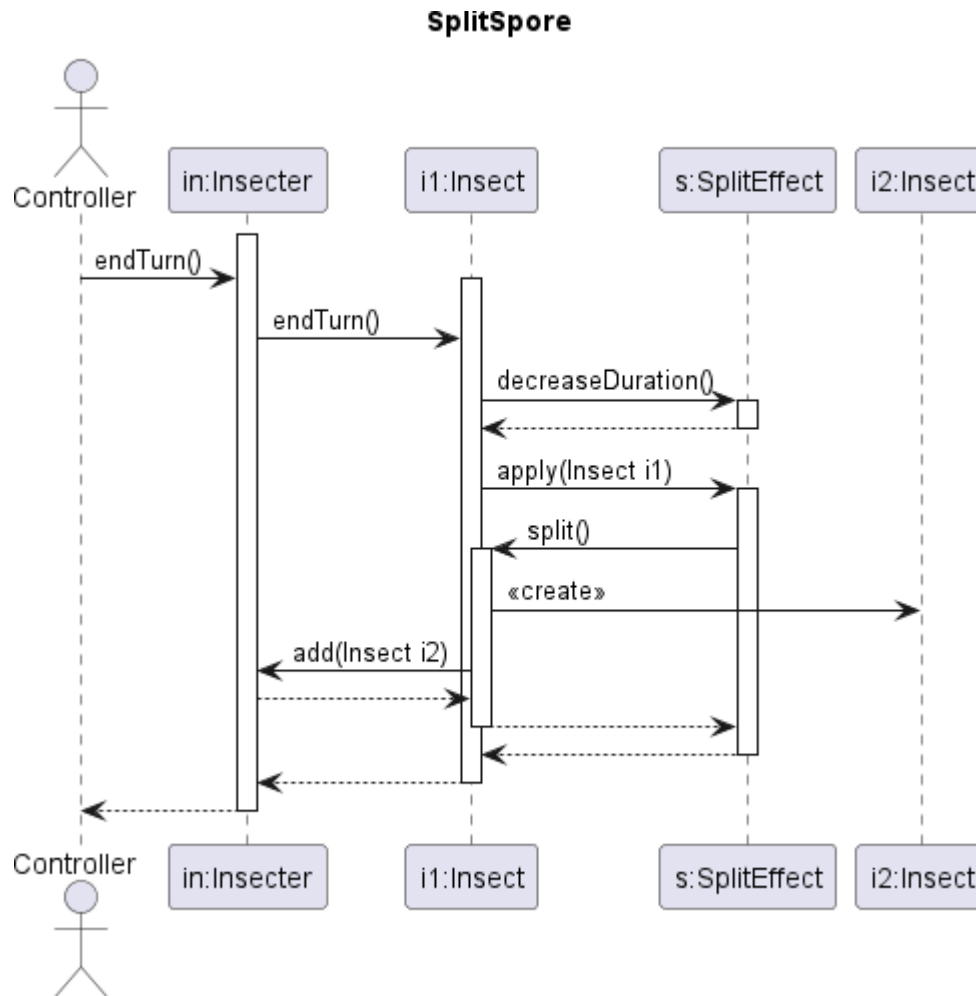
## 7.0.3 Szekvencia-diagramok











## 7.1 Prototípus interface-definíciója

### 7.1.1 Az interfész általános leírása

Az interfész jellemzői:

- **Típusa:** karakteres felület, amely a felhasználói parancsokat a szabványos bemenetről (stdin) fogadja, és a válaszokat a szabványos kimenetre (stdout) írja.
- **Parancsbevitel:** a játék futása során a felhasználó parancsokat közvetlenül a konzolon keresztül adhat meg.
- **Kimenet:** minden parancs végrehajtását követően a rendszer az érintett objektumok állapotváltozásait a konzolra írja ki.
- **Fájlból történő vezérlés:** az interfész támogatja a parancsok fájlból történő beolvasását is, lehetővé téve szkriptek futtatását.

- **Kimenet naplózása:** a konzolra írt kimenetek fájlba is irányíthatók, ezáltal visszakereshetők és összevethetők a tesztek során.
- **Serializáció:** az interfész lehetőséget biztosít az aktuális objektumállapotok JSON fájlba történő mentésére, amely támogatja az állapot visszatöltését és a tesztek determinisztikus végrehajtását.

### 7.1.2 Bemeneti nyelv

Kétféle parancs létezik:

- Rendszerparancs: / karakterrel kezdődnek, a játék általános működését befolyásolják
- Játékparancs: ! karakterrel kezdődnek, és a kör során lévő játékos által vezérelt műveleteket jelentik

#### **Rendszerparancsok:**

*/start*

Leírás: Új játék betöltése

*/load [fájl elérési útja]*

Leírás: Megadott állapotból indul a rendszer.

*/save [fájl elérési útja]*

Leírás: Jelenlegi állapot fájlba mentése.

*/exec*

Leírás: Script futtatása

*/rand*

Leírás

:

Deter

minis

ztikus

ság

kivála

sztása

Opció

k:

-enable: A játékban vannak véletlenszerű elemek.

-disable: A játékban nincsenek véletlenszerű elemek.

*/addplayers*

Leírás: Egy gombászt és egy rovarászt ad hozzá a játékhoz.

*/manualtrigger [játékparancs]*

Leírás: Egy játékparancs manuális kiváltása.

*/list*

Leírás:

Objektuj

mok

listázása

ID-kal

együtt.

Opciók:

*/map*

-player: A játékosok listázása minden hozzájuk tartozó objektummal együtt.

-mushroomstem: A gombatestek listázása ownerrel, tektonnal, szinttel együtt.

*-mushroomthread*: A gombafonal listázása ownerrel, tektonnal együtt.

*-insect*: A rovarok listázása tektonnal együtt.

*-spore*: A spóráklistázása

ownerrel, tektonnal,

típussal együtt. Leírás:

Kiírja mapot.

*/help*

Leírás: Kiírja az összes rendszer- és játékparancsot.

**Játékparancsok:***!move [insectID] [tectonID]*

Leírás: A rovar a megadott tektonra mozdítjuk.

*!eat [insectID] [sporeID]*

Leírás: A rovar megeszi a spórát.

*!cut [insectID] [mushroomthreadID]*

Leírás: A rovar elvágja a gombafonalat.

*!grow [tectonID]*

Leírás: A gombász fonalat növeszt.

*!plant [tectonID]*

Leírás: A gombász növeszt egy gombatestet.

*!throw [mushroomstemID] [sporeID] [tectonID]*

Leírás: A gombász spórát dob. A spóra random választódik ki.

*!endturn*

Leírás: A játékos befejezi a körét.

### 7.1.3 Kimeneti nyelv

A kimenet alapvetően a konzolra történik, minden parancs után kiírjuk azon objektumok régi és új állapotát, melyek állapotja a parancs hatására változott.

Formátum: [TYPE]([ID]): [CHANGED\_ATTRIBUTE]: [OLD\_VALUE]->[NEW\_VALUE]

- TYPE: Az objektum neve
- ID: Az objektum azonosítója
- CHANGED\_ATTRIBUTE: A változott attribútum neve
- OLD\_VALUE: Régi érték
- NEW\_VALUE: Új érték

Lehetőség az objektumok jelenlegi állapotát JSON fájlba kimenteni (/save parancs).

## 7.2 Összes részletes use-case

Use-case neve	SplitSpore
Rövid leírás	SplitSpore spóra dobása
Aktorok	Mushroomer
Forgatókönyv	Játékos ledob egy SplitSpore-t, amit majd ha megesz egy rovarász akkor duplikálódik a rovarja

Use-case neve	Eating with Thread
Rövid leírás	A fonál, megeszi a kábult rovar
Aktorok	Mushroomer
Forgatókönyv	A kör végén a fonál valamilyen valószínűséggel megeszi a kábult rovar

Use-case neve	Start Game
Rövid leírás	Játékos elkezd a játékot
Aktorok	User
Forgatókönyv	Játékos elkezd a játékot, a pálya inicializálódik

Use-case neve	End turn
Rövid leírás	Játékos befejezi a körét
Aktorok	User
Forgatókönyv	Játékos befejezi a körét, továbbadja a következő játékosnak, ha round vége van akkor változhat a pálya, ekkor eshetnek szét a tectonok, több fajtára is.

Use-case neve	Insect eats claw paralyzing spore
Rövid leírás	Rovar spórát eszik
Aktorok	Insecter
Forgatókönyv	Rovar megeszi a spórát, ennek következtében nem fog tudni elvágni fonalakat

Use-case neve	Moves insect
Rövid leírás	Rovar mozgása

<b>Aktorok</b>	Insecter
----------------	----------



<b>Forgatókönyv</b>	Rovarász mozgatja a rovar egy elérhető távolságban lévő tektonra
---------------------	--

<b>Use-case neve</b>	Moves insect unsuccessfully
<b>Rövid leírás</b>	Rovar sikertelen mozgatása
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovarász egy olyan tektont választ ki ami túl messze van, ennek következtében nem fog mozogni

<b>Use-case neve</b>	Insect eats paralyzing spore
<b>Rövid leírás</b>	Rovar kábító spórát eszik
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar kábító spórát eszik, ennek következtében nem fog tudni pár körig mozogni, enni, fonalat vágni, illetve megehetővé válik a gombafonalak számára

<b>Use-case neve</b>	Insect eats slowing spore
<b>Rövid leírás</b>	Rovar lassító spórát eszik
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar lassító spórát eszik, ennek következtében következő körökben kevesebbet tud mozogni

<b>Use-case neve</b>	Insect eats speeding spore
<b>Rövid leírás</b>	Rovar gyorsító spórát eszik
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar gyorsító spórát eszik, ennek következtében következő pár körben többet tud mozogni

<b>Use-case neve</b>	Cut with insect
<b>Rövid leírás</b>	Rovar elvágja a gombafonlat
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar elvágja a fonlat és az egy idő után (ami a fonlat típustól függ) eltűnik a pályáról.

<b>Use-case neve</b>	Cut with insect but can't because stun
<b>Rövid leírás</b>	Rovarnak nem sikerült a vágás
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar megpróbálja elvágni a fonlat, de nem sikerül neki, mert kábító hatás van rajta

<b>Use-case neve</b>	Cut with insect but can't because claw stun
<b>Rövid leírás</b>	Rovarnak nem sikerült a vágás
<b>Aktorok</b>	Insecter
<b>Forgatókönyv</b>	Rovar megpróbálja elvágni a fonlat, de nem sikerül neki, mert a vágóján van kábító hatás

## 7. Prototípus koncepciója

totoro

<b>Use-case neve</b>	Plants stem
<b>Rövid leírás</b>	Gomba nő
<b>Aktorok</b>	Mushroomer

<b>Forgatókönyv</b>	Ha elég spóra van egy tektonon, vagy a fonal megevett egy rovar, akkor tud rajta gombatest nőni
---------------------	---

<b>Use-case neve</b>	Plants stem on stemless tecton
<b>Rövid leírás</b>	Gombát próbálunk növesztetni, de nem sikerül
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Gombát próbálunk növesztetni egy olyan tektonra amire nem lehet, ezért nem sikerül

<b>Use-case neve</b>	Plants stem on occupied tecton
<b>Rövid leírás</b>	Gombát próbálunk növesztetni, de nem sikerül
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Gombát próbálunk növesztetni egy olyan tektonra amin már van másik gomba, ezért nem sikerül

<b>Use-case neve</b>	Throws claw paralyzing spore
<b>Rövid leírás</b>	Spóra dobása
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Csáprágó kábító spóra dobása, kiválasztunk egy tektont és rá dobjuk a spórát

<b>Use-case neve</b>	Throws paralyzing spore
<b>Rövid leírás</b>	Spóra dobása
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Kábító spóra dobása, kiválasztunk egy tektont és rá dobjuk a spórát

<b>Use-case neve</b>	Throws slowing spore
<b>Rövid leírás</b>	Spóra dobása
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Lassító spóra dobása, kiválasztunk egy tektont és rá dobjuk a spórát

<b>Use-case neve</b>	Throws speeding spore
<b>Rövid leírás</b>	Spóra dobása
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Gyorsító spóra dobása, kiválasztunk egy tektont és rá dobjuk a spórát

<b>Use-case neve</b>	Throws claw paralyzing spore but can't
<b>Rövid leírás</b>	Spóra dobása olyan tektonra ami nem lehetséges
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Csáprágó kábító spóra dobása, kiválasztunk egy olyan tektont ami túl távol van és nem érzük el, majd megpróbáljuk rádobni a spórát, de nem sikerült neki

## 7. Prototípus koncepciója

totoro

<b>Use-case neve</b>	Throws paralyzing spore but can't
<b>Rövid leírás</b>	Spóra dobása olyan tektonra ami nem lehetséges
<b>Aktorok</b>	Mushroomer

<b>Forgatókönyv</b>	Kábító spóra dobása, kiválasztunk egy olyan tektont ami túl távol van és nem érjük el, majd megpróbáljuk rádobni a spórát, de nem sikerült neki
---------------------	---

<b>Use-case neve</b>	Throws slowing spore but can't
<b>Rövid leírás</b>	Spóra dobása olyan tektonra ami nem lehetséges
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Lassító spóra dobása, kiválasztunk egy olyan tektont ami túl távol van és nem érjük el, majd megpróbáljuk rádobni a spórát, de nem sikerült neki

<b>Use-case neve</b>	Throws speeding spore but can't
<b>Rövid leírás</b>	Spóra dobása olyan tektonra ami nem lehetséges
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Gyorsító spóra dobása, kiválasztunk egy olyan tektont ami túl távol van és nem érjük el, majd megpróbáljuk rádobni a spórát, de nem sikerült neki

<b>Use-case neve</b>	Grows thread
<b>Rövid leírás</b>	Fonal növesztése
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Kiválasztjuk, hogy milyen irányba akarunk gombafonalat növesztetni

<b>Use-case neve</b>	Grows thread on occupied single threaded tecton
<b>Rövid leírás</b>	Fonal növesztése olyan tektonra, amire nem már lehet
<b>Aktorok</b>	Mushroomer
<b>Forgatókönyv</b>	Fonal növesztése olyan tektonra, amire nem már lehet

### 7.3 Tesztelési terv

<b>Teszt-eset neve</b>	Mozgás
<b>Rövid leírás</b>	Létrehozzuk a pályát és a rovar elmozdítjuk egy olyan pozícióba ami lehetséges.
<b>Teszt célja</b>	Mozgás jól működjön

<b>Teszt-eset neve</b>	Mozgás rosszul működése
<b>Rövid leírás</b>	Létrehozzuk a pályát és a rovar megpróbáljuk elmozdítani egy olyan pozícióba ahova nem tudna.
<b>Teszt célja</b>	Mozgás jól működjön

<b>Teszt-eset neve</b>	Mozgás gyorsító effect alatt
------------------------	------------------------------

## 7. Prototípus koncepciója

totoro

<b>Rövid leírás</b>	Létrehozzuk a pályát és a rovarra ráakunk gyorsító effectet a rovar elmozdítjuk egy olyan pozícióba ami lehetséges
<b>Teszt célja</b>	Mozgás jól működjön

<b>Teszt-eset neve</b>	Mozgás lassító effect alatt
<b>Rövid leírás</b>	Létrehozzuk a pályát és a rovarra rá rakunk lassító effectet a rovart elmozdítjuk egy olyan pozícióba ami lehetséges
<b>Teszt célja</b>	Mozgás jól működjön

<b>Teszt-eset neve</b>	Mozgás kábító effect alatt
<b>Rövid leírás</b>	Létrehozzuk a pályát és a rovarra kábító effectet. Próbáljuk a rovar mozgatni, de nem fog sikerülni.
<b>Teszt célja</b>	Mozgás jól működjön

<b>Teszt-eset neve</b>	Mozgás csábrágó kábító effect alatt
<b>Rövid leírás</b>	Létrehozzuk a pályát és a rovarra rá rakunk csábrágó kábító effectet. Megpróbálunk elvágni a gombafonalat ami nem fog sikerülni
<b>Teszt célja</b>	Gombafonal vágása jól működjön

<b>Teszt-eset neve</b>	Cut
<b>Rövid leírás</b>	Teszteljük, hogy a rovar jól vágja el tudja-e vágni a fonalat. Ha elvágja akkor eltűnik és azt is ellenőrizzük, hogy a gomba testtől elszeparálódott fonalak is meghalnak pár körrel később.
<b>Teszt célja</b>	Gombafonal vágás jól működjön

<b>Teszt-eset neve</b>	Spóra evése
<b>Rövid leírás</b>	Létrehozzuk a pályát és lerakunk spórát a rovar alá. Rovar megeszi és megnézzük, hogy az effect rajta van-e
<b>Teszt célja</b>	Spóra evése jól működjön

<b>Teszt-eset neve</b>	Gombafonal növesztése
<b>Rövid leírás</b>	Létrehozzuk a pályát, egyik gombából megpróbálunk gombafonalat növeszteni.
<b>Teszt célja</b>	Gombafonal növés jól működjön

<b>Teszt-eset neve</b>	Gombafonal növesztése olyan tectonra ahova nem lehet
<b>Rövid leírás</b>	Létrehozzuk a pályát, egyik gombából megpróbálunk gombafonalat növeszteni egy olyan tectonra ahova nem lehet.
<b>Teszt célja</b>	Gombafonal növés jól működjön

<b>Teszt-eset neve</b>	Gombatest növesztése
<b>Rövid leírás</b>	Létrehozzuk a pályát, egyik tectonra teszünk spórát, ott a játékos növeszt egy gombát
<b>Teszt célja</b>	Gomba növesztése jól működjön

<b>Teszt-eset neve</b>	Gombatestet próbálunk növeszteni
------------------------	----------------------------------

<b>Rövid leírás</b>	Létrehozzuk a pályát, egyik tectonra teszünk spórákat, de nem eleget ahhoz, hogy ott gomba nőjön. Ott a
---------------------	---



	játékos megpróbél egy gombát növesztetni ami nem sikerül.
<b>Teszt célja</b>	Gomba növesztése jól működjön

<b>Teszt-eset neve</b>	Spóra dobása
<b>Rövid leírás</b>	Létrehozzuk a pályát, gombatestekkel. Egyikből dobunk egy általunk kiválasztott spórát, majd megnézzük, hogy ott van-e.
<b>Teszt célja</b>	Spóra dobása jól működjön

<b>Teszt-eset neve</b>	Spóra dobása távoli tectonra
<b>Rövid leírás</b>	Létrehozzuk a pályát, gombatestekkel. Egyikből próbálunk dobni egy általunk kiválasztott spórát olyan távolságba ami nem lehetséges, majd megnézzük, hogy ott van-e.
<b>Teszt célja</b>	Spóra dobása jól működjön

<b>Teszt-eset neve</b>	Spóra dobása nagyobb szintű gomatestből
<b>Rövid leírás</b>	Létrehozzuk a pályát, gombatestekkel. Egyikből próbálunk dobni egy általunk kiválasztott spórát olyan távolságba ami nem lehetséges alap szintű gombával, csak fejlettebbel, majd megnézzük, hogy ott van-e.
<b>Teszt célja</b>	Spóra dobása jól működjön

<b>Teszt-eset neve</b>	Kör vége
<b>Rövid leírás</b>	Játékos kör végét választja, majd megnézzük, hogy a következő játékos van-e soron.
<b>Teszt célja</b>	Kör vége jól működjön

<b>Teszt-eset neve</b>	Round vége
<b>Rövid leírás</b>	Round vége lesz, ekkor tudnak osztódni a tectonok, tűnnek el a fonalak. Teszteljük, ezek sikeresek. Mivel a tecton osztódása random ezért többször is futtatjuk, hogy biztosan szétessenek.
<b>Teszt célja</b>	Round vége jól működjön

#### 7.4 Tesztelést támogató segéd- és fordítóprogramok specifikálása

Windows operációs rendszer standard parancsait használjuk, a 7.1.2-ben említett bemeneti parancsokat. Ezeket majd txt fileból olvassuk be.

Külön program nem kell hozzá

**7.5 Napló**

<b>Kezdet</b>	<b>Időtartam</b>	<b>Résztevők</b>	<b>Leírás</b>
2025.03.26. 20:00	1.5 óra	Bohus	Tevékenység: Teszt Use case írása
2025.03.26. 21:40	1 óra	Galambos	Tevékenység: Prototípus interfész definíciója, kimeneti nyelve
2025.03.26. 22:00	3 óra	Végvári	Tevékenység: Módosult osztálydiagram elkészítése
2025.03.27. 8:30	1 óra	Bohus	Tevékenység: szekvencia diagram
2025.03.27. 22:00	2 óra	Végvári	Tevékenység: Módosult osztálydiagram elkészítése
2025.03.27 18:30	2 óra	Horváth	Tevékenység: Parancsok írása
2025.03.27 20:00	1 óra	Dúcz	Tevékenység: Szekvencia diagramok készítése
2025.03.28 10:00	2 óra	Dúcz	Tevékenység: Szekvencia diagramok készítése
2025.03.28 10:00	2 óra	Galambos	Tevékenység: Szekvencia diagramok készítése
2025.03.30 22:00	1 óra	Horváth	Tevékenység: Dokumentum befejezése
2025.03.31 10:00	1 óra	Bohus	Tevékenység: Ellenőrzés, véglegesítés

## 8. Részletes tervek

23 – totoro

Konzulens:

Haragos Gergő Viktor

### Csapattagok

Bohus János  
Ducz Ákos  
Galambos Csaba  
Horváth Avarka  
Végvári Péter

XYOVFZ	janos.bohus1@gmail.com
GC1RTE	akos.ducz@gmail.com
BVBQKN	galambos.csaba05@gmail.com
ODMXV	avarkahorvath@gmail.com
AW3JH1	vegvaripeter21@gmail.com

## 8. Részletes tervek

### 8.1 Változtatások.

- JSON helyet, java binárisba fog történni a serializáció.
- Identifiable osztály, amiből mindenki leszarmazik.

### 8.2 Osztályok és metódusok tervei.

#### 8.2.1 Identifiable

- **Felelősség**

Egyedi azonosítót biztosító absztrakt osztály.

- **Ősosztályok**

- **Interfészek**

- **Attribútumok**

- **private final int id:** egyedi azonosító.

- **Metódusok**

- **public int getId():** visszaadja az azonosítót.

#### 8.2.2 Player

- **Felelősség**

Absztrakt osztály, ebből származnak le a játékosok

- **Ősosztályok**

Identifiable

- **Interfészek**

- **Attribútumok**

- **private int score:** A játékos pontszáma

- **Metódusok**

- **public void addScore(int i):** hozzáad i-t a játékos pontszámához.
- **public int getScore():** visszadja a játékos pontszámát
- **public void endTurn():** játékos befejezi a körét

### 8.2.3 Effect

- **Felelősség**

A rovarokra érvényes hatások őse. Minden spóra egy effectet tárol, ha a rovar megeszi akkor lesz érvényes rá.

- **Ősosztályok**

Identifiable

- **Interfészek**

- **Attribútumok**

- **-int duration:** Tárolja, hogy hány körig van még érvényben az effect.

- **Metódusok**

- **+ abstract apply(insect: Insect):void** Ez a függvény fogja az effectet alkalmazni a rovaron.
- **+abstract remove(insect: Insect):void** Ez a függvény törli az effectet a rovarról.
- **+decreaseDuration():void** : Csökkenti a durationt.

### 8.2.4 ParalyzeEffect

- **Felelősség**

A ParalyzeEffect osztály célja, hogy egy rovar ideiglenesen megbénítson. A hatás alatt álló rovar nem tud mozogni, spórát enni vagy gombafonalat vágni.

- **Ősosztályok**

Identifiable → Effect → ParalyzeEffect

- **Interfészek**

Nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **+apply(insect: Insect): void:** A hatás alkalmazása a célzott rovarra. Beállítja, hogy a rovar bénult állapotba kerüljön (ne tudjon aktív műveleteket végrehajtani).
- **+remove(insect: Insect): void:** Törli a bénított hatást az adott rovarról.

### 8.2.5 FastEffect

- **Felelősség**

A FastEffect osztály célja, hogy egy rovar ideiglenesen felgyorsítson. A hatás alatt álló rovar többet tud mozogni.

- **Ősosztályok**

Identifiable → Effect → FastEffect

- **Interfészek**

Nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **+apply(insect: Insect): void:** A hatás alkalmazása a célzott rovarra. Beállítja, hogy a rovar gyorsult állapotba kerüljön
- **+remove(insect: Insect): void:** A rovar sebességét visszaállítja alaphelyzetbe, törli az effectet.

### 8.2.6 ClawParalyzeEffect

- **Felelősség**

A ClawParalyzeEffect osztály célja, hogy egy rovar csáprágóját ideiglenesen megbénítsa. A hatás alatt álló rovar nem tud, spórát enni, gombafonalat vágni.

- **Ősosztályok**

Identifiable → Effect → ClawParalyzeEffect

- **Interfészek**

Nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **+apply(insect: Insect): void:** A hatás alkalmazása a célzott rovarra. Beállítja, hogy a rovar csáprágója lebénuljon.
- **+remove(insect: Insect): void:** Eltörli az effectet a rovarról.

### 8.2.7 SplitEffect

- **Felelősség**

A SplitEffect osztály célja, hogy egy rovar csáprágóját ideiglenesen megbénítsa. A hatás alatt álló rovar nem tud, spórát enni, gombafonalat vágni.

- **Össztályok**

Identifiable → Effect → SplitEffect

- **Interfészek**

Nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **+apply(insect: Insect): void:** A hatás alkalmazása a célzott rovarra. Létrejön egy új rovar.
- **+remove(insect: Insect): void:** Eltörli az effectet a rovarról.

### 8.2.8 SlowEffect

- **Felelősség**

A SlowEffect osztály célja, hogy egy rovar ideiglenesen lelassítsa. A hatás alatt álló rovar kevesebbet tud mozogni.

- **Össztályok**

Identifiable → Effect → SlowEffect

- **Interfészek**

Nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

- **+apply(insect: Insect): void:** A hatás alkalmazása a célzott rovarra. Fele annyit fog tudni a rovar mozogni.
- **+remove(insect: Insect): void:** Eltörli az effectet a rovarról.

### 8.2.9 IStem

- **Felelősség**

Gombatestek interfésze. Ez felel azért, hogy legyen add és remove függvénye azoknak az osztályoknak akik használják a gombatesteket.

- **Össztályok**

- **Interfészek**

- **Attribútumok**

- **Metódusok**

- **+add(mushroomStem: MushroomStem): boolean:** Új gombatest növesztése ezzel lehetséges.
- **+remove(mushroomStem: MushroomStem): boolean:** Gombatest törlése ezzel lehetséges, ha meghal

### 8.2.10 Mushroomer

- **Felelősség**

Gombász játékos osztálya. Ez az osztály tárolja a gombákat, irányít mindent amit a gombász tud.

- **Össztályok**

Identifiable → Player

- **Interfészek**

IStem, ISpore, IThread

- **Attribútumok**

- **-spores: List<Spore>:** Tárolja a spórákat amiket lerakott a gombász
- **-stems: List<MushroomStem>:** Gombász gombatestjei
- **-threads: List<MushroomThread>:** Gombász gombafonaljai

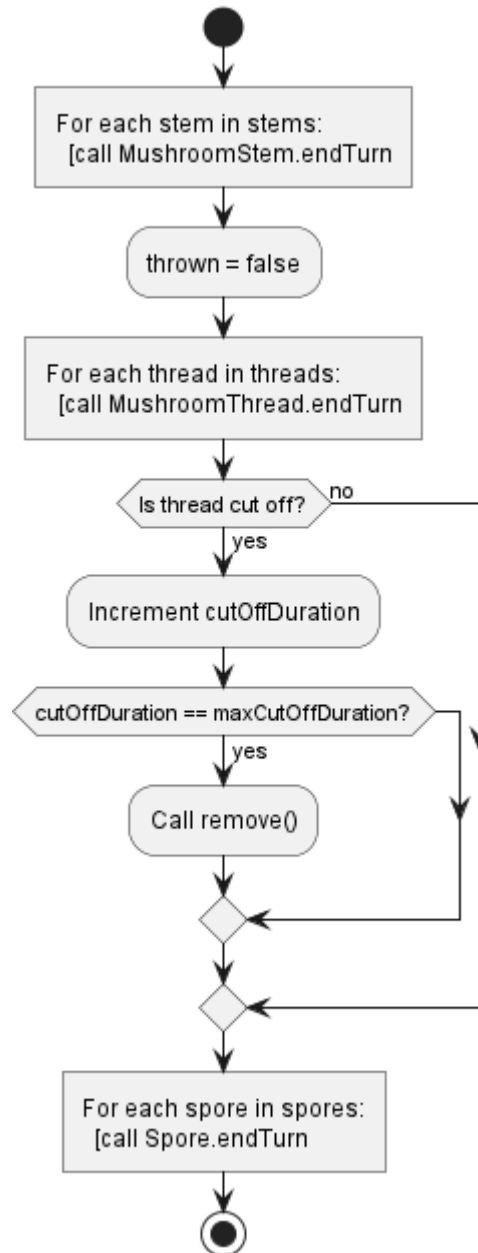
- **Metódusok**

- **add(MushroomThread): boolean:** Hozzáad egy MushroomThreadet a listához.
- **add(Spore): boolean:** Hozzáad egy Spórát a listához.
- **add(MushroomStem): boolean:** Hozzáad egy gombatestet a listához.
- **eatWith(MushroomThread, Insect): boolean:** Kiválasztja melyik fonallal eszi meg melyik rovar.



- **endTurn(): void:** Kör vége, ekkor hajtódik végre a spórák, fonalak, gombák

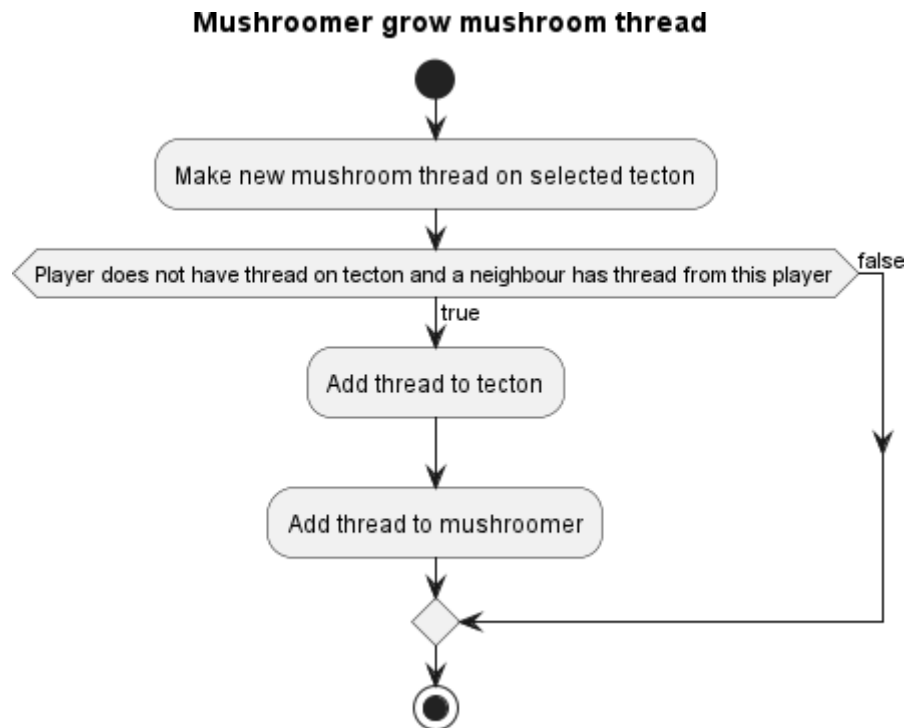
### Mushroomer End Turn Sequence



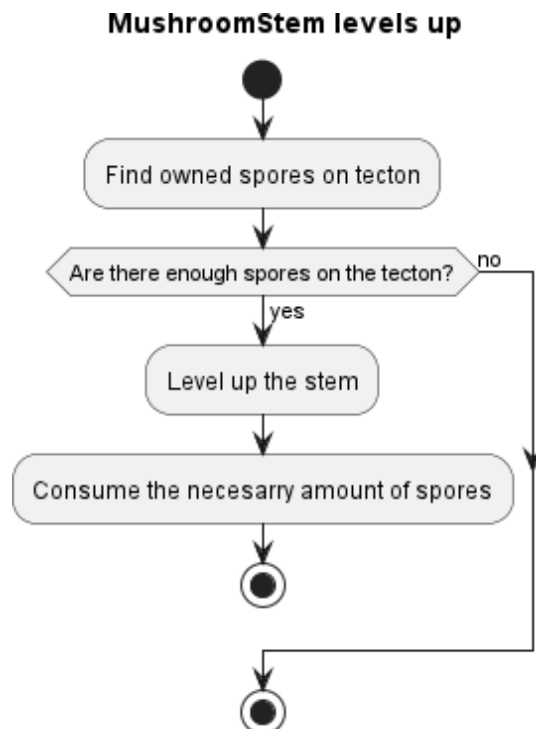
tulajdonságai.

- **getSpores(Tecton): List<Spore>:** Getter függvény a spórákhoz.

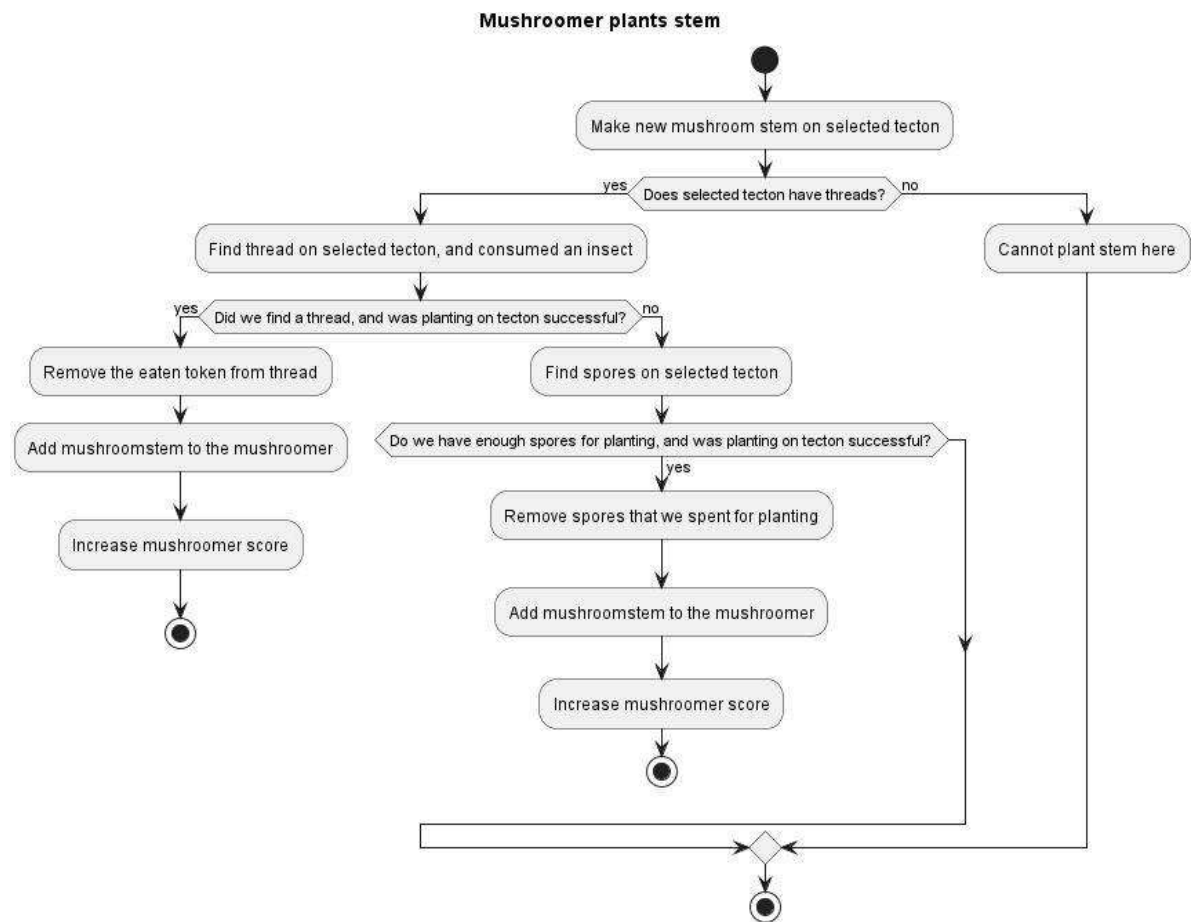
- **growMushroomthread(Tecton): boolean:** Növeszt egy fonalat, ellenőrzi, hogy lehet-e oda növesztetni.



- **hasThread(Tecton): boolean:** Megnézi hogy van e fonal az adott tectonon.
- **levelUp(MushroomStem): Boolean:** Szintet kép a gomba.

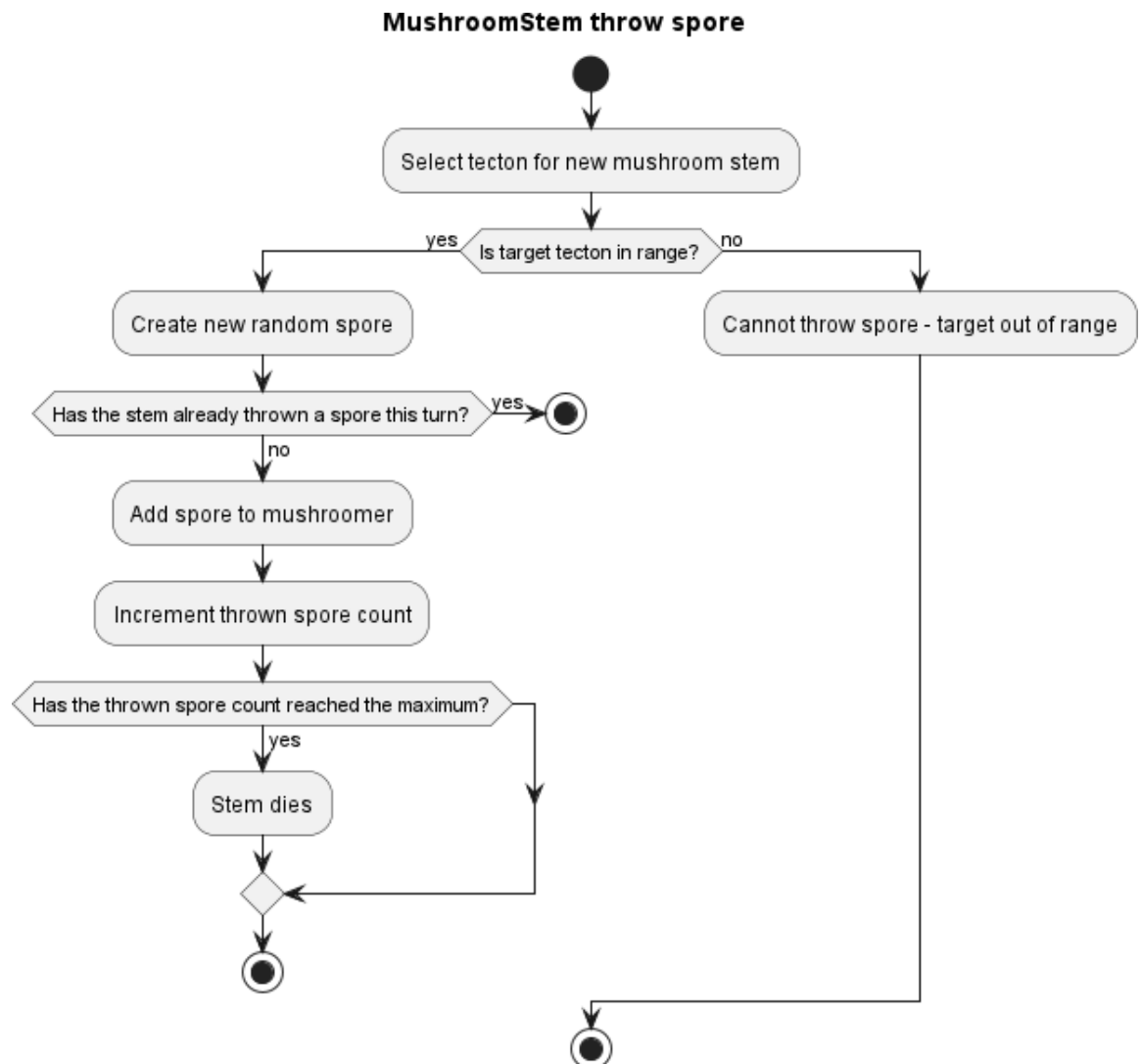


- **plantMushroomstem(Tecton): Boolean:** Növeszt egy gombatörzset, ha lehetséges.



- **remove(Spore): boolean:** Kitörli a spórát a listából.
- **remove(MushroomStem): boolean:** Kitörli a gombatestet a listából.
- **remove(MushroomThread): boolean:** Kitörli a gombafonalat a listából.

- **throwSpore(MushroomStem, Tecton): Boolean:** Dob egy általunk kiválasztott spórát az adott tectonra.



### 8.2.11 ISpore

- **Felelősség**

Spórák interfésze. Ez felel azért, hogy legyen add és remove függvénye azoknak az osztályoknak akik használják a spórákat.

- **Ősosztályok**

- **Interfészek**

- **Attribútumok**

- **Metódusok**

- **+add(spore: Spore):boolean:** Ezzel a függvénnyel lehet egy spórát lerakni, ez egy interfész függvénye, nincs megvalósítva.
- **+remove(spore: Spore):boolean:** Ezzel lehet kitörölni egy spórát, ez egy interfész függvénye, nincs megvalósítva.

### 8.2.12 Spore

- **Felelősség**

Ez a spórák őssztálya, ezt az osztályt tárolják. Absztrakt

- **Ősosztályok**

Identifiable → Entity

- **Interfészek**

- **Attribútumok**

- **private nutrition:** tápérték, amit megkap a rovarász ha az egyik rovára megeszi.
- **private effect:** hatás, amit ki fog fejteni a rovarra, ami megeszi.

- **Metódusok**

- **public Effect getEffect():** Visszaadja a hatását.
- **public int getNutrition():** Visszaadja a tápértékét.

### 8.2.13 ClawParalyzingSpore

- **Felelősség**

Olyan spóra, ami hogy ha megeszik, lebénítja a rovar csáprágóját.

- **Ősosztályok**

Indentifiable → Entity → Spore

- **Interfészek**

- **Attribútumok**

- **private nutrition:** tápérték, amit megkap a rovarász ha az egyik rovára megeszi.
- **private effect:** hatás, amit ki fog fejteni a rovarra, ami megeszi. ClawParalyzeEffect.

- **Metódusok**

- **public Effect getEffect():** Visszaadja a hatását.
- **public int getNutrition():** Visszaadja a tápértékét. ClawParalyzeEffect.

### 8.2.14 ParalyzingSpore

- **Felelősség**

Olyan spóra, ami hogy ha megeszik, lebénítja a rovar.

- **Ősosztályok**

Indentifiable → Entity → Spore

- **Interfészek**

- **Attribútumok**

- **private nutrition:** tápérték, amit megkap a rovarász ha az egyik rovára megeszi.
- **private effect:** hatás, amit ki fog fejteni a rovarra, ami megeszi. ParalyzeEffect.

- **Metódusok**

- **public Effect getEffect():** Visszaadja a hatását.
- **public int getNutrition():** Visszaadja a tápértékét. ParalyzeEffect.

### 8.2.15 SlowingSpore

- **Felelősség**

Olyan spóra, ami hogy ha megeszik, lelassítja a rovar.

- **Ősosztályok**

Identifiable → Entity → Spore

- **Interfészek**

- **Attribútumok**

- **private nutrition:** tápérték, amit megkap a rovarász ha az egyik rovára megeszi.
- **private effect:** hatás, amit ki fog fejteni a rovarra, ami megeszi. SlowingEffect.

- **Metódusok**

- **public Effect getEffect():** Visszaadja a hatását. SlowingEffect.
- **public int getNutrition():** Visszaadja a tápértékét.

### 8.2.16 SpeedingSpore

- **Felelősség**

Olyan spóra, ami ha megeszik, lelassítja a rovar.

- **Ősosztályok**

Identifiable → Entity → Spore

- **Interfészek**

- **Attribútumok**

- **private nutrition:** tápérték, amit megkap a rovarász ha az egyik rovára megeszi.
- **private effect:** hatás, amit ki fog fejteni a rovarra, ami megeszi. FastEffect.

- **Metódusok**

- **public Effect getEffect():** Visszaadja a hatását. FastEffect.
- **public int getNutrition():** Visszaadja a tápértékét.

### 8.2.17 SplittingSpore

- **Felelősség**

Olyan spóra, amit ha megesznek, osztódásra kényszeríti a rovar.

- **Ősosztályok**

Identifiable → Entity → Spore

- **Interfészek**

- **Attribútumok**

- **private nutrition:** tápérték, amit megkap a rovarász ha az egyik rovára megeszi.
- **private effect:** hatás, amit ki fog fejteni a rovarra, ami megeszi. FastEffect.

- **Metódusok**

- **public Effect getEffect():** Visszaadja a hatását. SplitEffect.
- **public int getNutrition():** Visszaadja a tápértékét.

### 8.2.18 Entity

- **Felelősség**

Absztrakt osztály. Minden olyan objektum ebből származik le, amit birtokol egy játékos.

- **Ősosztályok**

Identifiable

- **Interfészek**

- **Attribútumok**

- **Player owner:** Az a játékos, aki birtokolja az objektumot
- **Tecton tecton:** Az entitást tartalmazó tecton.

- **Metódusok**

- **public abstract void remove()**
- **public Player getOwner():** Visszaadja az objektumot birtokló játékost.
- **public Tecton getLocation():** Visszaadja a tectont, amin az entitás van.
- **public void setLocation(Tecton location):** Beállítja az entitás tartalmazó tectont
- **public void endTurn()**



### 8.2.19 Insect

- **Felelősség**

Rovar osztály, egy rovar reprezentál.

- **Ősosztályok**

Identifiable □ Entity

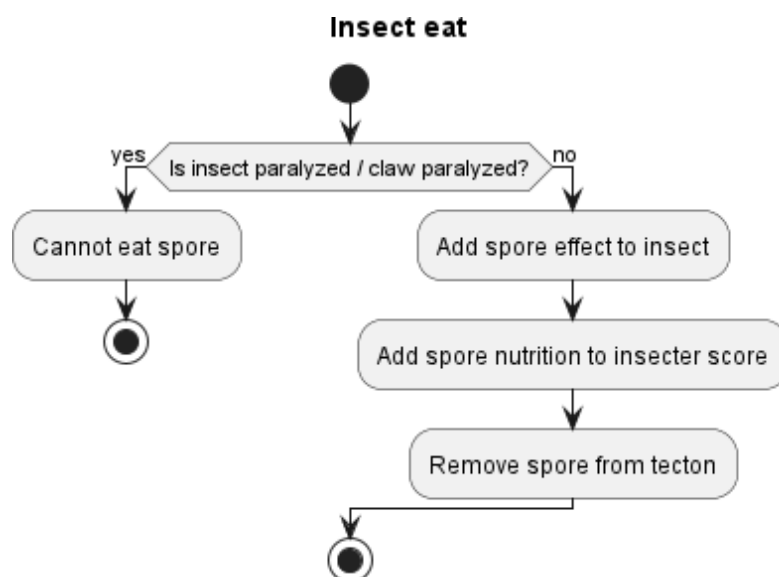
- **Interfészek**

- **Attribútumok**

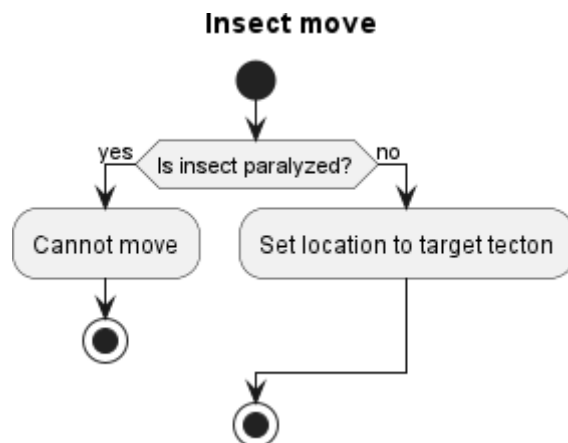
- **protected List<Effect> effects:** ilyen hatások vannak a rovaron.
- **protected boolean paralyzed:** le van-e bénulva a rovar.
- **protected boolean clawParalyzed:** le van-e bénulva a csáprágója.
- **protected int baseSpeed:** rovar alap sebessége.
- **protected double speedModifier:** rovar jelenlegi sebessége.

- **Metódusok**

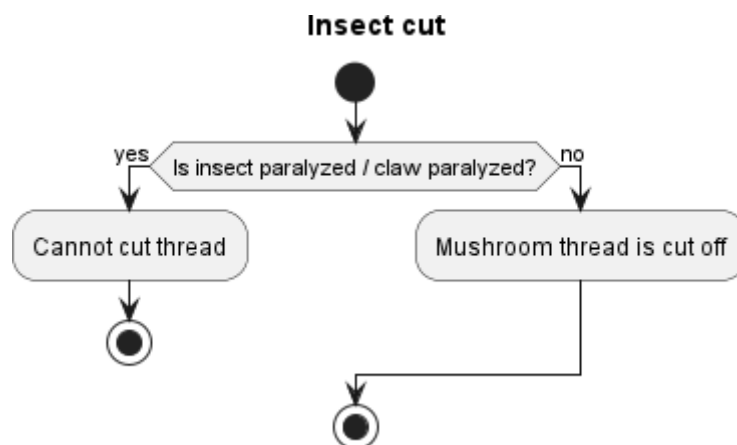
- **public boolean isParalyzed():** megállapítja, hogy a rovar bénult állapotban van-e.
- **public void setParalyzed(boolean paralyzed):** beállítja a rovar bénulási állapotát.
- **public boolean isClawParalyzed():** megállapítja, hogy a rovar csáprágója képes-e fonalat vágni.
- **public void setClawParalyzed(boolean clawParalyzed):** beállítja a karom bénulás állapotát.
- **public int getBaseSpeed():** visszaadja a rovar módosítók nélküli mozgási sebességét.
- **public double getSpeedModifier():** lekéri a sebességmódosítót.
- **public void setSpeedModifier(double speedModifier):** beállítja a sebességmódosítót.
- **public void add(Effect e):** új hatást ad a rovarhoz.
- **public void remove(Effect e):** eltávolít egy hatást a rovarról.
- **public boolean eat(Spore sp):** megpróbál elfogyasztani egy gombaspórát.



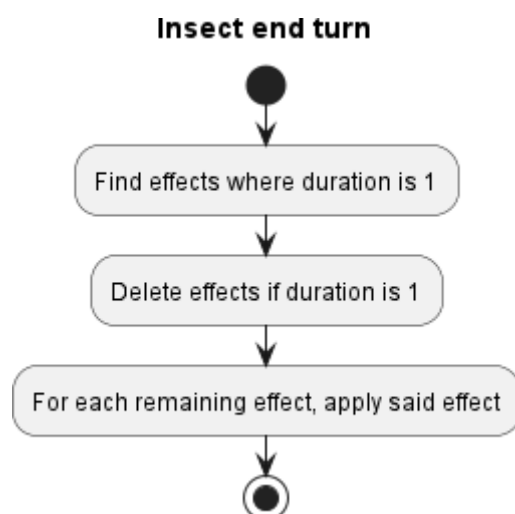
- **public boolean move(Tecton targetTecton):** megpróbál mozogni.



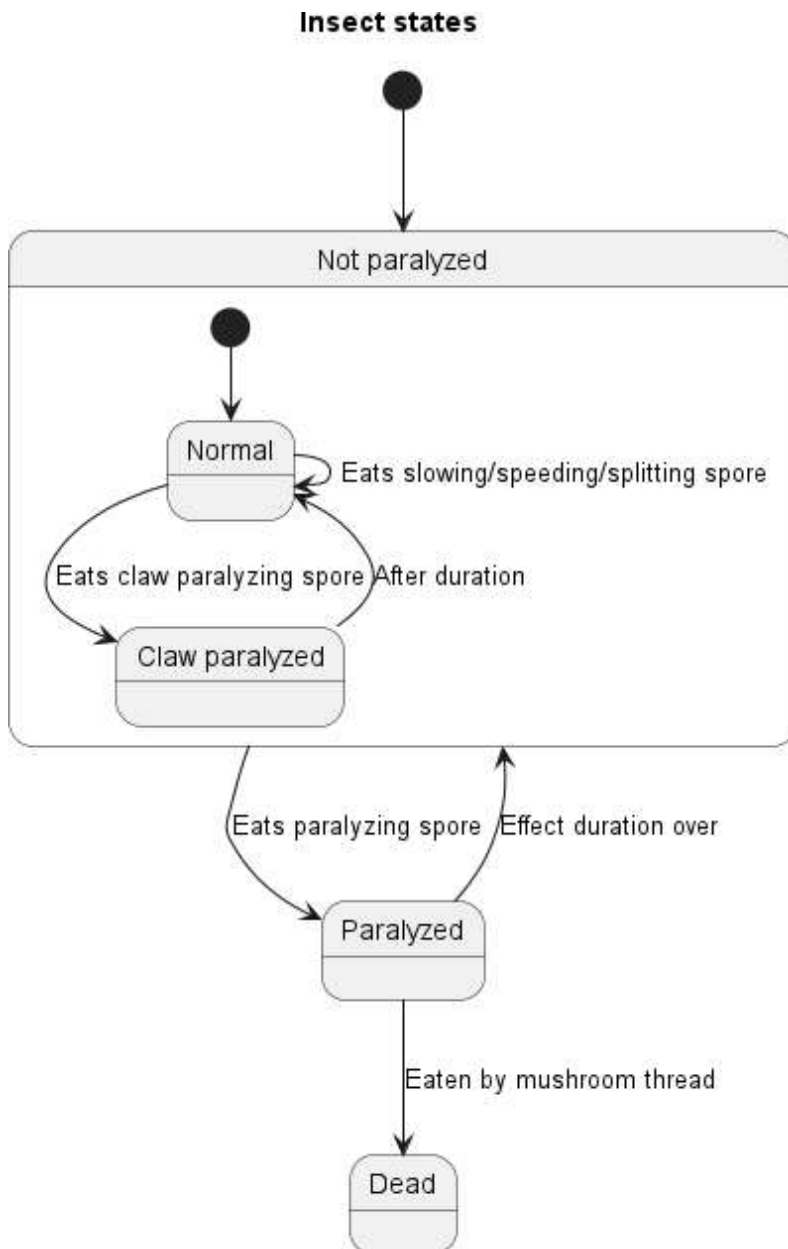
- **public boolean cut(MushroomThread th):** megpróbál elvágni egy gombafonalat.



- **public void split():** rovar kettéosztja.
- **public void setLocation(Tecton location):** beállítja a rovar helyét
- **public void endTurn():** végrehajtja a kör végén szükséges folyamatokat. Törli a lejárt hatásokat, ezután újra kiértékeli az összesített hatást.



- **public void remove():** eltávolítja a rovar.



### 8.2.20 Insecter

- **Felelősség**

Rovarász osztály megvalósítása.

- **Ősosztályok**

Identifiable -> Player

- **Interfészek**

Insect

- **Attribútumok**

- **private List<Insect> insect:** rovarász által irányított rovarok.

- **Metódusok**

- **public boolean eatWith(Insect insect, Spore sp):** megesz egy spórát az adott rovarral.
- **public boolean moveWith(Insect insect, Tecton t):** egy tectonra lép az adott rovarral.
- **public boolean cutWith(Insect insect, MushroomThread th):** gombafonalat vág el az adott rovarral.
- **public void endTurn():** befejezi a körét.
- **public boolean add(Insect insect):** hozzáadja a rovar a rovarjai közé.
- **public boolean remove(Insect insect):** kiveszi a rovar a rovarjaiból.
- **public List<Insect> getInsects():** visszaadja a rovarait.

### 8.2.21 MushroomThread

- **Felelősség**

A gombászok gombafonala, tektonokat köt össze, nő, esetleg rovart eszik

- **Ősosztályok**

Identifiable -> Entity

- **Interfészek**

- **Attribútumok**

- **-bool eaten:** Tárolja, a fonal evett-e már rovar.
- **-bool connected:** Tárolja, hogy a fonal kapcsolódik-e gombatesthez.
- **-bool cutoff:** Tárolja, hogy a fonalat elvágták-e.

- **Metódusok**

- **public bool eat(insect: Insect):** Megeszik egy rovar, és beállítja az eaten változót.
- **public void remove(insect: Insect):** Eltávolítja a fonalat a játékból.
- **public void endTurn():** Csökkenti a fonal hátralévő idejét, ha el van vágva, vagy nem connected.
- **public bool hasEaten():** Visszaadja az eaten változó értékét.
- **public void setEaten(bool: Boolean):** Beállítja az eaten változó értékét.

- **public void setCutoff(bool: Boolean):** Beállítja a cutoff változó értékét.
- **public bool isConnected():** Visszaadja az eaten változó értékét.

*isConnected():*

*indulóTecton = this.location*

*if indulóTecton.hasMushroomSteam():*

*return true*

*látogatott = üres halmaz*

*sor = új üres sor (pl. Queue)*

*sor.push(indulóTecton)*

*látogatott.add(indulóTecton)*

*amíg sor nem üres:*

*aktuális = sor.pop()*

*ha aktuális.hasMushroomSteam():*

*return true*

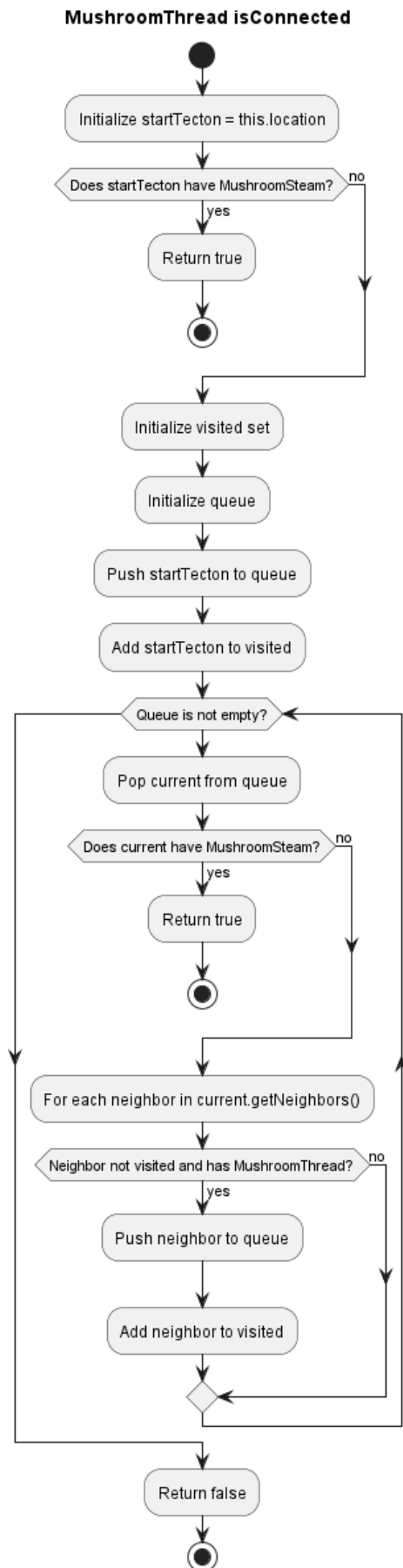
*minden szomszéd in aktuális.getNeighbors():*

*if szomszéd nincs a látogatott-ban ÉS szomszéd.hasMushroomThread():*

*sor.push(szomszéd)*

*látogatott.add(szomszéd)*

*return false*



## 8.2.22 MushroomStem

- **Felelősség**

A gombászok gombateste, spórákat lő, pontot ér és a fonalaknak szüksége van rá, hogy életben maradjon

- **Ősosztályok**

Identifiable -> Entity

- **Interfészek**

- **Attribútumok**

- **int cost:** Tárolja a gombatest növesztés költségét.
- **int maxSporeThrows:** Tárolja, hogy hány spóradobás után hal meg a gombatest.
- **int numThrownSpores:** Tárolja, hány spórát dobott a gombatest.
- **int level:** Tárolja, hanyas szintű a gombatest

- **Metódusok**

- **public bool levelUp():** Növeli a level változó értékét.
- **public void remove(insect: Insect):** Eltávolítja a fonalat a játékból.
- **public void endTurn():** Csökkenti a fonal hátralévő idejét, ha el van vágva, vagy nem connected.
- **public bool throwSpore(tecton: Tecton):** Spórát dob a megadott tektonra.
- **public int getCost():** Visszaadja a cost változó értékét.
- **public int getMaxSporeThrows():** Visszaadja a maxSporeThrows változó értékét.
- **public int getNumThrownSpores():** Visszaadja a numThrownSpores változó értékét.

### 8.2.23 Tecton

- **Felelősség**

A pálya építőelemei, a gombák és rovarok a tektonokon vannak, élnek, mozognak. . .

- **Ősosztályok**

Identifiable

- **Interfészek**

IRound, IThread, ISpore, IStem

- **Attribútumok**

- **List<spore> spores:** Tárolja a tektonon levő spórákat.
- **MushroomStem stem:** Tárolja a tektonon levő gombatestet.
- **List<tecton> neighbours:** Tárolja a tektonnal szomszédos tektonokat.
- **List<MushroomThread> threads:** Tárolja a tektonon levő gombafonalakat.
- **List<Insect> insects:** Tárolja a tektonon levő rovarokat.

- **Metódusok**

- **public List<spore> getSpores():** Visszaadja a tektonon levő spórák listáját.
- **public void remove(insect: Insect):** eltávolítja a fonalat a játékból.
- **public void endTurn():** Csökkenti a fonal hátralévő idejét, ha el van vágva, vagy nem connected.
- **public void addNeighbour(tecton: Tecton):** Hozzáad egy szomszédot a szomszédokhoz.
- **public tecton split():** Kettétöri a tektont, és visszaadja az újat.

*split():*

```
newTecton = new Tecton()
```

```
randomNumber = random between 0 and 3
```

```
if randomNumber == 0:
```

```
    newTecton = new SingleThreadedTecton()
```

```
else if randomNumber == 1:
```

```
    newTecton = new StemlessTecton()
```

```
else if randomNumber == 2:
```

```
    newTecton = new ThreadConsumingTecton()
```

```
for each neighbor in currentTecton.neighbors:
```

```
    if random true or false:
```

```
        newTecton.addNeighbour(neighbor)
```

```
currentTecton.threads.clear()
```

```
sporesToMove = empty list
```

```
for each spore in currentTecton.spores:
```

```
    if random true or false:
```

```
        sporesToMove.add(spore)
```

```
for each spore in sporesToMove:
```

```
    spore.setLocation(newTecton)
```



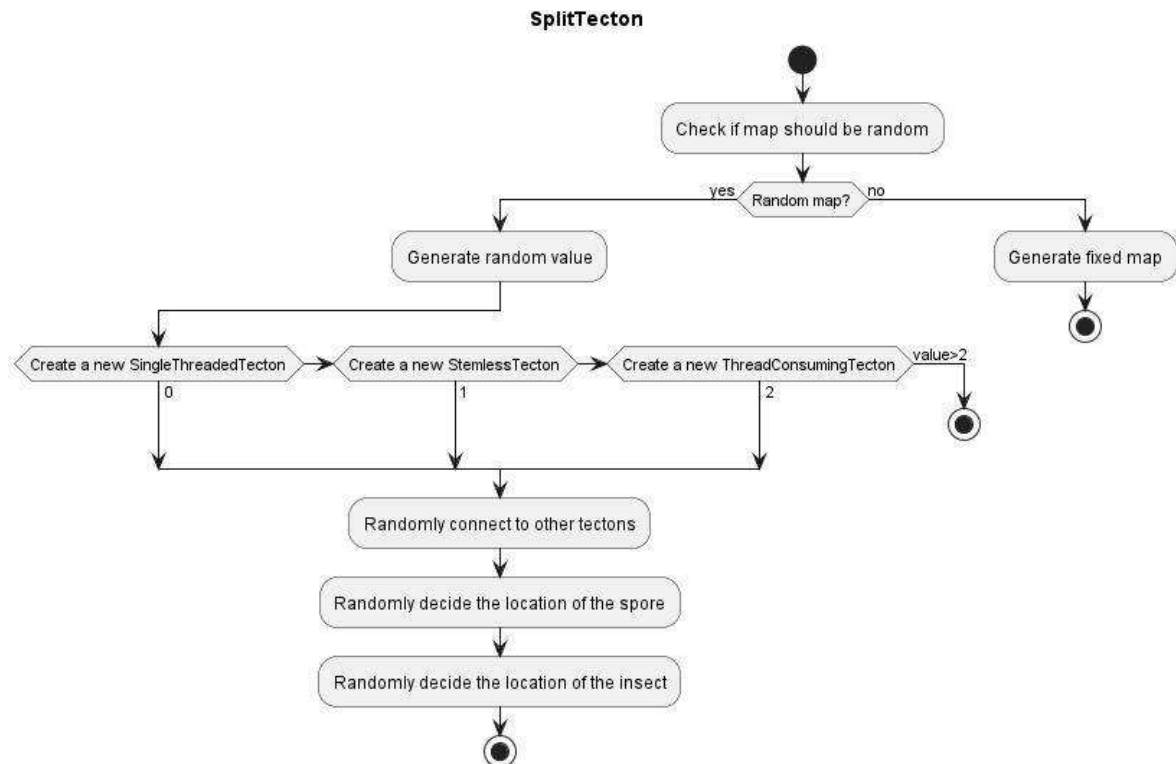
```

insectsToMove = empty list
for each insect in currentTecton.insects:
    if random true or false:
        insectsToMove.add(insect)

for each insect in insectsToMove:
    insect.setLocation(newTecton)

return newTecton

```



- **public bool add(e: Entity):** Hozzáadja az entitást a megfelelő helyre.
- **public bool remove(e: Entity):** eltávolítja az entitást a tektonról.
- **public List<Entity> get():** visszadja a megfelelő változó értékét.
- **public void removeUnconnectedThreads():** Csökkenti a gombatesthez nem kapcsolódó gombafonalak hátrlévő idejét.

### 8.2.24 LifeSupportTecton

- **Felelősség**

Olyan tekton, ami életben tartja a rajta lévő fonalakat, ha azok nem csatlakoznak gombatesthez.

- **Ősosztályok**

Identifiable -> Tecton

- **Interfészek**

- **Attribútumok**

- **Metódusok**

- **public void removeUnconnectedThreads():** Nem csinál semmit.

### 8.2.25 SingleThreadedTecton

- **Felelősség**

Olyan tekton, amin legfeljebb 1 gombafonal lehet.

- **Ősosztályok**

Identifiable -> Tecton

- **Interfészek**

- **Attribútumok**

- **Metódusok**

- **public bool add(msThread: MushroomThread):** Rárakja a fonalat a tektonra, és figyel rá, hogy legfeljebb 1 lehessen.

### 8.2.26 StemlessTecton

- **Felelősség**

Olyan tekton, amin nem nőhet gombatest.

- **Ősosztályok**

Identifiable -> Tecton

- **Interfészek**

- **Attribútumok**

- **Metódusok**

- **public bool add(stem: MushroomStem):** Hamissal visszatér

### 8.2.27 ThreadConsumingTecton

- **Felelősség**

Olyan tekton, amin folyamatosan halnak meg a gombafonalak.

- **Ősosztályok**

Identifiable -> Tecton

- **Interfészek**

- **Attribútumok**

- **Metódusok**

- **public void endRound():** Minden fonal életét csökkenti.

### 8.2.28 Map

- **Felelősség**

Maga a pálya, tárolja a tektonokat.

- **Ősosztályok**

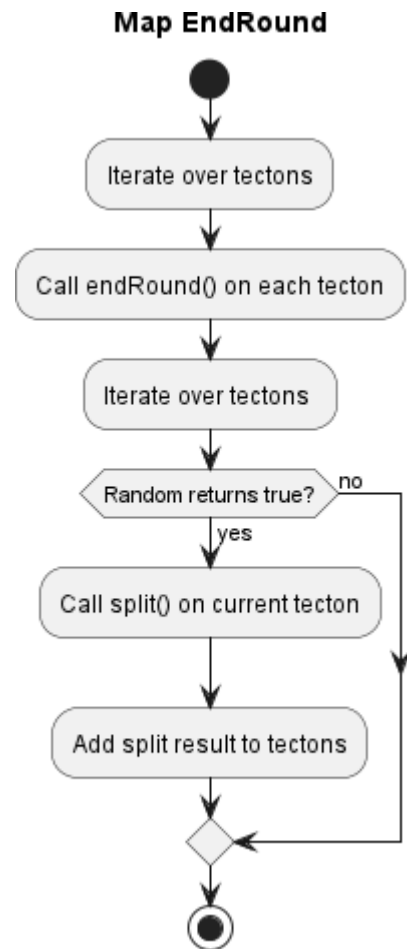
- **Interfészek**

IRound

- **Attribútumok**

- **List<Tecton> tectons:** Tárolja a pályát alkotó tektonokat.

- **Metódusok**



- **public void endRound():** Minden kör végén lefut.
- **public void generate():** Generál egy pályát.
- **public void connect(t1: Tecton, t2: Tecton):** Összeköti a két tektont (Beállítja őket egymás szomszédjának).

## 8.3 A tesztek részletes tervei, leírásuk a teszt nyelvén

### 8.3.1 Teszt: rovarok mozgása

- **Leírás**

Rovar mozgásának tesztelése bizonyos körülmények között

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Azt teszteljük, hogy a rovar mozgatása helyesen működik-e például effektek hatása alatt is.

- **Bemenet**

*/rand -disable*

*# tesztpálya betöltése*

*# több insect van a pályán, egyesek effektek hatásai alatt*

*# mindegyik a 0-ás tectonon van alaphól, innen az 1-esre tudnak mozogni, a 4-es pedig túl távoli*

*/load testmap.map*

*/start*

*# sikeres mozgás*

*/manualtrigger !move 0 1*

*# sikertelen mozgás (elérhetetlen tecton)*

*/manualtrigger !move 0 4*

*# mozgás paralyze effekt alatt (1-es insect)*

*/manualtrigger !move 1 1*

*# mozgás slow effekt alatt (2-es insect)*

*/manualtrigger !move 2 1*

*# mozgás speed effekt alatt (3-es insect)*

*/manualtrigger !move 3 4*

- **Elvárt kimenet**

*Insect(0): location: Tecton(0) -> Tecton(1)*

*Insect(0): location: Tecton(1) -> Tecton(1)*

*Insect(1): location: Tecton(0) -> Tecton(0)*

*Insect(2): location: Tecton(0) -> Tecton(0)*

*Insect(3): location: Tecton(0) -> Tecton(4)*

### 8.3.2 Teszt: fonal elvágása

- **Leírás**

A fonalak rovarok általi elvágását teszteljük

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Azt teszteljük, hogy a fonalvágás spórák hatása alatt is helyesen viselkedik-e, illetve hogy az elszeparált fonalak tényleg megszűnnek-e néhány körön belül.

- **Bemenet**

*/rand -disable*

*# tesztpálya betöltése*

*# több insect van a pályán, egyesek effektek hatásai alatt*

*# mindegyik a 0-ás tectonon van alaptól, innen az 1-esre tudnak mozogni, a 4-es pedig túl távoli*

*/load testmap.map*

*/start*

*# sikertelen vágás, mivel ez az insect spóra hatása miatt nem vághat*

*/manualtrigger !cut 4 0*

*# sikeres vágás*

*/manualtrigger !cut 0 0*

*!endturn*

*!endturn*

*!endturn*

*!endturn*

*...*

*# néhány körön belül meg kell szűnnie a fonálnak.*

*!endturn*

*....*

*# néhány újabb körön belül az 1-es ID-vel rendelkező elszeparált thread is megszűnik.*

- **Elvárt kimenet**

Thread(0): cutoff: false -> false

Thread(0): cutoff: false -> true

Thread(0): cutoffDuration: 0 -> 1

Turn over: next player: Player(1)

Thread(0): cutoffDuration: 1 -> 2

Turn over: next player: Player(2)

Thread(0): cutoffDuration: 2 -> 3

Turn over: next player: Player(3)

Thread(0): cutoffDuration: 3 -> 4

Turn over: next player: Player(4)

*....*

Thread(0): [removed]

Turn over: next player: Player(n)

*....*

Thread(1): [removed]

Turn over: next player: Player(n+k)

### 8.3.3 Teszt: spóra megevése

- **Leírás**

A spórák megevését teszteli.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Azt teszteljük, hogy a rovarok képesek-e megenni a spórákat, illetve hogy ekkor a megfelelő effektek érvénybe lépnek-e rajtuk.

- **Bemenet**

*/rand -disable*

*# tesztpálya betöltése*

*# több insect van a pályán, egyesek effektek hatásai alatt*

*# mindegyik a 0-ás tectonon van alaptól, innen az 1-esre tudnak mozogni, a 4-es pedig túl távoli*

*/load testmap.map*

*/start*

*# sikertelen evés, mivel nem egy tektonon vannak.*

*/manualtrigger !eat 0 0*

*# sikeres evések, effektek kipróbálása*

*/manualtrigger !eat 0 1*

*/manualtrigger !eat 0 2*

*/manualtrigger !eat 0 3*

*/manualtrigger !eat 0 4*

- **Elvárt kimenet**

*# az első sornak nincs hatása*

*Insect(0): speedModifier: 0 -> 1*

*Spore(1): [removed]*

*Insect(0): paralyzed: false -> true*

*Spore(2): [removed]*

*Insect(0): clawParalyzed: false -> true*

*Spore(3): [removed]*

*Insect(0): speedModifier: 1 -> 0*

*Spore(4): [removed]*

### 8.3.4 Teszt: gombafonal növesztése

- **Leírás**

A gombafonalak növesztését teszteli.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Azt teszteljük, hogy a fonalak növesztésére vonatkozó szabályok fennállnak-e.

- **Bemenet**

*/rand -disable*

*# tesztpálya betöltése*

*# több insect van a pályán, egyesek effektek hatásai alatt*

*# mindegyik a 0-ás tectonon van alaphól, innen az 1-esre tudnak mozogni, a 4-es pedig túl távoli*

*/load testmap.map*

*/start*

*# ide nem szabadna fonalat növeszteni, mert itt már van*

*!grow 0*

*# itt nincs, ide tehát szabad*

*!grow 4*

- **Elvárt kimenet**

*# az első sornak nincs hatása*

Thread(n): [created]

Tecton(4): threads: Thread(n) added



### 8.3.5 Teszt: gombatest növesztése

- **Leírás**

A gombatestek növesztését teszteli.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Azt teszteljük, hogy a gombatestek növesztésére vonatkozó szabályok fennállnak-e.

- **Bemenet**

*/rand -disable*

*# tesztpálya betöltése*

*# több insect van a pályán, egyesek effektek hatásai alatt*

*# mindegyik a 0-ás tectonon van alaphól, innen az 1-esre tudnak mozogni, a 4-es pedig túl távoli*

*/load testmap.map*

*/start*

*# ide szabad növeszteni, és van is itt elegendő spóra hozzá (illetve fonal)*

*/manualtrigger !plant 0*

*# ide nem tudunk növeszteni, nincs hozzá spóra*

*/manualtrigger !plant 1*

*# itt fonal nincs hozzá*

*/manualtrigger !plant 4*

*# ide tudunk növeszteni, mivel a fonal evett meg korábban rovar*

*/manualtrigger !plant 2*

- **Elvárt kimenet**

MushroomStem(0): [created]

Tecton(0): stems: MushroomStem(0) added

# a második sornak nincs hatása

# a harmadik sornak sincs

MushroomStem(1): [created]

Tecton(2): stems: MushroomStem(1) added

Thread(n): eaten: true -> false

### 8.3.6 Teszt: spórák dobása

- **Leírás**

A spórák dobását teszteli

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Azt teszteljük, hogy a spórák dobására vonatkozó szabályok fennállnak-e.

- **Bemenet**

*/rand -disable*

*# tesztpálya betöltése*

*# több insect van a pályán, egyesek effektek hatásai alatt*

*# mindegyik a 0-ás tectonon van alaphól, innen az 1-esre tudnak mozogni, a 4-es pedig túl távoli*

*/load testmap.map*

*/start*

*# sikeres dobás tesztelése*

*/manualtrigger !throw 0 0*

*# sikertelen dobás, túl távoli a tecton*

*/manualtrigger !throw 0 4*

*# fejlettebb gombatest viszont tud messzebbre dobni*

*/manualtrigger !throw 1 4*

- **Elvárt kimenet**

Spore(n): [created]

MushroomStem(0): numThrownSpores: 0 -> 1

# a második sornak nincs hatása

Spore(n+1): [created]

MushroomStem(1): numThrownSpores: 0 -> 1

### 8.3.7 Teszt: kör és round vége

- **Leírás**

A körök/roundok végét teszteli

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Megnézzük hogy a játékosok megfelelő sorrendben következnek-e egymás után, illetve hogy a round vége is megtörténik-e.

- **Bemenet**

*/rand -disable*

*# tesztpálya betöltése*

*# több insect van a pályán, egyesek effektek hatásai alatt*

*# mindegyik a 0-ás tectonon van alaptól, innen az 1-esre tudnak mozogni, a 4-es pedig túl távoli*

*/load testmap.map*

*/start*

*!endturn*

*!endturn*

*!endturn*

*!endturn*

- **Elvárt kimenet**

Turn over: next player: Player(1)

Turn over: next player: Player(2)

Turn over: next player: Player(3)

Round over

Turn over: next player: Player(0)

### **8.4 A tesztelést támogató programok tervei**

A tesztelést a dinamikus JUnit tesztekkel valósítjuk meg. Mindegyik teszteset egy külön mappában helyezkedik el a Tests mappában. A mappa neve fogja azonosítani a tesztet. Egy tesztnek tartalmaznia kell az alábbiakat:

- **input.txt:** parancssori utasításokat tartalmaz, minden sorban pontosan egyet. A teszt beolvassa a tartalmát, majd egyesével végrehajtja a benne talált parancsokat. A legelsőnek a pályát kell betöltenie.
- **output.txt:** az utasítások végrehajtásakor a program log-olja, hogy mi történik, ezek írja ki ebbe a fájlba.
- **expected.txt:** a teszt várt eredménye.

Ha az output.txt megegyezik az expected.txt-vel, akkor sikeres a teszt.

A szükséges pályák mindig tesztfuttatáskor le lesznek generálva.

## 8.5 Napló

Kezdet	Időtartam	Résztevők	Leírás
2025.04.02.	1.5 óra	Bohus Dúcz Galambos Horváth Végyári	Tevékenység: megbeszélés, előző rész hibáinak kijavítása, feladatok felosztása
2025.04.07 14:00	1.5 óra	Végyári	Tevékenység: Tesztelést támogató program írása
2025.04.09 17:00	1.5 óra	Bohus	Tevékenység: Osztályok írása
2025.04.09 17:00	1.5 óra	Horváth	Tevékenység: Osztályok írása
2025.04.10 8:00	6 óra	Horváth	Tevékenység: Aktivitás diagramok, state chartok
2025.04.10 20:30	0,5 óra	Galambos	Tevékenység: Osztályok írása
2025.04.11 12:00	1.5 óra	Bohus	Tevékenység: Osztályok írása, activity diagramok
2025.04.11 19:00	1.5 óra	Végyári	Tevékenység: Osztályok írása
2025.04.11 20:00	2 óra	Dúcz	Tevékenység: Tesztesetek elkészítése
2025.04.11 20:00	1 óra	Végyári	Tevékenység: Osztályok írása
2025.04.13 02:00	2 óra	Galambos	Tevékenység: Osztályok írása, dokumentum átnézése
2025.04.13 12:00	1.5 óra	Végyári	Tevékenység: Activity diagramok írása
2025.04.13 13:00	0.5 óra	Bohus	Tevékenység: pszeduo kód, dokumentum rendezése
2025.04.13 13:30	0.5 óra	Végyári	Tevékenység: Osztályok írása

# 10. Prototípus elkészítése

23 – totoro

Konzulens:

Haragos Gergő Viktor

## Csapattagok

Bohus János  
Dúcz Ákos  
Galambos Csaba  
Horváth Avarka  
Végvári Péter

XYOVFZ	janos.bohus1@gmail.com
GC1RTE	akos.ducz@gmail.com
BVBQKN	galambos.csaba05@gmail.com
OODMXV	avarkahorvath@gmail.com
AW3JH1	vegvaripeter21@gmail.com

## 10. Prototípus beadása

### 10.1 Fordítási és futtatási útmutató

#### 10.1.1 Fájllista

Fájl neve	Méret	Keletkezés ideje	Tartalom
Main.java	751	2025-04-24 11:26:36	Beolvasás itt történik, main függvény
controller\Game.java	11208	2025-04-24 11:26:36	játék osztály
controller\Parser.java	13722	2025-04-24 11:26:36	beolvasó osztály, ez hívja meg a függvényeket
model\Map.java	4324	2025-04-24 11:26:36	térkép osztály, itt van az előre generált térkép is
model\core\Entity.java	3239	2025-04-24 11:26:36	őszosztály ebből származik le az összes entitás
model\core\Identifiable.java	2346	2025-04-24 11:26:36	Absztrakt osztály, amely egyedi azonosítóval lát el minden leszármazott objektumot.
model\core\IRound.java	237	2025-04-24 11:26:36	Interfész, olyan osztályok implementálják, akik minden kör után csinálnak valamit.
model\core\ITurn.java	246	2025-04-24 11:26:36	Intefész, olyan osztályok implementálják, amik egy lépés bejezésekor csinálnak valamit.
model\core\Player.java	1890	2025-04-24 11:26:36	Absztrakt osztály, amely egy játékost definiál
model\effect\ClawParalyzeEffect.java	325	2025-04-24 11:26:36	csáprágó kábító effect osztálya
model\effect\Effect.java	1371	2025-04-24 11:26:36	effect őszosztály
model\effect\FastEffect.java	312	2025-04-24 11:26:36	gyorsító effect osztálya
model\effect\ParalyzeEffect.java	415	2025-04-24 11:26:36	kábító effect osztálya
model\effect\SlowEffect.java	314	2025-04-24 11:26:36	lassító effect osztálya
model\effect\SplitEffect.java	260	2025-04-24 11:26:36	osztódó effect osztálya
model\insect\IInsect.java	392	2025-04-24 11:26:36	interface az insect osztályoknak
model\insect\Insect.java	9337	2025-04-24 11:26:36	rovar osztály
model\insect\Insecter.java	2494	2025-04-24 11:26:36	rovarász játékos osztálya
model\mushroom\IStem.java	1325	2025-04-24 11:26:36	interface a gombáknak
model\mushroom\IThread.java	1364	2025-04-24 11:26:36	interface a gomafonalaknak
model\mushroom\Mushroomer.java	12036	2025-04-24 11:26:36	gombász játékos osztálya

model\mushroom\MushroomStem.java	5201	2025-04-24 11:26:36	gombatest osztálya
model\mushroom\MushroomThread.java	5644	2025-04-24 11:26:36	gombafonal osztálya
model\mushroom\spore\ClawParalyzingSpore.java	930	2025-04-24 11:26:36	csáprágó spóra osztálya
model\mushroom\spore\ISpore.java	920	2025-04-24 11:26:36	spókák interface-e
model\mushroom\spore\ParalyzingSpore.java	558	2025-04-24 11:26:36	kábító spóra osztálya
model\mushroom\spore\SlowingSpore.java	549	2025-04-24 11:26:36	lassító spóra osztálya
model\mushroom\spore\SpeedingSpore.java	554	2025-04-24 11:26:36	gyorsító spóra osztálya
model\mushroom\spore\SplitterSpore.java	555	2025-04-24 11:26:36	osztódó spóra osztálya
model\mushroom\spore\Spore.java	2674	2025-04-24 11:26:36	spóra ösosztály
model\tecton\LifeSupportTecton.java	275	2025-04-24 11:26:36	Olyan tekton fajta, amin nem pusztulnak el a gombafonalak, ha nincsenek a gombafonalak összekötve gombatesttel.
model\tecton\SingleThreadedTecton.java	608	2025-04-24 11:26:36	Olyan tekton fajta, amin csak egy gombafonal lehet.
model\tecton\StemlessTecton.java	442	2025-04-24 11:26:36	Olyan tekton fajta, amin nem lehet gombatest.
model\tecton\Tecton.java	11528	2025-04-24 11:26:36	Tekton osztály, tárolja a hozzá tartozó gombatest, fonalakat, spórákat, rovarokat és szomszédos tektonokat.
model\tecton\ThreadConsumingTecton.java	305	2025-04-24 11:26:36	Olyan tekton fajta, amin minden kör végén eltűnnek a gombafonalak.
test\controller\GameTest.java	1569	2025-04-24 13:43:57	Teszt osztály, csak a helyes működés ellenőrzésére van.
test\model\InsectTest\InsectTest.java	2409	2025-04-24 11:26:36	Teszt osztály, csak a helyes működés ellenőrzésére van.
test\model\mushroom\MushroomerTest.java	7952	2025-04-24 11:26:36	Teszt osztály, csak a helyes működés ellenőrzésére van.
test\model\mushroom\MushroomStemTest.java	3819	2025-04-24 11:26:36	Teszt osztály, csak a helyes működés ellenőrzésére van.
test\model\mushroom\MushroomThreadTest.java	3850	2025-04-24 11:26:36	Teszt osztály, csak a helyes működés ellenőrzésére van.
test\model\tecton\LifeSupportTectonTest.java	816	2025-04-24 11:26:36	Teszt osztály, csak a helyes működés ellenőrzésére van.



test\model\tecton\SingleThreadedTectonTest.java	791	2025-04-24 11:26:36	Teszt osztály, csak a helyes működés ellenőrzésére van.
test\model\tecton\StemlessTectonTest.java	624	2025-04-24 11:26:36	Teszt osztály, csak a helyes működés ellenőrzésére van.
test\model\tecton\TectonTest.java	5163	2025-04-24 11:26:36	Teszt osztály, csak a helyes működés ellenőrzésére van.
test\model\tecton\ThreadConsumingTest.java	792	2025-04-24 11:26:36	Teszt osztály, csak a helyes működés ellenőrzésére van.
cuttest.txt	1KB	2025-04-24 11:26:36	Cut teszt
growtest.txt	1KB	2025-04-24 11:26:36	Grow teszt
movetest.txt	1KB	2025-04-24 11:26:36	Move teszt
planttest.txt	1KB	2025-04-24 11:26:36	Plant teszt
roundtest.txt	1KB	2025-04-24 11:26:36	Round teszt
sporetest.txt	1KB	2025-04-24 11:26:36	Spore teszt
throwtest.txt	1KB	2025-04-24 11:26:36	Throw teszt

### 10.1.2 Fordítás

A program fordítása a Visual Studio Code-ban, Extension Pack for Java kiegészítővel történik, automatikusan futtatáskor. A fordításhoz OpenJDK 17 vagy annál újabb szükséges.

### 10.1.3 Futtatás

A futtatás szintén VSCode-ban zajlik, felül a Run->Start Debugging(F5) gombbal lehet elindítani a programot, ami az integrált terminálba fog futni, ahol majd a bementét is várja.

## 10.2 Tesztek jegyzőkönyvei

### 10.2.1 CutTest

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 27.

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 25.
<b>Teszt eredménye</b>	/list parancs hatására pálya betöltésekor egy helyett két gombatest jött létre játékosonként.
<b>Lehetséges hibaok</b>	A kód hibásan kétszer is létrehoz a játékoshoz tartozó gombatestet.
<b>Változtatások</b>	Hibás kód javítása a mushroomer konstruktorában.

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 25.

<b>Teszt eredménye</b>	A játékos képes volt olyan fonalat is elvágni, amely tektonon nem tartózkodott.
<b>Lehetséges hibaok</b>	A kód nem ellenőrzi hogy milyen távol van az adott fonal.
<b>Változtatások</b>	Hibás kód javítása.

### 10.2.2 GrowTest

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 27.

### 10.2.3 MoveTest

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 27.

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 24.
<b>Teszt eredménye</b>	Nemlétező Insect ID megkapásakor a !move parancs lefagyott.
<b>Lehetséges hibaok</b>	A kód nem kezeli a hibás bemenetet.
<b>Változtatások</b>	Try-catch blokk hozzáadása a parancsértelmezőhöz.

### 10.2.4 PlantTest

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 27.

### 10.2.5 RoundTest

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 27.

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 26.
<b>Teszt eredménye</b>	A round-ok vége nem futott le. Ennek javítása után egyes teszteseteket a tektonok több részre szakadása megbonyolított, vagy elrontott.
<b>Lehetséges hibaok</b>	Hiányzó függvényhívás, illetve ebből fakadóan rosszul designolt tesztek.
<b>Változtatások</b>	A hiányzó kód kiegészítése, illetve opcionálisan kikapcsolható tektonszakadás a /split paranccsal.

### 10.2.6 SporeTest

<b>Tesztelő neve</b>	Dúcz Ákos
----------------------	-----------

<b>Teszt időpontja</b>	2025. 04. 27.
------------------------	---------------

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 25.
<b>Teszt eredménye</b>	Insect splitter spóra esetén nem jelenik meg az új insect.
<b>Lehetséges hibaok</b>	Hiányzó kódrészlet.
<b>Változtatások</b>	Hiányzó kódrészlet hozzáadva.

### 10.2.7 ThrowTest

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 27.

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 25.
<b>Teszt eredménye</b>	A játékos csak egyszer tudott spórát dobni egy játékon belül.
<b>Lehetséges hibaok</b>	Kör vége fv. nincs meghívva a mushroomstem esetében.
<b>Változtatások</b>	Hiányzó kódrészlet hozzáadva.

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 25.
<b>Teszt eredménye</b>	A !levelup függvény esetén nemlétező parancs hibát dob a játék.
<b>Lehetséges hibaok</b>	Elfelejtettük implementálni a parancsot.
<b>Változtatások</b>	Hiányzó parancs implementálva.

<b>Tesztelő neve</b>	Dúcz Ákos
<b>Teszt időpontja</b>	2025. 04. 26.
<b>Teszt eredménye</b>	A mushroomstem tetszőleges távolságra képes spórát dobni szinttől függetlenül.
<b>Lehetséges hibaok</b>	Hiányzó ellenőrző kód.
<b>Változtatások</b>	Hiányzó kódrészlet hozzáadva.

## 10.3Értékelés

Tag neve	Tag neptun	Munka százalékban
Bohus János	XYOVFZ	20%
Dúcz Ákos	GC1RTE	20%
Galambos Csaba	BVBQKN	20%
Horváth Avarka	OODMXV	20%
Végvári Péter	AW3JH1	20%

**10.4 Napló**

Kezdet	Időtartam	Résztevők	Leírás
2025.04.16 12:00	1,5 óra	Bohus Dúcz Galambos Horváth Végvári	Megbeszélés: feladatok szétosztása
2025.04.16-27.	15 óra	Végvári	Tevékenységek: skeleton átírása prototyba, Game osztály, Tecton tesztek
2025.04.19 9:00	4 óra	Horváth	Tevékenység: gomba osztályok megírása
2025.04.22 19:00	1 óra	Galambos	Tevékenység: Parser osztály írása
2025.04.23 12:00	4 óra	Galambos	Tevékenység: Parser osztály írása
2025.04.24 9:00	6 óra	Horváth	Tevékenység: unit tesztek a gombákhoz, javítás
2025.04.24 10:00	5 óra	Bohus	Tevékenység: Insecthez kötődő osztályok írása tesztelése
2025.04.24 16:00	3 óra	Dúcz	Tevékenység: Tesztelés támogatásának implementálása, tesztek írása
2025.04.25 10:00	4 óra	Dúcz	Tevékenység: Tesztelés, hibák javítása
2025.04.26	2 óra	Dúcz	Tevékenység: Tesztek írása, hibák javítása

# 11. Grafikus felület specifikációja

23 – totoro


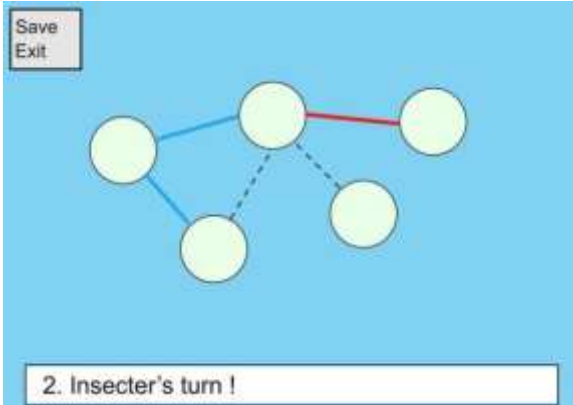
Konzulens:

Haragos Gergő Viktor

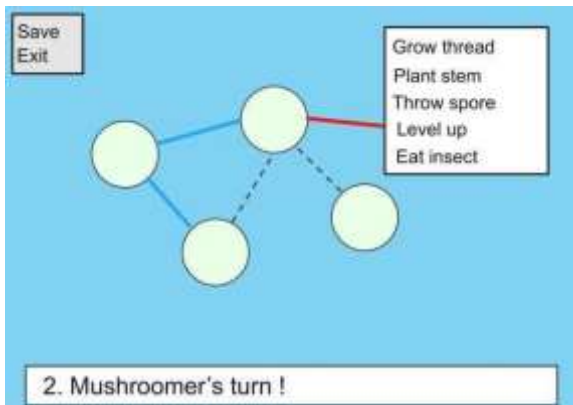
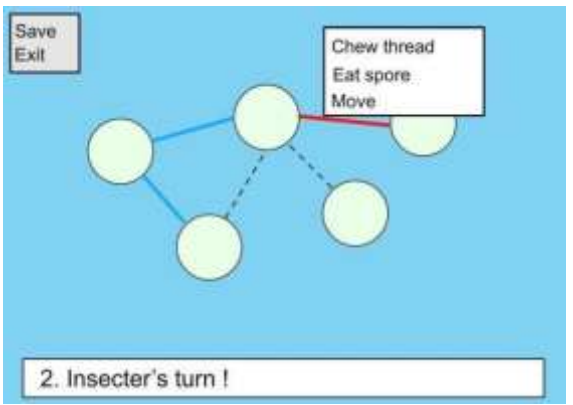
## Csapattagok

Bohus János	XYOVFZ	janos.bohus1@gmail.com
Dúcz Ákos	GC1RTE	akos.ducz@gmail.com
Galambos Csaba	BVBQKN	galambos.csaba05@gmail.com
Horváth Avarka	OODMXV	avarkahorvath@gmail.com
Végvári Péter	AW3JH1	vegvaripeter21@gmail.com

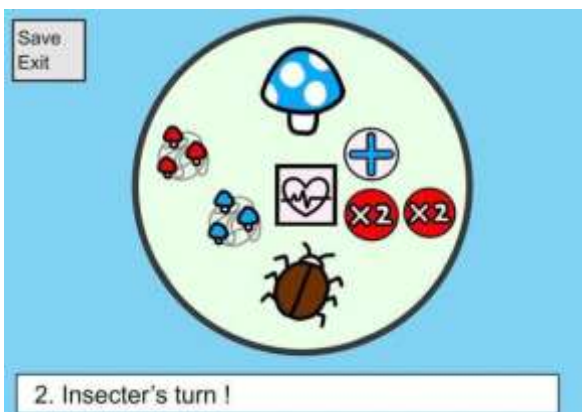
## 11.1 A grafikus interfész

Main menu	Játék közben:
	

Jobb klikk után:









Gombász	Rovarász
	

Tekton közelről:











Használt ikonok:

Rovar:










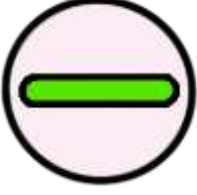

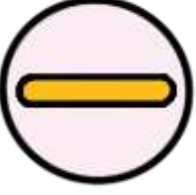








	Insecter 1	Insecter 2	Insecter 3	Insecter 4
Normal/Other effect				
Paralyzed				

Gomba:

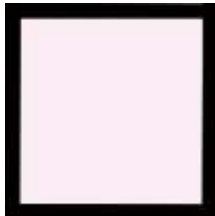




	Mushroomer 1	Mushroomer 2	Mushroomer 3	Mushroomer 4
mushroom stem				
mushroom thread				



Spóra:

	Mushroomer 1	Mushroomer 2	Mushroomer 3	Mushroomer 4
claw paralyzing				
paralyzing				
slowing				
speeding				
splitting				

Tekton:

Basic	Life support	Single threaded	Stemless tecton	Thread consuming
				

## 11.2A grafikus rendszer architektúrája

### 11.2.1 A felület működési elve

[Le kell írni, hogy a grafikai megjelenésért felelős osztályok, objektumok hogyan kapcsolódnak a meglevő rendszerhez, a megjelenítés során mi volt az alapelv. Törekedni kell az MVC megvalósításra. Alapelvek lehetnek: **push** alapú: a modell értesíti a felületet, hogy változott; **pull** alapú: a felület kérdezi le a modellt, hogy változott-e; **kevert**: a kettő kombinációja.]

A grafikai modell egy pull-alapú keretrendszert valósít meg. Két rétege létezik, a Window és a View.

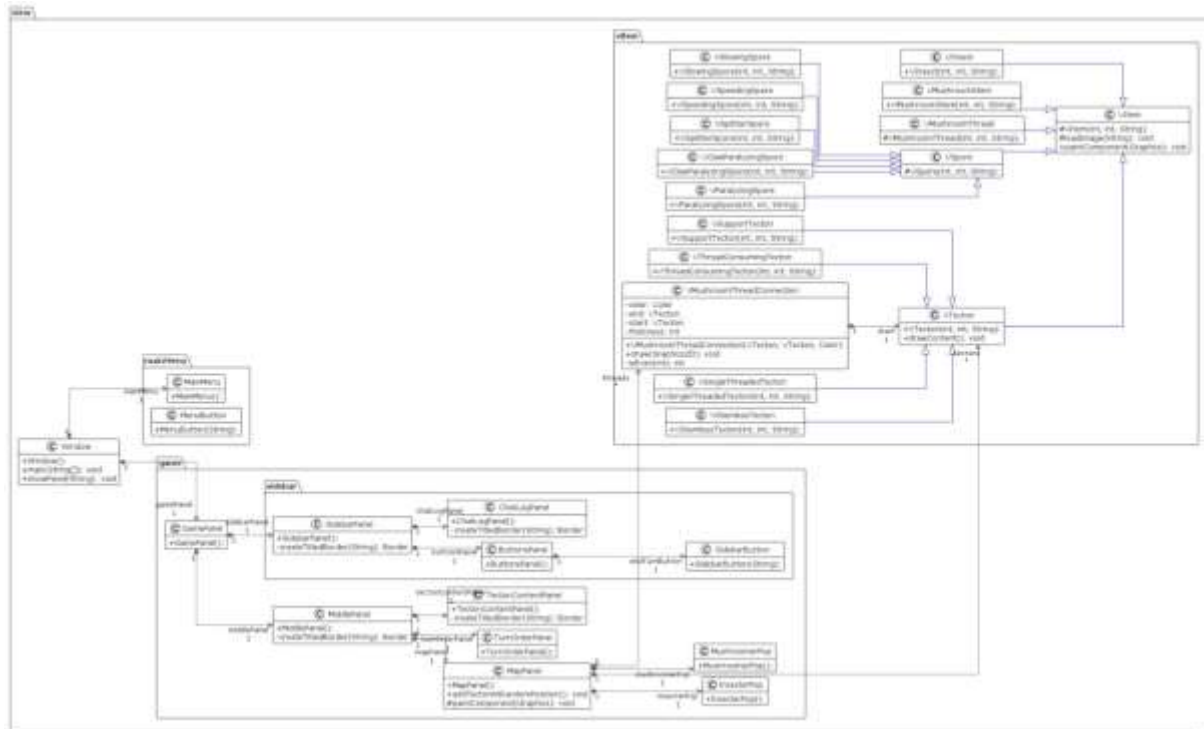
A Window réteg felelős a felhasználói felület megjelenítéséért, a játéktér kirajzolásáért és a játékos által indított események vezérléséért. Ez értesíti a modellt ha a játékos valamilyen akciót hajt végre, majd ezután frissíti a felhasználói felületet (eközben frissítve a View-beli osztályok állapotát is).

Két alapvető GUI mód létezik: egy főmenü illetve egy játéktér, melyek között a Window réteg megfelelően váltogat.

A View réteg a játékelemek megjelenítéséért felelős. Ez kezeli a rovarokhoz, gombokhoz, és egyéb játékelemekhez tartozó ikonokat, és irányítja azok elhelyezkedését a pályán. Egyes View-beli osztályok csupán egy egyszerű ikont rajzolnak ki a játéktér egy megadott pozíciójára (pl.: VInsect), míg mások bonyolult

megjelenítési logikával rendelkeznek (pl.: VMushroomThreadConnection). Az összes modellben szereplő kirajzolandó osztálynak van View-beli megfelelője.

### 11.2.2 A felület osztály-struktúrája



### 11.3A grafikus objektumok felsorolása

[Az új osztályok felsorolása. Az régi osztályok közül azoknak a felsorolása, ahol változás volt. Ezek esetén csak a változásokat kell leírni.]

### 11.3.1 Window

## • Felelősség

A játéklablakot megjelenítő osztály. Magában foglalja a főmenüt és a játékeret megjelenítő paneleket is. A Game objektum mellett ez jön létre elsőként.

## • Össztályok

[Mely osztályokból származik (öröklési hierarchia)]

JFrame ← Window

- **Interfészek**

-

## • Attribútumok

[Milyen attribútumai vannak]

- **CardLayout cardLayout** : A főmenü/játékteret tartalmazó cardlayout.
- **JPanel cardPanel**: A főmenü/játékteret tartalmazó cardpanel.
- **MainMenu mainMenu**: a főmenü kezelő objektum
- **GamePanel gamePanel**: a játékteret kirajzoló objektum
- **Metódusok**
  - **Konstruktor**: létrehozza az ablakot és a megfelelő paneleket
  - **public void showPanel(string name)**: a megfelelő panel megjelenítése név alapján
  - **public static void main()**: létrehozza a Game és Window objektumokat.

### 11.3.2 MainMenu

- **Felelősség**

A főmenü megjelenítéséért felelős.

- **Ősosztályok**

[Mely osztályokból származik (öröklési hierarchia)]

JPanel ← MainMenu

- **Interfészek**

-

- **Attribútumok**

- -

- **Metódusok**

- **Konstruktor**: Létrehozza a megjelenítendő gombokat és beállítja a megjelenítést.

### 11.3.3 ButtonsPanel

- **Felelősség**

A ButtonsPanel osztály felelős három gomb („End Turn”, „Save”, „Exit”) megjelenítéséért és elrendezéséért egy függőleges, szellősen tagolt panelen belül, a felhasználói felület oldalsávjában.

- **Ősosztályok**

JComponent -> JPanel -> ButtonsPanel

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- Nincs saját attribútuma (csak örökölt).

- **Metódusok**

- **ButtonsPanel()**: Konstruktor, létrehozza és elhelyezi a három gombot. [public]

### 11.3.4 ChatLogPanel

- **Felelősség**

A ChatLogPanel egy nem szerkeszthető, görgethető szövegpanelt jelenít meg a csevegési napló számára, a felhasználói felület oldalsávján belül.

- **Össztályok**

JComponent -> JPanel -> ChatLogPanel

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

[Milyen attribútumai vannak]

- **attribútum1**: attribútum jellemzése: mire való, láthatósága (UML jelöléssel), típusa
- **attribútum2**: attribútum jellemzése: mire való, láthatósága (UML jelöléssel), típusa

- **Metódusok**

- chatLog: A megjelenített szövegterület. [– JTextArea]
- chatScrollPane: A görgetést lehetővé tevő komponens. [– JScrollPane]

### 11.3.5 SidebarButton

- **Felelősség**

A SidebarButton osztály egy testreszabott kinézetű, egységes stílusú gombot reprezentál, amelyet a felhasználói felület oldalsávjában használnak.

- **Össztályok**

AbstractButton -> JButton -> SidebarButton

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- Nincs saját attribútuma (csak örökölt).

- **Metódusok**

- SidebarButton(String text): Konstruktor, létrehozza a gombot a megadott szöveggel. [public]
- – initializeButton(): Beállítja a gomb stílusát és tulajdonságait. [private]

### 11.3.6 SidebarPanel

- **Felelősség**

A SidebarPanel osztály összefogja és elrendezi az oldalsáv komponenseit: a gombokat tartalmazó panelt és a csevegési naplót megjelenítő panelt.

- **Össztályok**

JComponent -> JPanel -> SidebarPanel

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**
  - Nincs saját attribútuma (csak örökölt).
- **Metódusok**

[SidebarPanel()]: Konstruktor, létrehozza és elrendezi az oldalsáv részeit. [public]

– createTitledBorder(String title): Segédfüggvény a keret létrehozásához. [private]

### 11.3.7 VClawParalyzingSpore

- **Felelősség**

A VClawParalyzingSpore egy vizuális spóraelem, amelyet a játékban jelenítenek meg, és kinézete a „claw-paralyzing-spore.png” képfájlon alapul.

- **Ősosztályok**

VItem → VSpore → VClawParalyzingSpore

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**
  - imagePath: a spóra képének fájlvonala. [~ String] (örökölt)
- **Metódusok**
  - VClawParalyzingSpore(int x, int y, String tooltipText): Beállítja a pozíciót, tooltipet és a képfájlt. [public]

### 11.3.8 VParalyzingSpore

- **Felelősség**

A VParalyzingSpore egy megbénító hatású vizuális spóra, amelyhez a „images/spore/paralyzing\_spore.png” fájl tartozik.

- **Ősosztályok**

VItem → VSpore → VParalyzingSpore

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**
  - imagePath: a spóra képének fájlvonala. [~ String] (örökölt)
- **Metódusok**
  - VParalyzingSpore(int x, int y, String tooltipText): Inicializálja a spórát pozícióval és képpel. [public]

### 11.3.9 VSlowingSpore

- **Felelősség**

A VSlowingSpore egy lassító hatású vizuális spóraelem, mely a „spore/slowing\_spore.png” képfájllal jelenik meg.

- **Ősosztályok**

VItem → VSpore → VSlowingSpore

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- imagePath: a spóra képének fájlvonala. [~ String] (örökölt)

- **Metódusok**

- VSlowingSpore(int x, int y, String toolTipText): Létrehozza a lassító spórát a megadott paraméterekkel. [public]

### 11.3.10 VSpeedingSpore

- **Felelősség**

A VSpeedingSpore egy gyorsító hatású spóra megjelenítésére szolgál, képi reprezentációja a „speeding-spore.png” fájl.

- **Ősosztályok**

VItem → VSpore → VSpeedingSpore

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- imagePath: a spóra képének fájlvonala. [~ String] (örökölt)

- **Metódusok**

- VSpeedingSpore(int x, int y, String toolTipText): Létrehozza a gyorsító spórát. [public]

### 11.3.11 VSplitterSpore

- **Felelősség**

A VSplitterSpore a játékban megjelenített, osztódó spóra, amely a „images/spore/splitter\_spore.png” képfájlt használja.

- **Ősosztályok**

VItem → VSpore → VSplitterSpore

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- imagePath: a spóra képének fájlvonala. [~ String] (örökölt)

- **Metódusok**

- VSplitterSpore(int x, int y, String tooltipText): Konstruktor, beállítja az osztódó spóra megjelenését. [public]

### 11.3.12 VSpore

- **Felelősség**

A VSpore egy absztrakt osztály, amely a játékban megjelenő összes spóra közös vizuális alapjául szolgál.

- **Össztályok**

VItem → VSpore

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- Nincs saját attribútuma (csak örökölt attribútumokat használ, például imagePath).

- **Metódusok**

- VSpore(int x, int y, String tooltipText): Inicializálja a spóra pozícióját és tooltipjét. [protected]

### 11.3.13 VTecton

- **Felelősség**

A VTecton osztály a játékban megjelenő, vizuális gombaszerű entitást reprezentál, amelyhez szálak, spórák, szár és rovarok kapcsolódhatnak. Grafikai megjelenítését is kezeli.

- **Össztályok**

VItem → VTecton

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- diameter: Az entitás átmérője. [int]
- x, y: Az entitás pozíciója. [int]
- color: A kirajzolás színe. [Color]
- threads: A csatlakoztatott szálak listája. [List<VMushroomThreadConnection>]
- insects: A kapcsolódó rovarok listája. [List<VInsect>]
- spores: A kapcsolódó spórák listája. [List<VSpore>]
- stem: A kapcsolódó gombaszár. [VMushroomStem]

- **Metódusok**



- VTecton(int x, int y, String toolTipText): Konstruktor, inicializálja a pozíciót, tooltipet és képet. [public]
- addThread(VMushroomThreadConnection thread): Szál hozzáadása a listához. [public]
- addInsect(VInsect insect): Rovar hozzáadása. [public]
- addSpore(VSpore spore): Spóra hozzáadása. [public]
- setStem(VMushroomStem stem): Szár beállítása. [public]
- draw(Graphics2D g2d): Kirajzolja a VTectont. [public]
- drawContent(): Jelenleg üres, de tartalom rajzolására szolgálhat. [public]
- contains(Point p): Megadja, hogy egy pont az entitás körébe esik-e. [public]
- getX()/setX(int x), getY()/setY(int y): Pozíció elérők/beállítók. [public]
- getThreads(): A szálak számát adja vissza. [public]

### 11.3.14 VSingleThreadedTecton

#### • Felelősség

A VSingleThreadedTecton egy olyan VTecton, amely a "single\_threaded.png" képpel jelenik meg, vizuálisan jelezve a típust.

#### • Ősosztályok

VItem → VTecton → VSingleThreadedTecton

#### • Interfészek

Nem valósít meg külön interfészt.

#### • Attribútumok

- imagePath: A gomba típusát jelző kép útvonala. [~ String] (örökölt)

#### • Metódusok

- VSingleThreadedTecton(int x, int y, String toolTipText): Konstruktor, beállítja a képet és pozíciót. [public]

### 11.3.15 VStemlessTecton

#### • Felelősség

A VStemlessTecton egy szár nélküli gombát reprezentáló VTecton, a „stemless.png” kép alapján.

#### • Ősosztályok

VItem → VTecton → VStemlessTecton

#### • Interfészek

Nem valósít meg külön interfészt.

#### • Attribútumok

- imagePath: A reprezentációs kép útvonala. [~ String]

#### • Metódusok

- `VStemlessTecton(int x, int y, String toolTipText)`: Konstruktor, a kép és a pozíció inicializálására. [public]

### 11.3.16 VSupportTecton

- **Felelősség**

A `VSupportTecton` egy támogató gombatípus, amely megjelenéséhez a „support.png” képfájlt használja.

- **Ősosztályok**

`VItem` → `VTecton` → `VSupportTecton`

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- `imagePath`: A támogató típushoz tartozó kép elérési útja. [~ String]

- **Metódusok**

- `VSupportTecton(int x, int y, String toolTipText)`: Példányosítja a támogató gombát. [public]

### 11.3.17 VThreadConsumingTecton

- **Felelősség**

A `VThreadConsumingTecton` olyan `VTecton`, amely szálat „fogyaszt”, és ezt vizuálisan a „thread\_consuming.png” kép jeleníti meg.

- **Ősosztályok**

`VItem` → `VTecton` → `VThreadConsumingTecton`

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- `imagePath`: A kép útvonala, amely ezt a típust reprezentálja. [~ String]

- **Metódusok**

- `VThreadConsumingTecton(int x, int y, String toolTipText)`: Konstruktor, amely betölti a megfelelő képet. [public]

### 11.3.18 VItem

- **Felelősség**

A `VItem` egy absztrakt osztály, amely a játékban megjelenő grafikus elemek (pl. rovar, szál, szár, spóra stb.) közös ősosztálya. Kezeli a pozíciót, a tooltip szöveget és a kép betöltését, valamint a komponens kirajzolását.

- **Ősosztályok**

`JPanel` → `VItem`

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- image: Az elem megjelenítéséhez használt kép. [Image]
- imagePath: A kép elérési útja. [String]
- x, y: Az elem pozíciója. [int]
- tooltipText: Tooltip szöveg. [String]

- **Metódusok**

- VItem(int x, int y, String tooltipText): Inicializálja a pozíciót és a tooltipet. [protected]
- loadImage(): Betölti a képet az imagePath alapján. [protected]
- paintComponent(Graphics g): Kirajzolja az elemet a képpel. [protected override]
- 

### 11.3.19 VInsect

- **Felelősség**

A VInsect egy rovar grafikus reprezentációja a játékban.

- **Ősosztályok**

JPanel → VItem → VInsect

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- imagePath: "images/insect.png" – A rovar képe. [örökölt]

- **Metódusok**

- VInsect(int x, int y, String tooltipText): Létrehozza a rovar a megadott pozícióval és tooltip szöveggel. [public]

### 11.3.20 VMushroomStem

- **Felelősség**

A VMushroomStem egy gombaszár vizuális reprezentációja.

- **Ősosztályok**

JPanel → VItem → VMushroomStem

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- imagePath: "images/mushroomStem.png" – A szár képe.

- **Metódusok**
- VMushroomStem(int x, int y, String toolTipText): Konstruktor, amely beállítja a pozíciót és betölti a képet. [public]

### 11.3.21 VMushroomThread

- **Felelősség**

A VMushroomThread egy gombaszál képi ábrázolása.

- **Össztályok**

JPanel → VItem → VMushroomThread

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**
- imagePath: "images/mushroomThread.png" – A szál képe.

- **Metódusok**

- VMushroomThread(int x, int y, String toolTipText): Konstruktor, amely betölti a képet. [protected]

### 11.3.22 VMushroomThreadConnection

- **Felelősség**

A VMushroomThreadConnection két VTecton típusú elem közötti kapcsolatot reprezentál grafikus szálként, amelyet vonalként rajzol ki.

- **Össztályok**

Nincs (nem örököl más osztályból).

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**
- start, end: A kapcsolat kezdő és végpontjai (VTecton).
- color: A vonal színe. [Color]
- thickness: A vonal vastagsága. [int]
- startPos, endPos: A szálak indexe a kezdő és vég objektumban. [int]

- **Metódusok**

- VMushroomThreadConnection(VTecton start, VTecton end, Color color): Inicializálja a kapcsolatot. [public]
- draw(Graphics2D g2d): Kirajzolja a kapcsolatot egyenes vonalként a megadott vastagsággal és színnel. [public]
- setColor(Color color): Beállítja a szál színét. [public]
- setThickness(int thickness): Beállítja a szál vastagságát. [public]

- `getStart()/getEnd()`: Visszaadja a kapcsolat kezdő/vég objektumát. [public]

### 11.3.23 GamePanel

#### • Felelősség

A játékfelület fő panelje. Összefogja a bal oldali irányítópultot (SidebarPanel) és a középső játéktér panelt (MiddlePanel), és elrendezi azokat a BorderLayout szerint.

#### • Ősosztályok

Object → Component → Container → JComponent → JPanel → GamePanel

#### • Interfészek

Nem valósít meg interfészt.

#### • Attribútumok

- – (nincsenek attribútumai)

#### • Metódusok

- + `GamePanel()` – Inicializálja az elrendezést, háttérszínt, és hozzáadja az oldalsó, valamint középső paneleket.

### 11.3.24 InsecterPop

#### • Felelősség

A rovarjátékos akciógombjait tartalmazó panel. A gombok: Move, Eat, Cut. Minden gomb eseménykezelőhöz van kötve.

#### • Ősosztályok

Object → Component → Container → JComponent → JPanel → InsecterPop

#### • Interfészek

Nem valósít meg interfészt.

#### • Attribútumok

- – (csak lokális változók)

#### • Metódusok

- + `InsecterPop()` – A gombok létrehozása, eseménykezelők beállítása, horizontális elrendezés.

### 11.3.25 MapPanel

#### • Felelősség

A térkép felület megjelenítéséért felelős panel. Játék közben a térképen megjeleníti a VTecton elemeket, valamint a köztük lévő VMushroomThreadConnection szálakat. Kezeli a kattintásokat, húzásokat, és játékos típus alapján dinamikusan megjeleníti a megfelelő akciópanelt.

#### • Ősosztályok

Object → Component → Container → JComponent → JPanel → MapPanel

## • Interfészek

Nem valósít meg interfészt.

## • Attribútumok

- - tectons: List<VTecton> – Tárolja a térképen lévő tektonokat.
- - threads: List<VMushroomThreadConnection> – Tárolja a tektonokat összekötő szálakat.
- - selectedTecton: VTecton – Éppen mozgatott tekton.
- - insecterPop: InsecterPop – Rovarjátékos gombpanel példány.
- - mushroomerPop: MushroomerPop – Gombaplayer panel példány.

## • Metódusok

- + MapPanel() – Inicializálja a panelt, létrehozza és hozzáadja a tektonokat, szálakat, egérfigyelőket.
- # paintComponent(Graphics g) – Kirajzolja a tektonokat és szálakat.
- + addTectonAtRandomPosition() – Véletlenszerű pozícióban új tekton hozzáadása.

### 11.3.26 MiddlePanel

## • Felelősség

A középső fő GUI panel. Tartalmazza a körinfót (TurnOrderPanel), a térképet (MapPanel) és egy leíró alsó panelt (TectonContentPanel).

## • Ősosztályok

Object → Component → Container → JComponent → JPanel → MiddlePanel

## • Interfészek

Nem valósít meg interfészt.

## • Attribútumok

- + turnOrderPanel: TurnOrderPanel – A játék körinformációit megjelenítő felső rész.
- + mapPanel: MapPanel – A játékterület térképe.
- + tectonContentPanel: TectonContentPanel – Az alsó infopanel.

## • Metódusok

- + MiddlePanel() – Inicializálja a gyermekpanelek elrendezését és stílusát.
- - createTitledBorder(String title): Border – Segédfüggvény szegély létrehozására.

### 11.3.27 MushroomerPop

## • Felelősség

A gombajátékos vezérlőpanelje, melyen a Throw Spore, Eat Insect és Grow Thread gombok találhatók. Eseménykezelőkkel van bővítve.

## • Ősosztályok

Object → Component → Container → JComponent → JPanel → MushroomerPop

## • Interfészek

Nem valósít meg interfészt.

## • Attribútumok

- – (csak lokális gombpéldányok)

- **Metódusok**
- + MushroomerPop() – A panel inicializálása, gombok létrehozása, eseménykezelők beállítása.

### 11.3.28 TectonContentPanel

- **Felelősség**

Az alsó középső információs szekció megjelenítéséért felelős GUI-panel. Jelenleg egy statikus címkét tartalmaz.

- **Össztályok**

Object → Component → Container → JComponent → JPanel → TectonContentPanel

- **Interfészek**

Nem valósít meg interfészt.

- **Attribútumok**
- – (nincs külön attribútuma)
- **Metódusok**
- + TectonContentPanel() – Inicializálja a panelt és beállítja a szegélyt, háttérszínt, elrendezést.
- - createTitledBorder(String title): Border – Szegély létrehozásához segédfüggvény.

### 11.3.29 TurnOrderPanel

- **Felelősség**

A játék felső sávját jeleníti meg, ahol a körinformációk láthatók (pl. melyik játékos következik).

- **Össztályok**

Object → Component → Container → JComponent → JPanel → TurnOrderPanel

Legősebb osztály ® Össztály2 ® Össztály3...

- **Interfészek**

Nem valósít meg interfészt.

- **Attribútumok**
- – (csak lokális JLabel komponens)
- **Metódusok**
- + TurnOrderPanel() – Beállítja a hátteret, elrendezést, és hozzáadja a feliratot.

### 11.3.30 MenuButton

- **Felelősség**

Egyedi kinézetű, menükben használatos gomb. Beállítja a gomb méretét, stílusát és színeit.

- **Össztályok**

Object → Component → Container → JComponent → AbstractButton → JButton → MenuButton

- **Interfészek**

Nem valósít meg külön interfészt.

- **Attribútumok**

- – (nincs)

- **Metódusok**

- + MenuButton(String name) – Konstruktor: beállítja a gomb nevét, méretét, színeit és fókusz viselkedését.

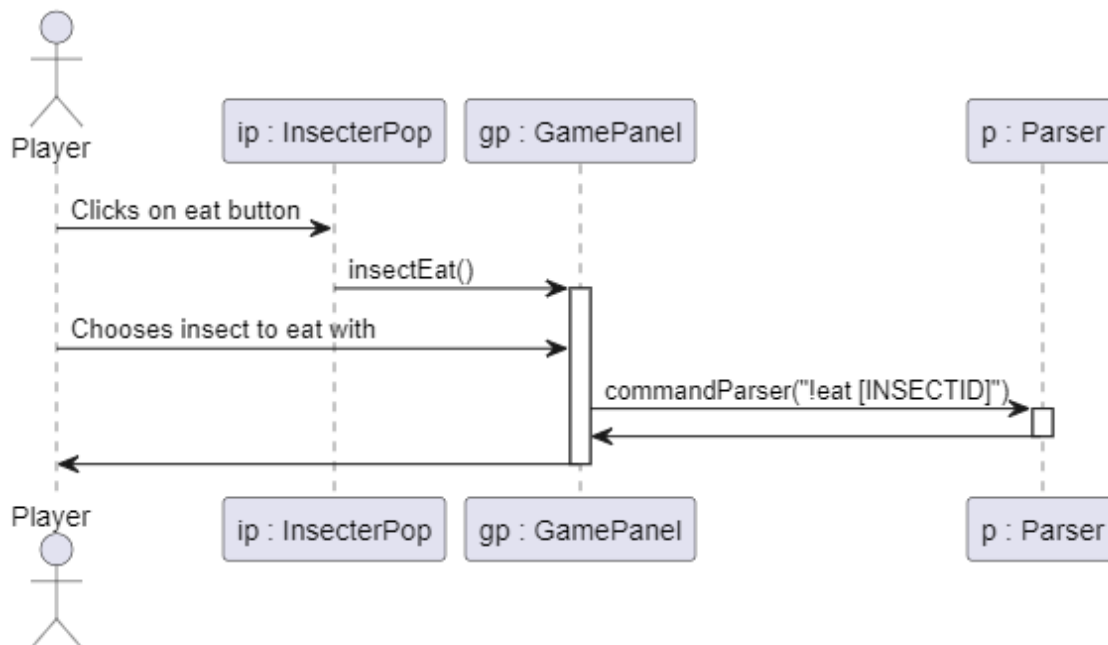
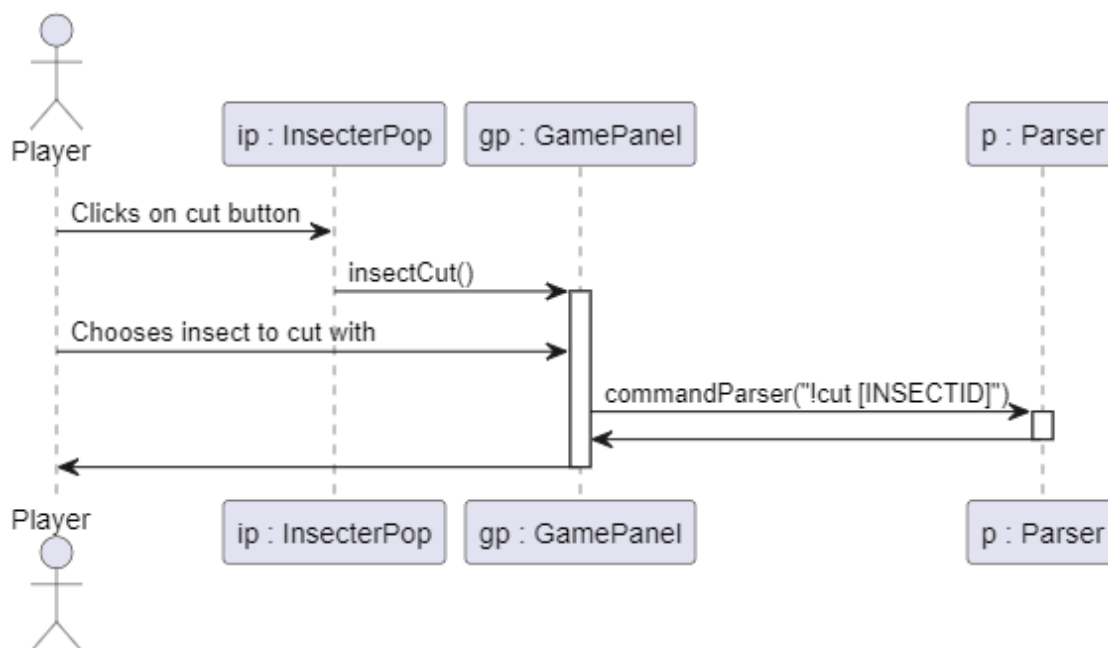
## 11.4 Kapcsolat az alkalmazói rendszerrel

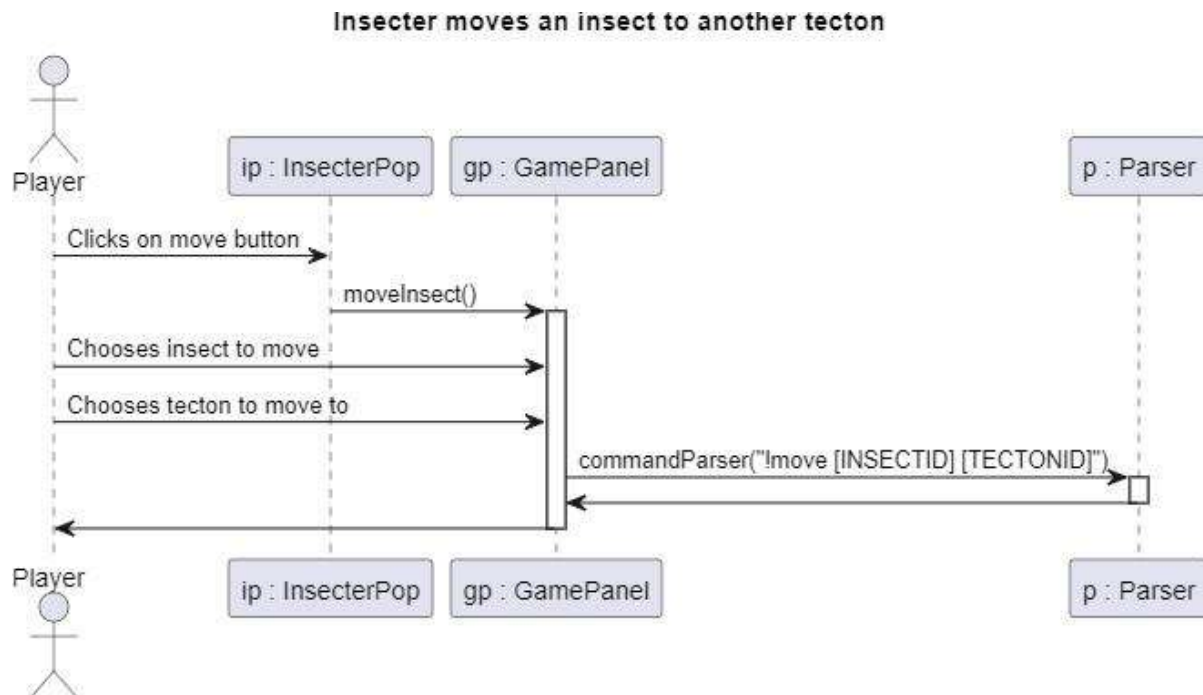
Ákos/Csabi

[Szekvencia-diagramokon ábrázolni kell a grafikus rendszer működését. Konzisztens kell legyen az előző alfejezetekkel. Minden metódus, ami ott szerepel, fel kell tűnjön valamelyik szekvenciában. Minden metódusnak, ami szekvenciában szerepel, szereplnie kell a valamelyik osztálydiagramon.]

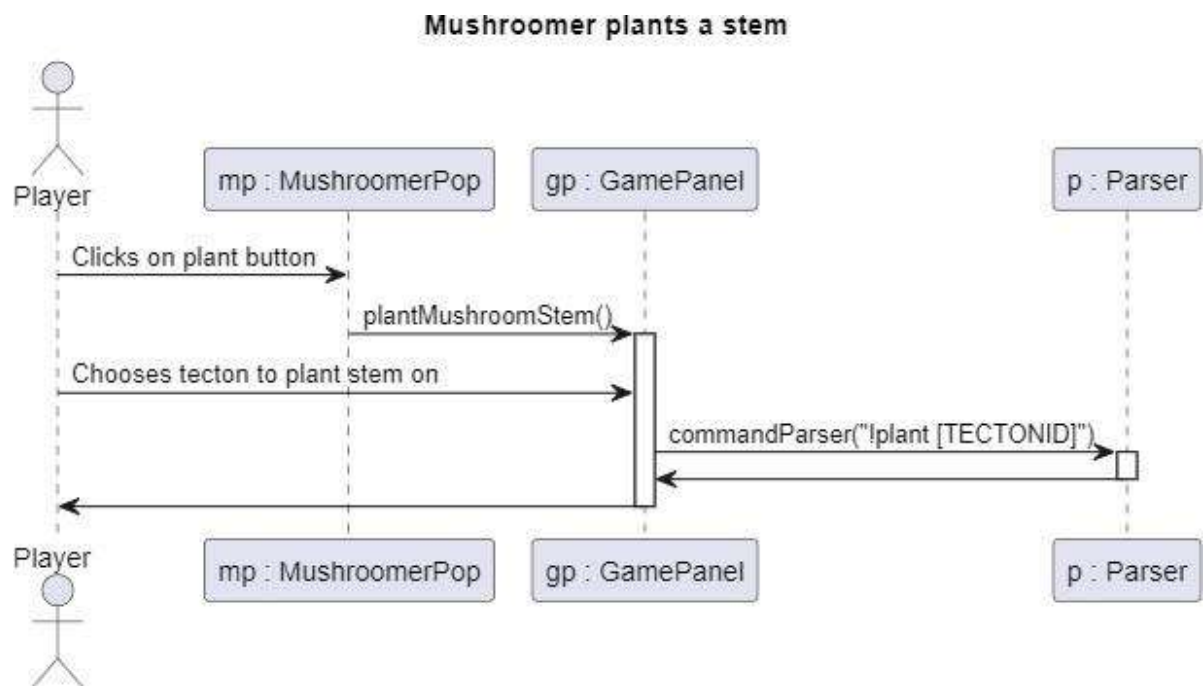
Rányomunk a Spóra dobásra, majd kiírja a chat hogy rányomtunk majd tectonra rányomunk és lerakja

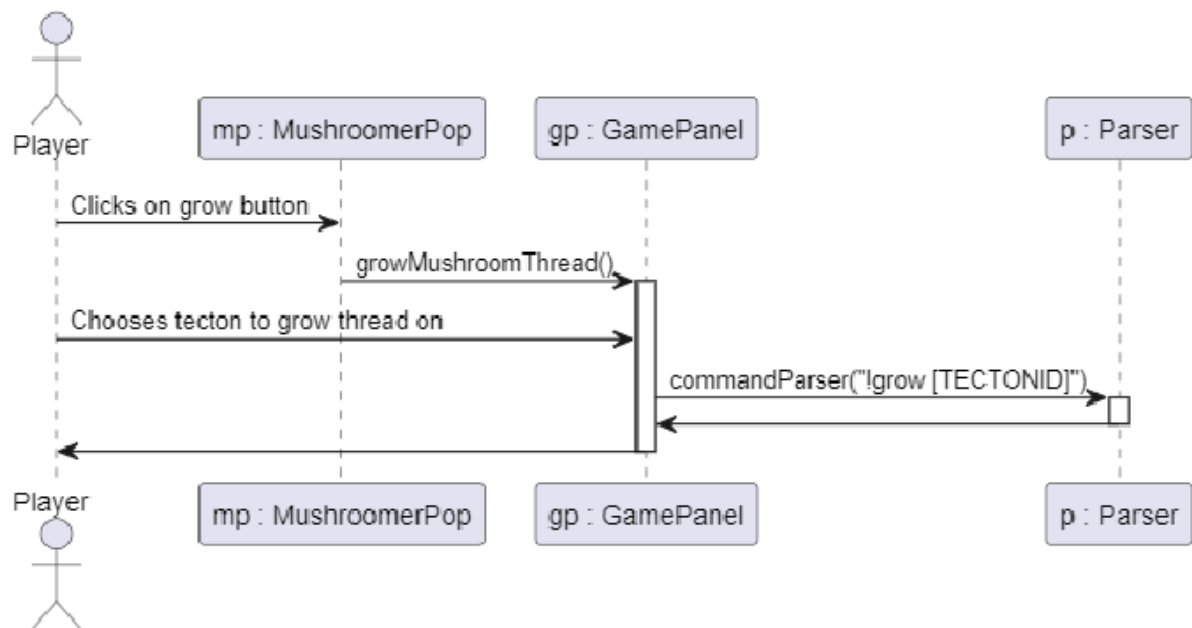
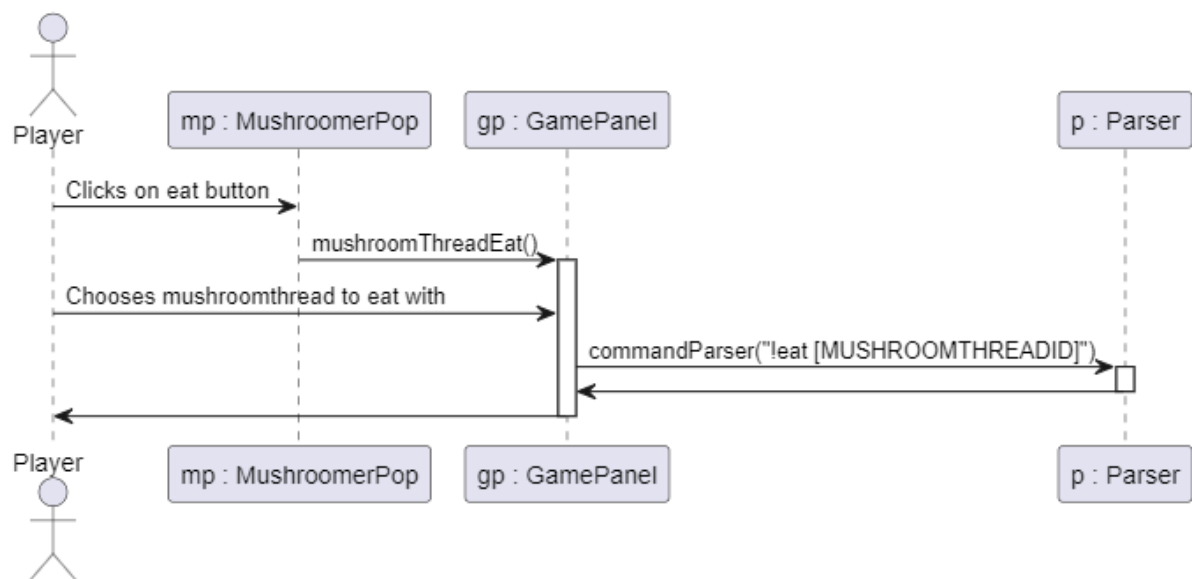


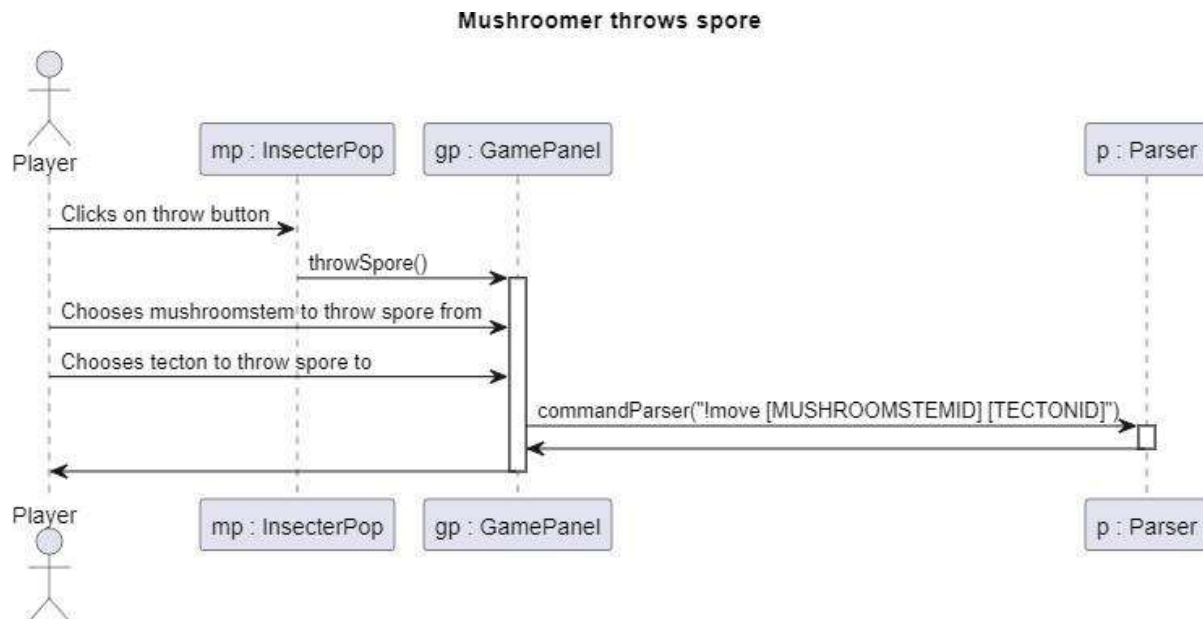
**11.4.1 Rovarász diagramok****Insecter eats a spore****Insecter cuts a mushroomthread with an insect**



### 11.4.2 Gombász diagramok



**Mushroomer grows a thread****Mushroomer eats an insect with a thread**



## 11.5 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2025.05.02 17:00	1.5 óra	Végvári Horváth Bohus	Értekezlet: Feladatok felosztása Döntések: 11.1 Horváth 11.2 Dúcz, Galambos 11.2.1 Bohus, Végvári 11.2.2 Végvári 11.3 Bohus, Végvári 11.4 Dúcz, Galambos
2025.05.02-04.	15 óra	Végvári	Tevékenység: 11.2, 11.3
2025.05.03 16:00	5 óra	Bohus	Tevékenység: Kód írása, osztályok
2025.05.04 8:00	10 óra	Horváth	Tevékenység: Ikonok tervezése
2025.05.04 18:00	2.5 óra	Dúcz	Tevékenység: 11.2, Objektumlista
2025.05.04 23:00	2 óra	Galambos	Tevékenység: 11.4, szekvenciadiagramok

# 13. Grafikus változat beadása

23 – totoro

Konzulens:

Haragos Gergő Viktor

## Csapattagok

Bohus János  
Ducz Ákos  
Galambos Csaba  
Horváth Avarka  
Végvári Péter

XYOVFZ	janos.bohus1@gmail.com
GC1RTE	akos.ducz@gmail.com
BVBQKN	galambos.csaba05@gmail.com
OODMXV	avarkahorvath@gmail.com
AW3JH1	vegvaripeter21@gmail.com

## 13. Grafikus változat beadása

### 13.1 Fordítási és futtatási útmutató

#### 13.1.1 Fájllista

Main.java 828 5/3/2025 17:26  
controller\Action.java 301 5/15/2025 14:41  
controller\Controller.java 5268 5/10/2025 20:50  
controller\visitor\DrawVisitor.java 5970 5/14/2025 17:11  
controller\visitor\GameObjectVisitor.java 3218 5/14/2025 17:11  
controller\visitor\NewObjectVisitor.java 6678 5/14/2025 17:11  
model\Game.java 13262 5/10/2025 20:50  
model\Map.java 5141 5/3/2025 17:26  
model\core\Entity.java 3281 5/3/2025 17:26  
model\core\GameObject.java 2138 5/10/2025 20:50  
model\core\IRound.java 237 5/3/2025 17:26  
model\core\ITurn.java 246 5/3/2025 17:26  
model\core\Player.java 1888 5/3/2025 17:26  
model\effect\ClawParalyzeEffect.java 325 5/3/2025 17:26  
model\effect\Effect.java 1404 5/3/2025 17:26  
model\effect\FastEffect.java 317 5/3/2025 17:26  
model\effect\ParalyzeEffect.java 320 5/3/2025 17:26  
model\effect\SlowEffect.java 319 5/3/2025 17:26  
model\effect\SplitEffect.java 258 5/3/2025 17:26  
model\insect\IInsect.java 1402 5/3/2025 17:26  
model\insect\Insect.java 9548 5/3/2025 17:26  
model\insect\Insecter.java 6255 5/3/2025 17:26  
model\mushroom\IStem.java 1325 5/3/2025 17:26  
model\mushroom\IThread.java 1364 5/3/2025 17:26  
model\mushroom\Mushroomer.java 12977 5/3/2025 17:26  
model\mushroom\MushroomStem.java 5403 5/3/2025 17:26  
model\mushroom\MushroomThread.java 5962 5/3/2025 17:26  
model\mushroom\spore\ClawParalyzingSpore.java 1083 5/3/2025 17:26  
model\mushroom\spore\ISpore.java 920 5/3/2025 17:26  
model\mushroom\spore\ParalyzingSpore.java 1697 5/3/2025 17:26  
model\mushroom\spore\SlowingSpore.java 1732 5/3/2025 17:26  
model\mushroom\spore\SpeedingSpore.java 1662 5/3/2025 17:26  
model\mushroom\spore\SplitterSpore.java 1681 5/3/2025 17:26  
model\mushroom\spore\Spore.java 2728 5/3/2025 17:26  
model\tecton\LifeSupportTecton.java 450 5/3/2025 17:26  
model\tecton\SingleThreadedTecton.java 781 5/3/2025 17:26  
model\tecton\StemlessTecton.java 615 5/3/2025 17:26  
model\tecton\Tecton.java 12282 5/3/2025 17:26  
model\tecton\ThreadConsumingTecton.java 478 5/3/2025 17:26  
resources\icons\insect\normal\insect\_normal\_1.png 16690 5/10/2025 20:50  
resources\icons\insect\normal\insect\_normal\_2.png 15505 5/10/2025 20:50  
resources\icons\insect\normal\insect\_normal\_3.png 16313 5/10/2025 20:50  
resources\icons\insect\normal\insect\_normal\_4.png 16472 5/10/2025 20:50  
resources\icons\insect\paralyzed\insect\_paralyzed\_1.png 32806 5/10/2025 20:50  
resources\icons\insect\paralyzed\insect\_paralyzed\_2.png 30871 5/10/2025 20:50

resources\icons\insect\paralyzed\insect\_paralyzed\_3.png 32302 5/10/2025 20:50  
resources\icons\insect\paralyzed\insect\_paralyzed\_4.png 32178 5/10/2025 20:50  
resources\icons\menu\MainMenu.png 2443428 5/11/2025 17:54  
resources\icons\menu\MainMenu2.png 45708 5/13/2025 21:44  
resources\icons\menu\winnerview.png 22030 5/18/2025 11:08  
resources\icons\mushroom\stem\mushroom\_stem\_1.png 15747 5/10/2025 20:50  
resources\icons\mushroom\stem\mushroom\_stem\_2.png 16001 5/10/2025 20:50  
resources\icons\mushroom\stem\mushroom\_stem\_3.png 15751 5/10/2025 20:50  
resources\icons\mushroom\stem\mushroom\_stem\_4.png 15884 5/10/2025 20:50  
resources\icons\mushroom\thread\mushroom\_thread\_1.png 24851 5/10/2025 20:50  
resources\icons\mushroom\thread\mushroom\_thread\_2.png 24623 5/10/2025 20:50  
resources\icons\mushroom\thread\mushroom\_thread\_3.png 24455 5/10/2025 20:50  
resources\icons\mushroom\thread\mushroom\_thread\_4.png 24824 5/10/2025 20:50  
resources\icons\spore\clawparalyzing\spore\_clawparalyzing\_1.png 27335 5/10/2025 20:50  
resources\icons\spore\clawparalyzing\spore\_clawparalyzing\_2.png 26556 5/10/2025 20:50  
resources\icons\spore\clawparalyzing\spore\_clawparalyzing\_3.png 26032 5/10/2025 20:50  
resources\icons\spore\clawparalyzing\spore\_clawparalyzing\_4.png 27129 5/10/2025 20:50  
resources\icons\spore\paralyzing\spore\_paralyzing\_1.png 33031 5/10/2025 20:50  
resources\icons\spore\paralyzing\spore\_paralyzing\_2.png 33537 5/10/2025 20:50  
resources\icons\spore\paralyzing\spore\_paralyzing\_3.png 28244 5/10/2025 20:50  
resources\icons\spore\paralyzing\spore\_paralyzing\_4.png 33247 5/10/2025 20:50  
resources\icons\spore\slowing\spore\_slowing\_1.png 16071 5/10/2025 20:50  
resources\icons\spore\slowing\spore\_slowing\_2.png 16039 5/10/2025 20:50  
resources\icons\spore\slowing\spore\_slowing\_3.png 16008 5/10/2025 20:50  
resources\icons\spore\slowing\spore\_slowing\_4.png 16106 5/10/2025 20:50  
resources\icons\spore\speeding\spore\_speeding\_1.png 17718 5/10/2025 20:50  
resources\icons\spore\speeding\spore\_speeding\_2.png 17785 5/10/2025 20:50  
resources\icons\spore\speeding\spore\_speeding\_3.png 17720 5/10/2025 20:50  
resources\icons\spore\speeding\spore\_speeding\_4.png 17850 5/10/2025 20:50  
resources\icons\spore\splitter\spore\_splitter4.png 24822 5/14/2025 17:11  
resources\icons\spore\splitter\spore\_splitter\_1.png 25149 5/14/2025 17:11  
resources\icons\spore\splitter\spore\_splitter\_2.png 24510 5/14/2025 17:11  
resources\icons\spore\splitter\spore\_splitter\_3.png 24553 5/14/2025 17:11  
resources\icons\tecton\tecton\_basic.png 1805 5/10/2025 20:50  
resources\icons\tecton\tecton\_lifesupport.png 15390 5/10/2025 20:50  
resources\icons\tecton\tecton\_singlethreaded.png 17834 5/10/2025 20:50  
resources\icons\tecton\tecton\_stemless.png 20912 5/10/2025 20:50  
resources\icons\tecton\tecton\_threadconsuming.png 29488 5/10/2025 20:50  
view\ImageManager.java 4232 5/10/2025 20:50  
view\IUpdateGUI.java 81 5/17/2025 21:39  
view\view.iml 450 5/10/2025 20:50  
view\View.java 2855 5/10/2025 20:50  
view\game\GamePanel.java 2045 5/3/2025 17:20  
view\game\ImagePanel.java 1870 5/10/2025 20:50  
view\game\MapPanel.java 3663 5/3/2025 17:20  
view\game\MiddlePanel.java 2627 5/3/2025 17:20  
view\game\TurnOrderPanel.java 1888 5/3/2025 17:20  
view\game\VMushroomThreadConnection.java 1407 5/10/2025 20:50  
view\game\buttons\GameButton.java 1935 5/13/2025 16:28  
view\game\buttons\InsectButton.java 2963 5/14/2025 17:11

```

view\game\buttons\MushroomStemButton.java    2940  5/14/2025 17:11
view\game\buttons\MushroomThreadButton.java  2923  5/14/2025 17:11
view\game\buttons\SporeButton.java 2205  5/14/2025 17:11
view\game\buttons\TectonButton.java    6327  5/13/2025 16:28
view\game\contentpanel\BasePanel.java    1972  5/14/2025 17:11
view\game\contentpanel\ContentPanel.java 4538  5/14/2025 17:11
view\game\contentpanel\InsectsPanel.java  1289  5/14/2025 17:11
view\game\contentpanel\SporesPanel.java  1228  5/14/2025 17:11
view\game\contentpanel\StemsPanel.java    1253  5/14/2025 17:11
view\game\contentpanel\ThreadsPanel.java 1215  5/14/2025 17:11
view\game\popup\InsectPop.java    2745  5/15/2025 14:41
view\game\popup\MushroomStemPop.java 1790  5/15/2025 14:41
view\game\popup\ThreadPop.java    2480  5/15/2025 14:41
view\game\sidebar\ButtonsPanel.java    2803  5/4/2025 18:48
view\game\sidebar\ChatLogPanel.java    2181  5/4/2025 18:48
view\game\sidebar\SidebarButton.java    1150  5/4/2025 18:48
view\game\sidebar\SidebarPanel.java    1730  5/4/2025 18:48
view\game\winner\WinnerView.java 2786  5/18/2025 11:05
view\mainMenu\MainMenu.java    3929  5/3/2025 17:22
view\mainMenu\MenuButton.java    1395  5/3/2025 17:22

```

### 13.1.2 Fordítás és telepítés

A program fordítása a Visual Studio Code-ban, Extension Pack for Java kiegészítővel történik, automatikusan futtatáskor. A fordításhoz OpenJDK 17 vagy annál újabb szükséges.

### 13.1.3 Futtatás

A program fordítása a Visual Studio Code-ban, Extension Pack for Java kiegészítővel történik, automatikusan futtatáskor. A fordításhoz OpenJDK 17 vagy annál újabb szükséges.

## 13.2 Értékelés

Tag neve	Tag neptun	Munka százalékban
Bohus János	XYOVFZ	20%
Dúcz Ákos	GC1RTE	20%
Galampos Csaba	BVBQKN	20%
Horváth Avarka	OODMXV	20%
Végvári Péter	AW3JH1	20%



### 13.3 Napló

Kezdet	Időtartam	Résztevők	Leírás
2025.05.07. 12:25	1,5 óra	Bohus Dúcz Galambos Horváth Végvári	Értekezlet. Döntés: Feladatkiosztás
2025.05.09 10:30	1,5 óra	Dúcz	Tevékenység: Grafikus felület megvalósításának megtervezése, elkezdése
2025.05.09 10:30	1,5 óra	Galambos	Tevékenység: Grafikus felület megvalósításának megtervezése, elkezdése
2025.07-18	15 óra	Bohus	Tevékenység: Program írása
2025.07-18	20 óra	Végvári	Tevékenység: Program írása
2025.07-18	15 óra	Dúcz	Tevékenység: Program írása
2025.07-18	15 óra	Galambos	Tevékenység: Program írása
2025.07-18	15 óra	Horváth	Tevékenység: Program írása

## 14. Összefoglalás

### 14.1 A projektre fordított összes munkaidő

[személyenként és összesen táblázatban ]

Tag neve	Munkaidő (óra)
Bohus	110
Dúcz	115
Galambos	112
Horváth	130
Végvári	145
<b>Összesen</b>	<b>612</b>

- ***A feltöltött programok forrásainak száma***

Fázis	Kódsorok száma
Szkeleton	1602
Prototípus	3662
Grafikus változat	5382
<b>Összesen</b>	<b>5382</b>

### 14.2• Projekt összegzés

[A projekt tapasztalatait összegző részben a csapatoknak a projektről kialakult véleményét várjuk. A megválaszolandók köre az alábbi:]

#### 14.2.1 Mit tanultak a projektből konkrétan és általában?

A projekt során megtanultuk, hogyan lehet hatékonyan csapatban dolgozni. Tapasztalatot szereztünk abban, hogyan kell a munkát beütemezni, és milyen módon érdemes egymással kommunikálni a gördülékeny együttműködés érdekében.

#### 14.2.2 Mi volt a legnehezebb és a legkönnyebb?

A legnehezebb az ötletek összeegyeztetése volt – gyakran nehézséget jelentett eldönteni, melyik irányba induljunk el. A legkönnyebb rész talán a feladatok felosztása volt, bár ezt többször is módosítottuk a munka során.

#### **14.2.3      Összhangban állt-e az idő és a pontszám az elvégzendő feladatokkal?**

A legtöbb feladatra elegendő idő állt rendelkezésre, azonban a grafikai részhez hasznos lett volna még körülbelül egy hét.

#### **14.2.4      Ha nem, akkor hol okozott ez nehézséget?**

A grafikus feladatok kivitelezése jelentett kihívást az idő szűkössége miatt, főleg azért, mert az utolsó hetekben más tantárgyakból is sok számonkérés volt.

#### **14.2.5      Milyen változtatási javaslatuk van?**

Az előző pontnak megfelelően szerkezeti átalakítást javasolnánk, hogy a grafikus munkákra is elegendő idő jusson. Emellett jó lenne, ha a feladatokat nem kellene nyomtatott formában is beadni.

#### **14.2.6      Milyen feladatot ajánlanának a projektre?**

Véleményünk szerint egy egyszerűbb tower defense típusú játék készítése jó jövőbeli projektfeladat lehetne.

#### **14.2.7      Egyéb kritika és javaslat**