

2. Követelmények, Projekt, Funkcionalitás

82 – being_chilling

Konzulens:
Tóth Bence

Csapattagok

Yin Jun	TN1KZI	1171276297yin@gmail.com
Bene Máté	K4C4W1	matthew.be04@gmail.com
Kattan Ibrahim Gábor	ZU22JJ	kattanbarhum2@gmail.com
Ji Jin Yan	ZGCKAL	jijinyan0201@gmail.com
Ji Yuang	UI7V2S	yuangji688@gmail.com

2025.02.23.

2. Követelmény, projekt, funkcionalitás

2.1 Bevezetés

2.1.1 Cél

A dokumentum célja, hogy megmutassuk Fungorium projekt kereteit, és a kért funkciók hogyan működnek. Mivel ezt mind a megrendelői, mind a szállítói oldal megtekinti és jóváhagyja, biztosítja, hogy minden fél elvárásainak megfelel, és elfogadható projekt jöjjön létre.

2.1.2 Szakterület

A szoftver egy stratégiai társasjáték, ami elsősorban kikapcsolódásra, szórakozásra lett kitalálva. Viszont a játék kiegyensúlyozott és igazságos szabályrendszere biztosítja, hogy minden játékos azonos esélyekkel induljon, ezért akár versenyszerű környezetben is megállja a helyét.

Emellett a játék egy érdekes bevezethető lehet gyereknek a gombák és rovarok ökológiai folyamatainak vizsgálatában, ezért tanárok is használhatják, mint érdekes és interaktív edukációs cédra.

Másrészt megismerhetik a biológiai és ökológiai folyamatokat, ahol a gombák, és a rovarok versengenek a túlélésért. Ez egy tökéletes példa lehet oktatás céljából is, hogy megmutassuk a természetben folyó versengési folyamatokat.

2.1.3 Definíciók, rövidítések

Játék = Fungorium játék program

DC = Discord

Tektonok= különböző alakú kéregdarabok

MUST = kötelező

pl = például

lab = laboratórium

stb = és a többi

UI = User Interface

2.1.4 Hivatkozások

Feladat részletes leírása: <https://www.iit.bme.hu/file/11582/feladat>

Elkészült feladatbeadó rendszer: <https://devil.iit.bme.hu:9181/hercules/login>

Szoftver projekt lab moodle: <https://edu.vik.bme.hu/course/view.php?id=14010>

Microsoft Teams: <https://teams.microsoft.com/v2/>

Github: <https://github.com/>

UML:<https://app.diagrams.net/>

2.1.5 Összefoglalás

A dokumentum továbbiakban a játék program architekturális képét, fontosabb alrendszerket, funkciókat mutatja be. Majd a futtatáshoz szükséges követelményeket, feltételeket írja le. Egy use-case diagrammal megmutatja felhasználó és program közötti kapcsolatot, és szótár, ami egy általánosabb fogalomtár, amely a követelmények alapján készül.

2.2 Áttekintés

2.2.1 Általános áttekintés

Játéklogika:

Ez az alrendszer felelős a játék szabályainak kezeléséért, mint például a gombák és rovarok interakciói, a spórák terjedése, a fonalak növekedése, valamint a tektonok közötti kapcsolat fenntartása. Az alapvető játékmenetet itt kell implementálni.

User Interface:

A felhasználók számára biztosít egy grafikus, megjelenést, ahol a gombákat, rovarokat, figyelhetik játék közben. Az UI-ban lehet menüt választani, ahol elindítani, és kilépni a programból.

Controll:

A felhasználói interakciók történhetnek egérrel, és billentyűzetről.

Kapcsolatok:

A játéklogika, és UI között kapcsolat van, mert a UI-nak tudnia kell, az aktuális állapotot például hol helyezkednek el a gombák, rovarok.

A Controll és Játéklogika: a Controll adja meg, hogy milyen inputot kapott, és átadja a játék logikának.

Hálózat igény nem szükséges, mivel offline játék lesz, ami lokálisan fut.

Adattárolási elvárás nincs.

2.2.2 Funkciók

A játékban kétféle szerep közül választhatnak a játékosok: lehetnek rovarászok, akik rovarokat irányítanak, vagy gombászok, akik gombákat növesztenek. Egy játékos a saját szerepében lévő játékosok ellen játszik, tehát a rovarászok más rovarászokkal, míg a gombászok más gombászokkal küzdenek meg. A játékban 4-6 játékos vehet részt csapatonként, és körökre osztott játékmenet szerint zajlik. A teljes játék 40 körből áll, felváltva 20-20 kört játszanak a gombászok és a rovarászok. A játékot a gombászok kezdkik, a rovarászoknak egy darab rovara van.

A játék Fungorium nevű bolygón játszódik, amely tektonokból épül fel. Ezek a tektonok elképzelhetők földrészeken, amelyek között rések találhatók. Egy tekton több másik tekton szomszédja is lehet, és nincsenek korlátozások a szomszédok számát illetően. A tektonok véletlenszerű hatásokkal rendelkeznek, amelyeket a létrejövésükkel véletlenszerűen kapnak meg.

A tektonok a következő típusúak lehetnek:

- Fonalszívó tekton: Olyan tekton, amelyen a rajta lévő gombafonalak egy idő után elpusztulnak és csak egy gombafonál nőhet.
- Többfonalas tekton: Olyan tekton, ahol több gomba fonala is kereszteződhet.
- Egyfonalas tekton: Olyan tekton, ahol egyszerre csak egy gomba fonala tud nőni.
- Gombanélküli tekton: Olyan tekton, ahol fonal lehet, de gombatest nem tud nőni és csak egy gombafonál nőhet.

Minden 5. körben egy tekton kettéválik, két új tektonná alakul, amelyek növekedése független lesz egymástól. A kettéválás során az eredeti tektonon lévő fonalak elszakadnak, megszakítva a fonalak közötti kapcsolatot. Tekton csak akkor tud kettéválni, ha nincsen rajta gombatest, de olyan tekton nem törhet össze, amelyen rovar tartózkodik.

A gombászok célja, hogy a lehető legtöbb gombatestet növesszenek a játék végéig. A gombák két részből állnak: gombafonalakból és gombatestekből. A gombafonalak a gombatestekből nőnek ki, és hidat képeznek a tektonok között, lehetővé téve a spórák terjedését.

A gombák spórákat termelnek, amelyek terjesztik a gombát más tektonokra. A rovarok célja az, hogy minél több spórát fogyasszanak el. A spóráknak azonos tápanyagmértékük van, illetve esélyük van arra, hogy az alábbi hatások egyikével bírjanak:

- Lassító spóra: A rovart a következő körben csak 1 tektonnyi mozgásra korlátozza (2 helyett).
- Gyorsító spóra: A rovar 3 tektonnyit mozoghat 1 körig (2 helyett).
- Bénító spóra: nem vághat fonalat a rovar 2 körig.
- Némító spóra: A rovar 2 körig nem ehet, így nem tud tápanyaghoz jutni.

A gombák spóraterjesztési szabályai is kötöttek. Egy gombatest csak 2 kör után képes spórát szórni, kezdetben csak a szomszédos tektonokra szórhat, majd további két három műlva fejlődik, és a szomszédoch szomszédaira is tud szórni. Egy körben egyszerre lehet gombatestenként spórát szórni és fonalat növeszteni. A fonálnövesztés vagy meglévő fonal tovább növesztéséből, vagy pedig új fonal képzéséből áll. Ha olyan tektonra nő egy fonal, amelyen van spóra, akkor a következő körben eggyel többet tud nőni. Egy gombatest növekedésének az a feltétele, hogy három meglévő spóra legyen a tektonon, és akkor nő csak gombatest, ha a játékos úgy dönt. Ha egy gombatest 5 alkalommal szórt spórát, akkor elpusztul, de a halott gombatestek is beleszámítanak a gombászok végső pontszámába.

A rovarászok csak egyfélle rovart irányítanak, amelyek csak a gombafonalakat követve képesek mozogni a tektonok között. A rovarok nem tudnak átlépmi a tektonok közötti réseken, csak ha gombafonal köt össze két tektont. Alapértelmezés szerint a rovarok 2 tektonnyit mozoghatnak egy körben. Gyorsító spóra hatására ez 3 tektonnyira növekszik egy kör erejéig, míg Lassító spóra esetén csak 1 tektonnyira csökken a következő körben.

A rovarok fonálvágó képességgel rendelkeznek, amely stratégiai előnyt biztosít, mert elvághatják a fonalakat, ezzel megszakítva a többi rovar útvonalát. Némító spóra hatására azonban 2 körig nem használhatják ezt a képességet.

A gombászok győzelme a gombatestek számának maximalizálásán múlik, beleértve a halott gombatesteket is. A rovarászok akkor győznek, ha rovaraik a legtöbb tápanyagot fogyasztották el a játék végére. A játék során stratégiai döntéseket kell hozni. A gombászoknak a spórák szórásának időzítését és a fonalak irányítását, míg a rovarászoknak a mozgásukat és a fonálvágást kell taktikusan megtervezniük.

2.2.3 Felhasználók

A játék elsősorban stratégiai gondolkodást kedvelő játékosokat céloz meg, akik élvezik a körökre osztott taktikázást és az aszimmetrikus szerepekből adódó különböző játékstílusokat. Emellett vonzó lehet versengő játékosoknak, akik másokkal akarnak megmérkőzni, valamint kreatív gondolkodóknak, akik szeretik a nem hagyományos megoldásokat és a kombinációk kihasználását.

2.2.4 Korlátozások

Az elkészült programnak szemantikailag az elvárt módon, valamint stabilan kell működnie, a program képesnek kell lennie kisebb hibák kezelni, pl.: egy nem várt játékosművelet megfelelő kezelésére, anélkül, hogy a program összeomlik.

2.2.5 Feltételezések, kapcsolatok

A projekt követelményei és tervezési inspirációja teljes mértékben a BMEVIIIA02 tárgy oldalán lévő „Feladat” dokumentumon alapul (<https://www.iit.bme.hu/file/11582/feladat>). Ebben a dokumentumban található tartalom nagy részét követi az említett dokumentumban meghatározott projetcélt, előre meghatározott funkciót és dokumentum vázát.

2.3 Követelmények

2.3.1 Funkcionális követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Use-case
FUNGK001	A játékosok két szerep közül választhatnak: rovarászok és gombászok.	A játék elindítása után, miután választott csapatot a játékos, abba a csapatba került e, amelyiket választotta.	MUST	Feladatkiírás	Játékos szerep választása
FUNGK002	A gombászok irányíthatják a gombafonalak növekedési irányát.	A fonál növekedése után a gombafonal a megfelelő tektonra átjut és a rovarok át tudnak így menni arra a tektonra a gombafonálon keresztül.	MUST	Feladatkiírás	Gombafonal irányítás
FUNGK003	A gombatestből csak szomszédos tektonra lehet gombafonalat növeszteni.	Csak akkor lehet gombafonalat növeszteni egy tektonra, ha a szomszédos tektonon van gombatest.	MUST	Csapat által	Gombafonal irányítás
FUNGK004	A gombászok kezdeményezhetik a gombatestek spóraszórását.	A spóra, melyet a gombász a gombatestből kiszórt, a megfelelő tektonra került.	MUST	Feladatkiírás	Spóra szórása
FUNGK005	Egy adott tektonon a gombász eldöntheti, hogy nőjön-e gombatest, ha van gombafonal a tektonon és mellé elegendő spóra is.	Ha a leírásban leírt követelményeknek eleget tesz, akkor az adott tektonon gombatest fog nőni és a spóra egy része a	MUST	Feladatkiírás	Új gombatest növesztése

		növesztéshez van felhasználva.			
FUNGK006	A rovarászok el tudják vágni a gombafonalakat.	A rovarász miután elvág egy gombafonalat, akkor azon a tektonon, amelyen nincs gombatest a gombafonál elpusztul.	MUST	Feladatkiírás	Rovarok irányítása
FUNGK007	A rovarászok a rovarokat mozgatják.	A rovar az egyik tektonról a másik tektonra kerül át.	MUST	Feladatleírás	Rovarok irányítása
FUNGK008	A rovarok különböző spórákat fogyasztva, különböző státusz hatást kaphatnak.	Miután egy rovar megevett egy spórát négy előre meghatározott hatások közül kapni fog egyet.	MUST	Feladatleírás	Rovarok táplálkozása
FUNGK009	A spóráknak adott, és minden egyenlő tápanyagtartalma van a rovarok számára.	A rovarok tápanyaga annyival nő, amennyi a spóra tápanyaga volt.	MUST	Feladatleírás	Rovarok táplálkozása
FUNGK010	A tektonok lehetnek fonalszívó, többfonalas, egyfonalas és gomba nélküli tektonok, melyeket a játék elindítása/generálás után lesz generálva.	Az adott tektonon csak olyan tulajdonsággal rendelkezik, amely a névből eredően meg van határozva.	MUST	Feladatkiírás	Játéktér létrehozása
FUNGK011	A gombák generálása játékosonként.	Minden játékosnak ugyanannyi gombával kell kezdenie a játékot.	MUST	Feladatkiírás	Játéktér létrehozása

FUNGK01 2	A Tektonok időnként kettétörnek	Egy tektonból kettő lesz, és a gombafonalak elpusztulnak ami a tektonon volt.	MUST	Feladatkiírás	Tektonok Kettétörése
FUNGK01 3	A játék turn-based elven működik.	Minden egyes játékos legalább egyszer tud lépni, mielőtt kiírja, hogy második kör.	MUST	Csapat	Játék működése
FUNGK01 4	Minden csapatban 4 játékosnak kell legalább lennie, maximum 6 játékos lehet.	Ha kevesebb játékos van a játék nem indul el, viszont több játékost a program nem enged felvenni.	MUST	Csapat	Játék működése
FUNGK01 5	A gombászok közül az nyer akinek a legtöbb gombatestje van.	A játék vége menüben a gombászok közül a nyertes az akinek a legtöbb gomba teste van.	MUST	Feladatkiírás	Játék működése
FUNGK01 6	A rovarászok közül az nyer akinek a legtöbb tápanyaga van.	A játék vége menüben a rovarászok közül a nyertes az akinek a legtöbb a tápanyagtartalma.	MUST	Feladatkiírás	Játék működése

2.3.2 Erőforrásokkal kapcsolatos követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás
FUNGE01	A fejlesztés java nyelven fog történni.	Csapattagok ellenőrzik, hogy a kód a java elveknek megfelelően lett megírva.	MUST	Csapattagok

FUNGE002	A szoftver működéséhez szükséges a java letöltése a számítógépre.	Java megléte a számítógépen	MUST	Java alkalmazások követelménye
FUNGE003	A szoftver windows 11 operációs rendszeren és windows 10 operációs rendszeren lesz tesztelve	Fordítás ezeken az operációs rendszereken	MUST	Csapattagok

2.3.3 Átadással kapcsolatos követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás
FUNGÁ001	A szoftver átadása a megrendelő által meghatározott formában történik.	A biztosított sablonnal való összevetés.	MUST	Megrendelő
FUNGÁ002	A dokumentumokat hétről hétre a megrendelő által biztosított helyiségen kerül leadásra.	A dokumentum leadó aláírásával.	MUST	Megrendelő
FUNGÁ003	A szoftver telepítése nem szükséges, csak az összes fájl megléte ami a szoftver futtatásához szükséges.	A program nem vagy hibásan működik az összes fájl megléte nélkül.	SHOULD	Csapattagok

2.3.4 Egyéb nem funkcionális követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás
FUNGEN001	A szoftver nem gyűjt adatot a felhasználóról.	A szoftver nem csatlakozik az internethez, csak ha a játékelemek miatt szükséges.	MUST	Csapat

2.4 Lényeges use-case-ek

2.4.1 Use-case leírások

Use-case neve	Gombafonal irányítás
Rövid leírás	Gombafonal növekedés iránya meghatározása
Aktorok	Gombászok
Forgatókönyv	A gombászok választhatja, hogy a gomba merre nőjön a gombafonal, amit azzal a következménnyel jár, hogy a tektonok a gombafonal összekötésével lehet egymásra juttatni.

Use-case neve	Spóra szórása
Rövid leírás	A gombász elindithatja a spóraszórást egy meglévő gombatestből
Aktorok	Gombászok
Forgatókönyv	A gombász választhatja, hogy a spórát szórja, amit azzal a következménnyel jár, hogy a tektonon a spóra száma nő és a gomba kevesebb spórát tud szórni.

Use-case neve	Új gombatest növesztése
Rövid leírás	Új gombatest növesztésének a lehetősége.
Aktorok	Gombászok
Forgatókönyv	Ha egy tektonon van gombafonal és elegendő spóra, a gombászok eldönthetik, hogy azon a tektonon növesztenek-e gombatestet.

Use-case neve	Rovarok irányítása
Rövid leírás	A rovarászok eldönthetik hogy a rovarok mit csináljanak.
Aktorok	Rovarászok
Forgatókönyv	A rovarászok választhatnak, hogy mit csináljanak a rovarok, elvágjanak egy gombafonalat, vagy mozogjanak.

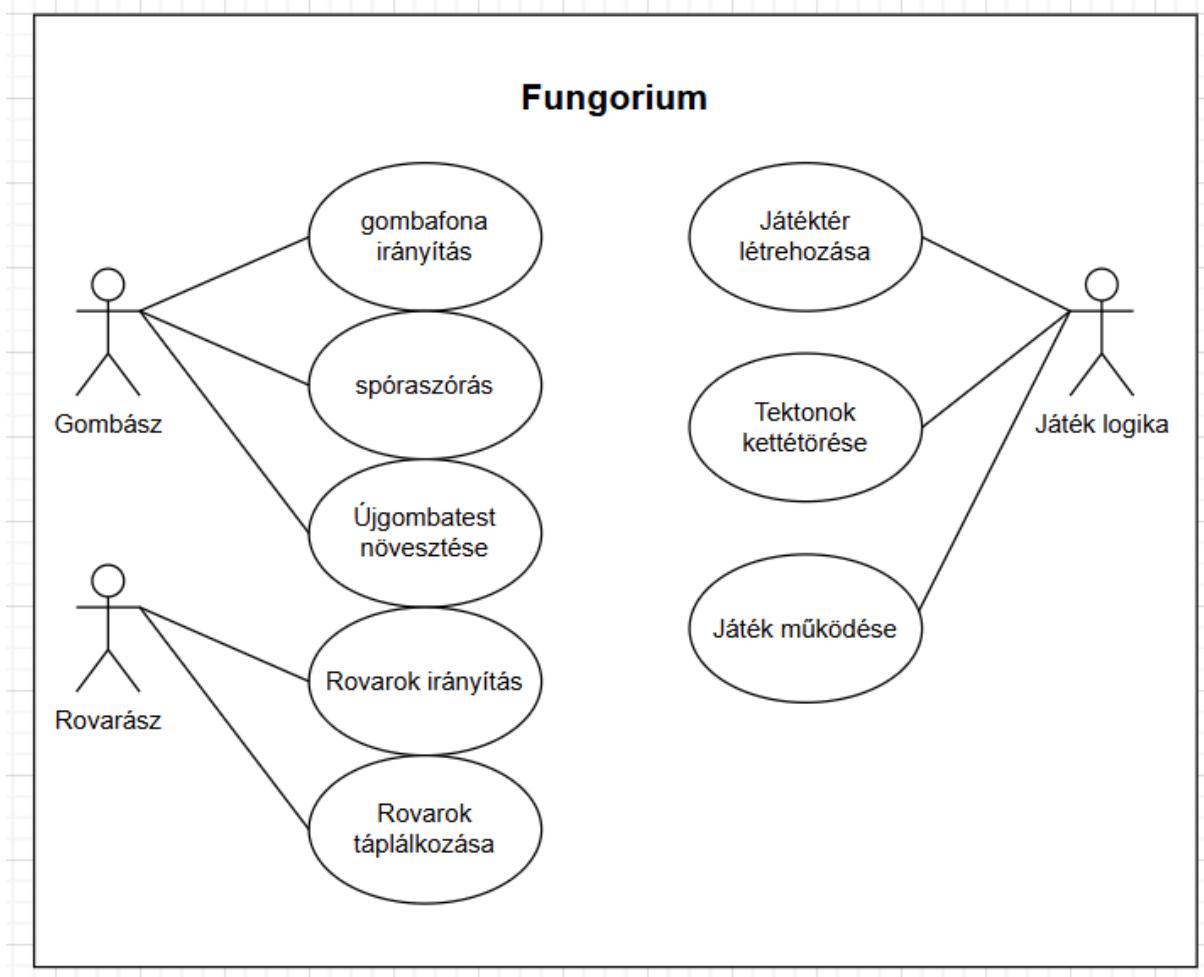
Use-case neve	Játéktér létrehozása
Rövid leírás	A játéktér generálása, beleértve a tektonokat és gombákat is.
Aktorok	Játék logika

Use-case neve	Rovarok táplálkozása
Rövid leírás	Rovarok spórákat ehetnek.
Aktorok	Rovarászok
Forgatókönyv	A rovarok különböző spórákat fogyasztva, különböző státusz hatást kaphatnak, melyek befolyásolják a rovarássz következő körét.

Use-case neve	Tektonok kettétörése
Rövid leírás	A Tektonok időnként kettétörése
Aktorok	Játék logika
Forgatókönyv	A tekton kettétörése a gombafonál elszakadásával jár. A tektonból 2 tekton lesz.

Use-case neve	Játék működése
Rövid leírás	A játék alapvető szabályai/működése
Aktorok	Játék logika
Forgatókönyv	A játék turn-based, vagyis minden játékos lép legalább egyszer a kör vége előtt. Hány játékos játszhat a játékkal, hányan lehet maximum játszani, ki nyer a csapatok közül.

2.4.2 Use-case diagram



2.5 Szótár

Bénító spóra: Olyan spóra, amely egy időre megbénítja a rovart.

Egyfonalas tekton: Olyan tekton, amelyen egyszerre csak egyetlen gombafonal nőhet.

Elpusztíthatatlan fonal: A játék generálásakor ezek a fonalok generálódnak, hogy megkönnyítsék a rovar mozgását

Fonalszívó tekton: Olyan tekton, amelyen a rajta lévő gombafonalak egy időn belül elpusztulnak.

Fungorium: Egy bolygó, ahol gombák és rovarok élnek, és amelynek felszínét tektonok alkotják.

Gombafonalak: A gombák egyik része, amelyek a tektonok felületén terjednek és hídként szolgálnak más tektonokra.

Gombanélküli tekton: Olyan tekton, ahol fonal lehet, de gombatest nem tud nőni.

Gombatest: A gombák által egyik része, amely spórákat termel és a gombafonalakból fejlődik ki. Egy gombatest meghalhat, amikor egy megszabott limiten túl szór spórát.

Gombák: A bolygón a tektonokon nőhetnek, gombafonalakból és gombatestből állnak, spórákat termelnek és terjedni képesek.

Gombászok: A játékosok egy csoportja, akik a gombák terjedését irányítják, feladata a gombatestek számának maximalizása a játék során.

Gyorsító spóra: Olyan spóra, amely a rovar elfogyasztás után felgyorsítja maradandóan.

Hatásos spórák: Olyan spórák, amelyek különböző hatásokkal hatnak a rovarokra, például gyorsító spóra, lassító spóra vagy bénító spóra stb..

Kéregdarabok (tektonok): A bolygó felszínét alkotó szilárd darabok, amelyek egy viszkózus rétegen úsznak.

Lassító spóra: Olyan spóra, amely a rovar elfogyasztás után lelassítja maradandóan.

Némító spóra: Olyan spóra, amely a rovar elfogyasztás után megnémítja a rovart, azaz elveszi a fonalvágó képességét a rovarnak.

Rovarok: A bolygón élőlények, amelyek a gombafonalakat követve közlekednek, spórákkal táplálkoznak, és képesek a fonalakat elvágni.

Rovarászok: A játékosok egy csoportja, akik különféle rovarokat irányítanak, feladata a tápanyag számának maximalizása a játék során.

Tápanyagtartalma: minden spórának ugyanannyi tápanyagot tartalmaz.

Tektonok kettétörése: Egy tekton széttörése két különálló részre, amely során a gombafonalak is elszakadhatnak.

Tektonok közti rés: A kéregdarabok közötti szűk tér, amelyen a rovarok nem tudnak áthaladni, de a gombafonalak átnőhetnek rajta.

Többfonalas tekton: Olyan tekton, amelyen több különböző gombafonal is átnőhet és egymással kereszteződhetnek.

2.6 Projekt terv

Fázis	Felelős	Határidő
1. Csapatalakítás	Yin	2025.02.14
2. Követelmény, projekt, funkcionalitás	Bene	2025.02.24
3. Analízis modell I.	Kattan	2025.03.03
4. Analízis modell II.	Ji Jin Yan	2025.03.10
5. Szkeleton tervezés	Ji Yuang	2025.03.17
6. Szkeleton elkészítés	Yin	2025.03.24
7. Prototípus koncepció	Bene	2025.03.31
8. Részletes tervezek	Kattan	2025.04.14
9. Prototípus elkészítés	Ji Jin Yan	2025.04.28
10. Grafikus változat tervezés	Ji Yuang	2025.05.05
11. Grafikus változat elkészítés	Yin	2025.05.19
12. Egyesített dokumentáció	Bene	2025.05.23

2.7 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2025.02.14 11:00	1 óra	Yin Bene Ji Jin Yan Ji Yuang Kattan	Értekezlet. Döntések: Yin fogja csinálni 2.2.4 - 2.2.5 és 2.5 - 2.6 Bene fogja csinálni 2.3.1 - 2.3.4 Ji Jin Yan fogja csinálni 2.1.1 - 2.2.1 Ji Yuang fogja csinálni 2.4.1 - 2.4.2

			Kattan fogja csinálni 2.2.2 - 2.2.3 Csapatmunka platformok meghatározása: DC, Github, Google Docs
2025.02.14 19:00	0.2 óra	Yin	Tevékenység: Fedlap elkészítése
2025.02.14 19:00	0.3 óra	Ji Jin Yan	Tevékenység: Definíciók kigyűjtése, Hivatkozás kigyűjtése
2025.02.14 19:00	1 óra	Ji Jin Yan	Tevékenység: 2.1.1-2.1.3 megírása
2025.02.14 15:00	0.5óra	Ji Yuang	Tevékenység: Szöveg értelmezése és rajz elkészítése
2025.02.14 19:30	1 óra	Yin	Tevékenység: 2.2.4 és 2.2.5 megírása
2025.02.15 20:00	0.5 óra	Yin	Tevékenység: szótár szavak gyűjtése
2025.02.15 17:00	0.5 óra	Ji Yuang	Tevékenység: 2.4.1
2025.02.16 19:00	1 óra	Yin	Tevékenység: Szótár megírása
2025.02.17 12:00	0.5 óra	Ji Jin Yan	Tevékenység: 2.1.4-2.1.5.
2025.02.17 12:00	2 óra	Ji Yuang	Tevékenység: 2.4.1-2.4.2.
2025.02.17 13:00	3 óra	Bene	Tevékenység: 2.3.1, 2.3.2, 2.3.3, 2.3.4 vázlatos megírása
2025.02.17 18:00	1 óra	Ji Jin Yan	Tevékenység: 2.2.1
2025.02.17 20:00	2.5 óra	Kattan	Tevékenység: 2.2.2 (Funkciók) első változatának elkészítése
2025.02.19 16:00	3 óra	Bene	Tevékenység: 2.3.1, 2.3.2, 2.3.3, 2.3.4 részletes megírása
2025.02.19 20:00	0.1 óra	Yin	Tevékenység: Fedlap kiegészítése(csoport sorszám)
2025.02.20 19:00	0.5 óra	Yin	Tevékenység: Projekt terv kidolgozása
2025.02.20 20:00	1.5 óra	Yin Bene Ji Jin Yan Ji Yuang Kattan	Értekezlet. funkciók, és követelménynek részletes pontosítása
2025.02.20 22:00	1 óra	Bene	Tevékenység: 2.3.1 vázlatos kiegészítése az értekezleten megbeszéltekkel.
2025.02.20 23:00	1 óra	Kattan	Tevékenység: 2.2.2 átirása/finomítása az értekezlet alapján megbeszélt dolgokkal.

2025.02.21 10:00	2 óra	Bene	Tevékenység: 2.3.1 táblázat véglegesítése, tényleges megírása
2025.02.22 10:30	1 óra	Ji Yuang	Tevékenység:2.4.1
2025.02.22 10:30	0.2 óra	Ji Yuang	Tevékenység:2.4.1-2.4.2

3. Analízis modell kidolgozása

82 – being_chilling

Konzulens:
Tóth Bence

Csapattagok

Yin Jun	TN1KZI	1171276297yin@gmail.com
Bene Máté	K4C4W1	matthew.be04@gmail.com
Kattan Ibrahim Gábor	ZU22JJ	kattanbarhum2@gmail.com
Ji Jin Yan	ZGCKAL	jijinyan0201@gmail.com
Ji Yuang	UI7V2S	yuangji688@gmail.com

2025.03.02.

3. Analízis modell kidolgozása

3.1 Objektum katalógus

3.1.1 Rovar:

A Rovar egy mozgó entitás a játékban, amely a térképen elhelyezkedő Tekton egységeken mozoghat. Különböző hatások érhetik, például lassulás, gyorsulás, vagy bénulás. minden Rovart egy Rovarász irányít. Felelősségei:

- Mozgás
- Spórafogyasztás
- Gombafonalak vágása
- Effektusok kezelése
- Elérhető tektonok listázása

3.1.2 Rovarász:

Játékosok választhatják ezt a szerepet, rovarászok rovarászok ellen vannak. Felelőssége a rovar irányítása.

3.1.3 Gombász:

A Gombász egy olyan játékos, aki a játék során gombákat növeszthet, spórákat szórhat szét és gombafonalakat terjeszthet. Felelősségei:

- Gombák termesztése a térképen
- Gombák irányítása a spórák szórásában
- Gombafonalak növesztésének irányítása

3.1.4 Gomba:

A Gomba egy növényi entitás a játékban, amely spórákat szórhat és fonalakat növeszthet. Egy adott Gombász játékoshoz tartozik. Felelősségei:

- Spórák szórása a környező Tekton-okra.
- Gombafonalak növesztése a térképen.
- Saját növekedési ciklusának kezelése.

3.1.5 Spóra:

A spórák olyan játék elemek, amelyek tápanyagként szolgálnak a rovarok számára, és velük tudnak a gombatestek nőni. Speciális spórák különféle hatásokkal bírhatnak a rovarok számára, ezeket ők kezelik.

3.1.6 Gombafonal:

A Gombafonal a gombák növekedését és terjeszkedését segítő elem, amely összekapcsolja őket más Tektonokkal. Felelőssége a növekedés, terjeszkedés.

3.1.7 Tekton:

A tektonok a játékmezők alapelemei, a térképen egy cellát reprezentálnak. minden tekton felelőssége:

- Kettészakadás kezelése
- Szomszédos tektonok nyilvántartása
- Rajta lévő spórák, gomba, gombafonalak nyilvántartása
- (Ha van) Hatásuk kezelése (Mint pl. fonalszívó tektonnál.)

3.1.8 Térkép:

A Térkép a játék fő struktúrája, amely tárolja az összes Tekton-t és Játékos-t, és kezeli a játéktér állapotát. Felelősségei:

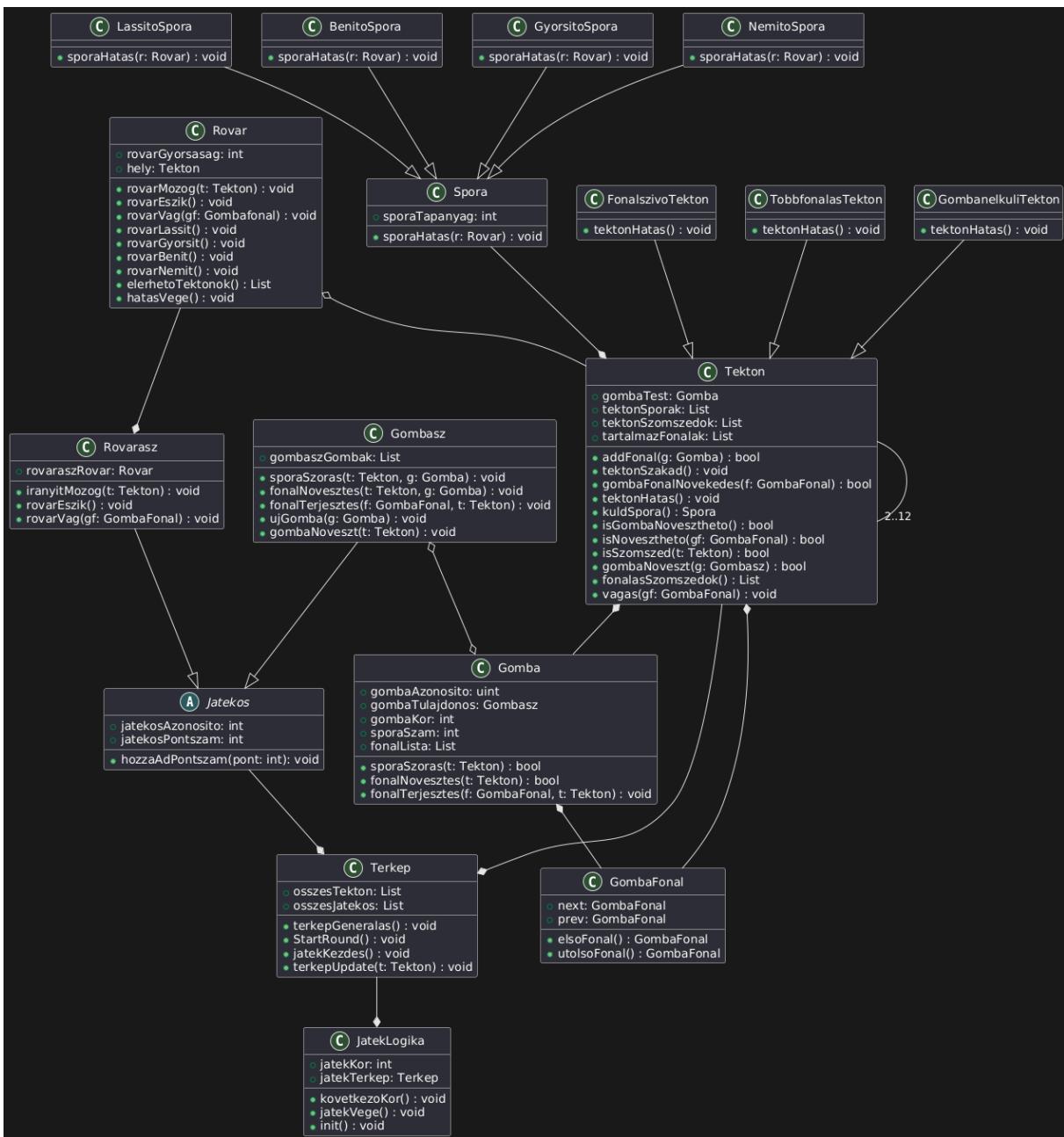
- A játékterület és annak elemeinek tárolása.
- A játékosok kezelésének biztosítása.
- A játékkörök indítása és a térkép frissítése.

3.1.9 JátékLogika:

A JátékLogika felelős a játékkörök vezérléséért, a játék állapotának frissítéséért és a végkimenetel kezeléséért. Felelősségei:

- A körök és a játékmenet irányítása.
- A játék befejezésének ellenőrzése.
- A játék inicializálása és kezelése.

3.2 Statikus struktúra diagramok



3.3 Osztályok leírása

3.3.1 BenitoSpora

- **Felelősség**
 - Az osztály a spórák megevése utáni hatásért, pontosabban a bénításért felel.
- **Ősosztályok**
 - Spora

- **Metódusok**
 - **void sporaHatas(Rovar r):** Az **r** Rovartól elveszi a vágás képességét kettő körre.

3.3.2 FonalszívóTekton

- **Felelősség**
 - Az osztály felelős a tektonon lévő gombafonal felszívódásáért. Ezt csak egy idő után teszi meg.
- **Ősosztályok**
 - Tekton
- **Metódusok**
 - **void tektonHatas():** A gombafonalat elpusztítja ami rajta van.

3.3.3 Gomba

- **Felelősség**
 - Az osztály arra vállal felelősséget, hogy spórát szórjon a szomszédos tektonok, illetve gombafonal növesztése a szomszéd tektonra.
- **Ősosztályok**
 - Interfész
 - Asszociációk
 - **Kompozíció:** A túloldali szereplő a Tekton osztály. A tektonon nőhet a gombatest, így a Tekton osztály tárolja a gombát.
 - **Aggregáció:** A túloldali szereplő a Gombasz osztály, a célja a kapcsolatnak az, hogy meghatározza a gomba osztály melyik Gombashoz tartozik.
- **Attribútumok**
 - **unsigned int gombaAzonosito:** Gomba megkülönböztetés.
 - **Gombasz gombaTulajdonos:** Megmondja ki a gombának a tulajdonosa.
 - **int gombaKor:** A gomba osztálynak a korra, pontosabban, hogy hány kört telt el a gomba létrehozása után.
 - **int sporaSzam:** Hány spóra szám van a Gombában.
 - **List fonalLista:** A gombafonal láncolt lista kezdő eleme.
- **Metódusok**
 - **bool sporaSzoras(Tekton t):** Megmondja, hogy a **t** Tektonra tudunk-e spórát szórni.
 - **bool fonalNovesztes(Tekton t):** Megmondja, hogy a **t** Tektonra tudunk-e gombafonalat növeszteni.
 - **void fonalTerjesztes(GombaFonal f, Tekton t):** Az **f** GombaFonal növesztése a **t** tektonra.

3.3.4 GombaFonal

- **Felelősség**
 - Egy láncolt lista, vagyis tárol egy előzőt és következőt, ami előtte, illetve utána van. Ez tárolja a gombatesthez tartozó fonalak egy ágát.
- **Asszociációk**
 - **Kompozíció:** Tekton a túloldali szereplő osztály. Célja a tektonon lévő fonalak tárolása.
 - **Kompozíció:** Gombasz osztály a túloldali szereplője. Célja, hogy a Gombasz az összes gombafonalat eltárolja.
- **Attribútumok**
 - **GombaFonal next:** következő gombafonal

- **GombaFonal prev:** előző gombafonál
- **Metódusok**
 - **GombaFonal elsoFonal()** : a gombafonál első elemét adja vissza.
 - **GombaFonal utolsoFonal()** : a gombafonál utolsó elemét adja vissza.

3.3.5 GombanelkuliTekton

- **Felelősség**
 - Olyan tekton, amin nem lehet gombafonalat növeszteni.
- **Ősosztályok**
 - Tekton
- **Metódusok**
 - **void tektonHatas():** a gomba a Tektonra növesztési képessége megszűnik.

3.3.6 Gombasz

- **Felelősség**
 - A Gombasz osztály felel a gombatestek és gombafonalak irányításáért. Tárolja a saját gombatestjeit, amelyek szükségesek a növesztés és szórás szempontjából.
- **Ősosztályok**
 - Jatekos
- **Attribútumok**
 - **List gombaszGombak:** a Gombász gombatestjeit tárolja.
- **Metódusok**
 - **void gombaNoveszt(Tekton t):** gombát növeszt az adott **t** tektonon
 - **void sporaSzoras(Tekton t, Gomba g):** **g** gomba spóráját szórja **t** tektonra.
 - **void fonalNovesztes(Tekton t):** **t** szomszédos tektonra fonalat növeszt
 - **void fonalNovesztes(GombaFonal f, Tekton t):** **f** gombafonalat növeszti tovább **t** tektonra.
 - **void ujGomba(Gomba g):** a gombát hozzáadja a listához.

3.3.7 GyorsitoSpora

- **Felelősség**
 - Az osztály a spórák megevése utáni hatásért, pontosabban a gyosításért felel.
- **Ősosztályok**
 - Spora
- **Metódusok**
 - **void sporaHatas(Rovar r):** Az **r** Rovar meggyorsítása, ami azt jelenti, hogy három tektonnyi távolságot mehet a kettő helyett egy kör alatt.

3.3.8 Jatekos

- **Felelősség**
 - Az absztrakt osztály, ami származtatja a Gombasz és a Rovaraszt osztályokat. Felelősséget vállal a játékos pontszámának kalkulálásáért.
- **Asszociációk**
 - **Kompozíció:** Térkép a túloldali osztály. A célja a kapcsolatnak az, hogy a térkép generálás során a játékos is együttesen hozzunk létre. Ahol a térkép tárolja a játékosokat.

- **Attribútumok**
 - **int jatekAzonosito:** A játékosokat megkülönböztetjük az azonosító alapján.
 - **int jatekAontszam:** Megszámolja, hogy egy játéknak mennyi a pontszáma.
- **Metódusok**
 - **int hozzaAdPontszám(int pont):** hozzáad **pont** pontszámot, miután szerzett pontot a játékos.

3.3.9 JatekLogika

- **Felelősség**
 - Felelős a játéknak az elkezdése, futtatása, és befejezése, illetve a játékban szerepelt osztályok együttműködése.
- **Asszociációk**
 - **Kompozíció:** A Térkép a túloldali osztály. A célja a kapcsolatnak, hogy a játékLogika futtatás során képbe legyen a Térkép helyzetének, hogy sikerüljön a futtatás a térkép osztály szerint.
- **Attribútumok**
 - **int JatekKor :** megmondja hogy hányadik játékkörnél tart a játék.
 - **Terkep jatekTerkep:** Egy terkep osztály, ami eszerint fog a játéklogika működni.
- **Metódusok**
 - **void kovetkezoKor():** következő kör indítása.
 - **void jatekVege():** a játéknek a bejelezési feltétele, és játékból kilépése.
 - **void init():** Játék elindítása.

3.3.10 LassitoSpora

- **Felelősség**
 - Az osztály a spórák megevése utáni hatásért, pontosabban a lassításért felel.
- **Ősosztályok**
 - Spora
- **Metódusok**
 - **void sporaHatas(Rovar r):** Az **r** Rovar lelassítása, ami azt jelenti, hogy egy tektonnyi távolságot mehet a kettő helyett egy kör alatt.

3.3.11 NemitoSpora

- **Felelősség**
 - Az osztály a spórák megevése utáni hatásért, pontosabban a némitásért felel.
- **Ősosztályok**
 - Spora
- **Metódusok**
 - **void sporaHatas(Rovar r):** Az **r** Rovar elnémitása, ami azt jelenti, hogy a Rovar kettő körig nem ehet, így nem tud tápanyaghoz jutni.

3.3.12 Rovar

- **Felelősség**
 - Felel a Rovar mozgásáért, táplálkozásáért, fonal vágásáért.

- **Asszociációk**
 - **Kompozíció:** Rovarasz a túloldali szereplő osztály. A rovar létezése csak Rovaraszon keresztül lehet, mert Rovarasz irányítja a rovarnak a mozgását és nélküle nem lehetne semmit se csinálni a Rovarral.
- **Attribútumok**
 - **int rovarGyorsassag:** megmondja hogy mekkora sebességgel rendelkezik a rovar, pontosabban azt, hogy mennyi tektonon tud átmenni egy kör alatt.
 - **Tekton hely:** a Rovar ezen a tektonon található.
- **Metódusok**
 - **void rovarMozog(Tekton: t)**: Rovar átmozgatás **t** Tektonról a **t** szomszédos Tektronrra.
 - **void rovarEszik()**: Rovar megeszik s Spora-t.
 - **void rovarVag(Tekton: t)**: Rovar elvágja a **t** Tektonon lévő gombaFonalat.
 - **void rovarLassit()**: Rovar a benitoSpora megevése után lelassul. Pontosabban, egy tektonnyi távolságot tesz meg kettő helyett egy kör alatt.
 - **void rovarGyorsit()**: Rovar a gyorsitoSpora megevése után felgyorsul. Pontosabban, három tektonnyi távolságot tesz meg kettő helyett egy kör alatt.
 - **void rovarNemit()**: Rovar a nemitoSpora megevése után elnemít. Pontosabban, a rovar kettő körön keresztül nem tud tápanyaghoz jutni.
 - **void rovarBemit()**: Rovar a nemitoSpora megevése után lebénül. Pontosabban, a rovar egy kört kihagy a játékból.
 - **List elerhetoTektonok()**: A Rovarnak elérhető Tektonok megadása.
 - **void hatasVege()**: Az adott hatás véget ér, ami a Rovaron volt.

3.3.13 Rovarasz

- **Felelősség**
 - A rovar irányításáért felel. Ő mozgatja a rovart, adja tovább a megteendő lépéseket.
- **Ősosztályok**
 - Jatekos
- **Attribútumok**
 - **Rovar rovaraszRovar:** a rovarász rovara, amit irányít.
- **Metódusok**
 - **void iranyitMozog(Tekton t):** a Rovart, amit irányít mozgatja a **t** tektonra.
 - **void rovarEszik()**: az meglévő tektonon megeszik egy spórát.
 - **void rovarVag(GombaFonal gj):** a rovar elvágja **gj** gombafonalat.

3.3.14 Spora

- **Felelősség**
 - Tápanyag kibocsátásáért vállal felelősséget, illetve az új gomba növesztésének a feltétele.
- **Attribútumok**
 - **int sporaTapanyag:** Adott tápanyag tartalom kínáltatás.
- **Metódusok**
 - **void sporaHatas(Rovar: r):** különböző féle Sporák, amik különböző hatással rendelkeznek **r** rovarok számára.

3.3.15 Tekton

- **Felelősség**

- Az osztály a Gomba növesztése ért, a Fonalak és Sporák tartalmazása, illetve Tekton kettétörésért vállal felelősséget.

- **Asszociációk**

- **Kompozíció:** A Gomba a túloldali szereplő osztály. A cél a kapcsolatnak, hogy a Gomba növesztést azt kizárolag a Tektonon tehető meg.
- **Kompozíció:** A Sporák a túloldali szereplő osztály. A cél a kapcsolatnak, hogy a Sporát kivételesen csak Tekton tartalmazhat.
- **Kompozíció:** A GombaFonal a túloldali szereplő osztály. A cél a kapcsolatnak, hogy a GombaFonalat kivételesen csak Tekton tartalmazhat.
- **Kompozíció:** A Terkep a túloldali szereplő osztály. A cél a kapcsolatnak, hogy a Terkep létrehozása esetén szükséges a Tekton, és ha Terkepet töröljük, akkor a Terkepen szereplő Tektonok is eltűnnek.
- **Aggregáció:** A rovar a túloldali szereplő osztály. A kapcsolat célja, hogy a Rovar tudja melyik tektonon tartózkodik éppen.

- **Attribútumok**

- **Gomba gombaTest:** A hozzáartozó Gomba
- **List tektonSporak:** A hozzáartozó Sporák felismerése
- **List tektonSzomszedok:** Melyik Tektonokkal van szomszédban.
- **List tartalmazFonalal:** TöbbfonalasTekton esetén, meghatározza, hogy mely Gombafonalakat tartalmazza a Tekton, egyébként csak 1 fonalat tartalmaz.

- **Metódusok**

- **Bool addFonal(Gomba: g):** Eldönti, hogy a **g** Gombára tud-e több Gombafonalat növeszteni.
- **void tektonSzakad():** A Tekton egy eltelt idő után kettére törik.
- **Bool gombaFonalNovekedese(GombaFonal f):** Eldönti, hogy az **f** Gombafonalra tud-e tovább Gombafonalat növeszteni.
- **void tektonHatas():** Különböző Tektonok különböző hatással rendelkeznek.
- **Spora kuldsSpora():** Tud-e erre a Tektonra Sporát küldeni, ha igen akkor a küldhető Sporára tér vissza, ha nem akkor null.
- **Bool isGombaNovesztheto():** Eldönti, hogy gomba növeszthető e a tektonon.
- **Bool isNovesztheto(GombaFonal f):** Gombafonal növeszthető e a tektonon.
- **Bool isSzomszed(Tekton t):** Eldönti, hogy a **g** Gombának szomszéd-e a Tekton.
- **Bool gombaNoveszt(Gombasz g):** Eldönti, hogy a **g** Gombasz tud-e erre a Tektonra Gombát növeszteni.
- **List fonalasSzomszedok():** Meghatározza, hogy milyen FonalasTektonnal van szomszédban.
- **void vagas(GombaFonal gf):** elvágja a tektonon található gombafonalat.

3.3.16 Terkep

- **Felelősség**

- Az osztály a Tekton tárolásáért, illetve a Jatekos bevonásáért felelős.

- **Asszociációk**

- **Egyszerű asszociáció:** A JatekLogika a túloldali szereplő osztály. A célja a kapcsolatnak, hogy a JatekLogika tudjon a Terkephez szerint szinkronizálni.

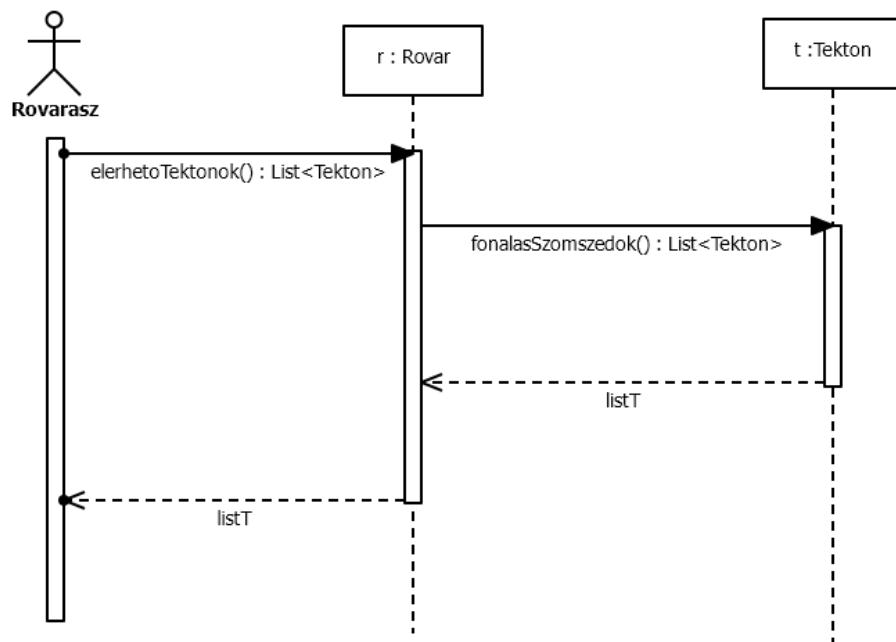
- **Kompozíció:** A Tekton a túloldali szereplő osztály. A célja a kapcsolatnak az, hogy a Tektonokat a Terkepben tarolja, és lehessen a Terkepben a Tektonokat módosítani. A Terkep elvesztés után a Tektonokat is elveszítenék.
- **Kompozíció:** A Jatekos a túloldali szereplő osztály. A célja a kapcsolatnak az, hogy Terkepben szerepeljenek a Jatekosok, amely saját objektumokat irányítanak a Teknokon keresztül. A Terkep elvesztés után a Jatekosokat is elveszítenék.
- **Attribútumok**
 - **List_osszesTekton:** Az összes Tekton eltárolása.
 - **List_osszesJatekos:** Az összes Játékos eltárolása.
- **Metódusok**
 - **void terkepGeneralas():** Terkep generálás, az attribútuma szerint, azaz a Tektonok és Játékosok alapján generálunk térképet.
 - **void startRound():** A Kör elindítása.
 - **void jatekKezdese():** A Játék indítása.
 - **void terkepUpdate(Tekton: t):** Az új állapot szerint, azaz t Tekton változtatás után, Térkép frissítése.

3.3.17 TobbfonalasTekton

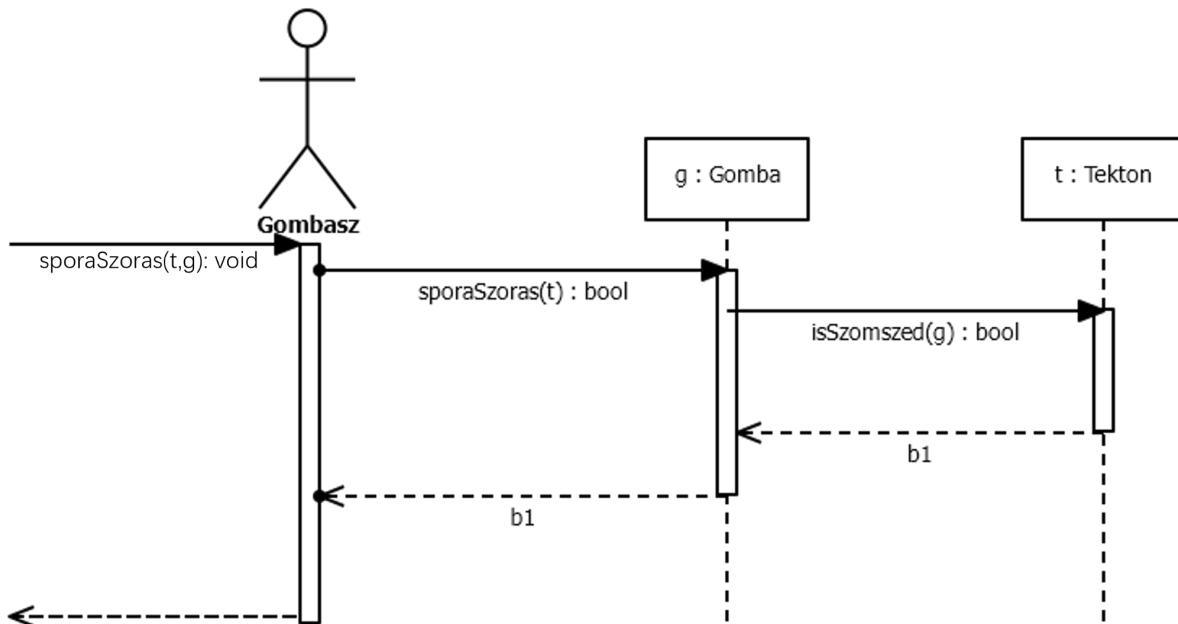
- **Felelősség**
 - Olyan Tekton, ami felelős a több GombaFonal növesztésének engedélyezéséért.
- **Ősosztályok**
 - Tekton
- **Metódusok**
 - **void tektonHatas():** A Tektonra több GombaFonal növesztése képes.

3.4 Szekvencia diagramok

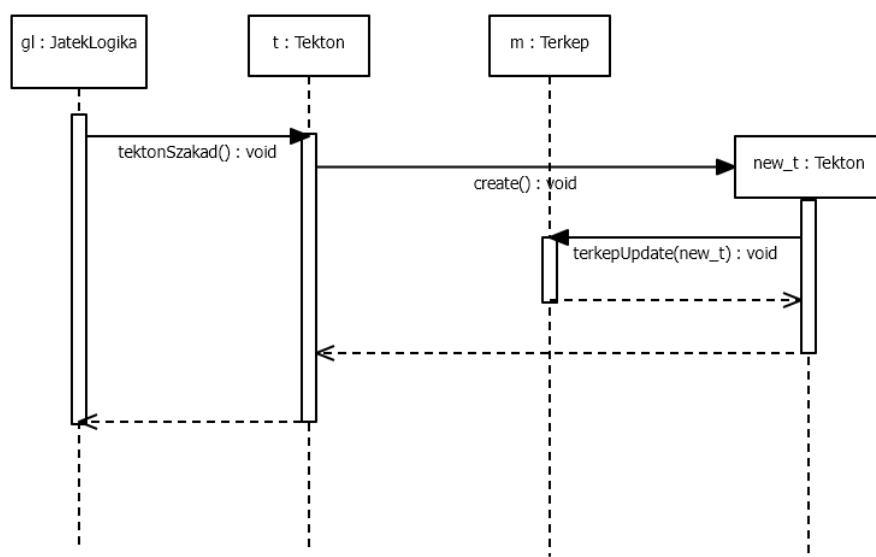
sd Rovarok irányítása



sd Spóra szórás



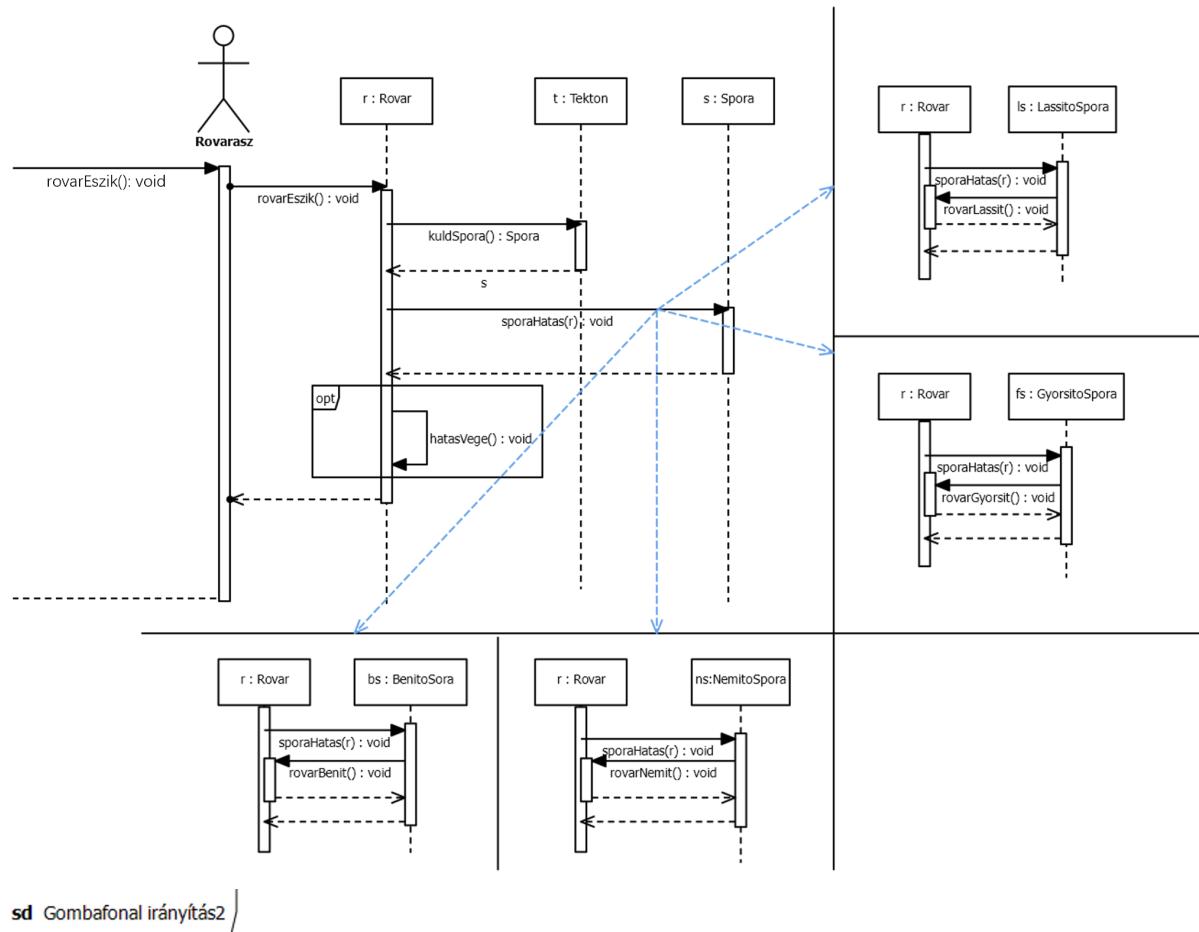
sd Tektonok kettétörése



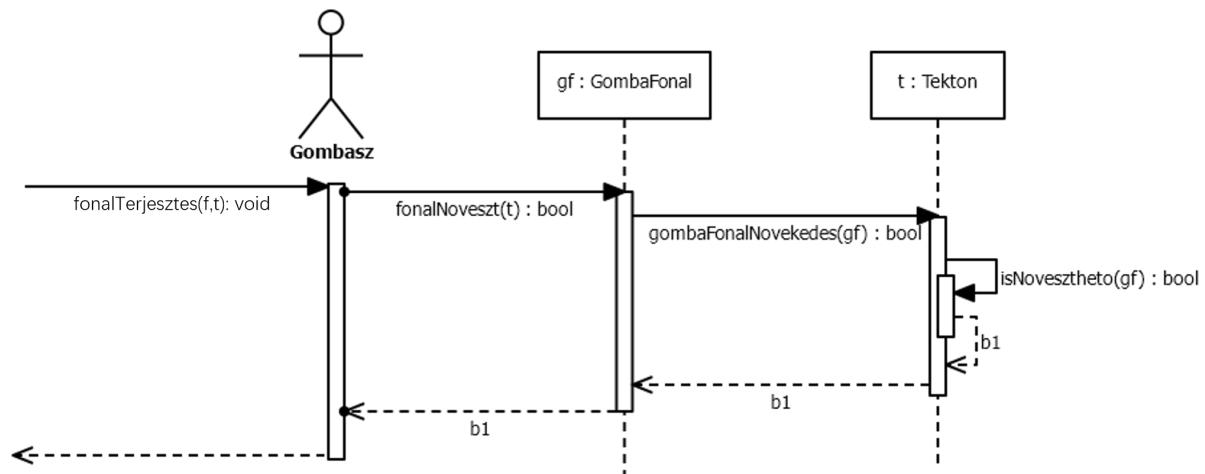
3. Analízis modell kidolgozása

Being_chilling

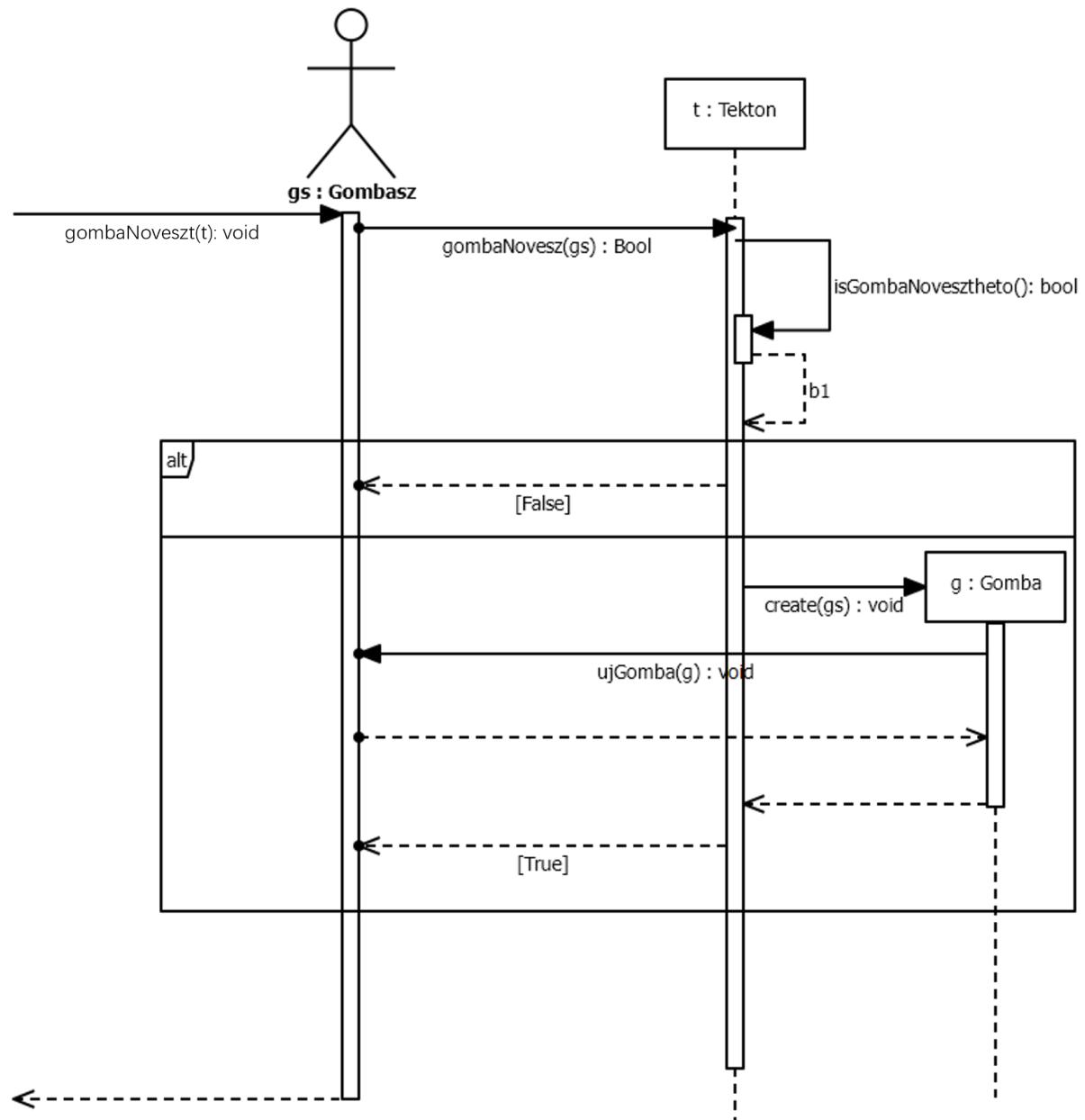
sd Rovarok táplálkozása

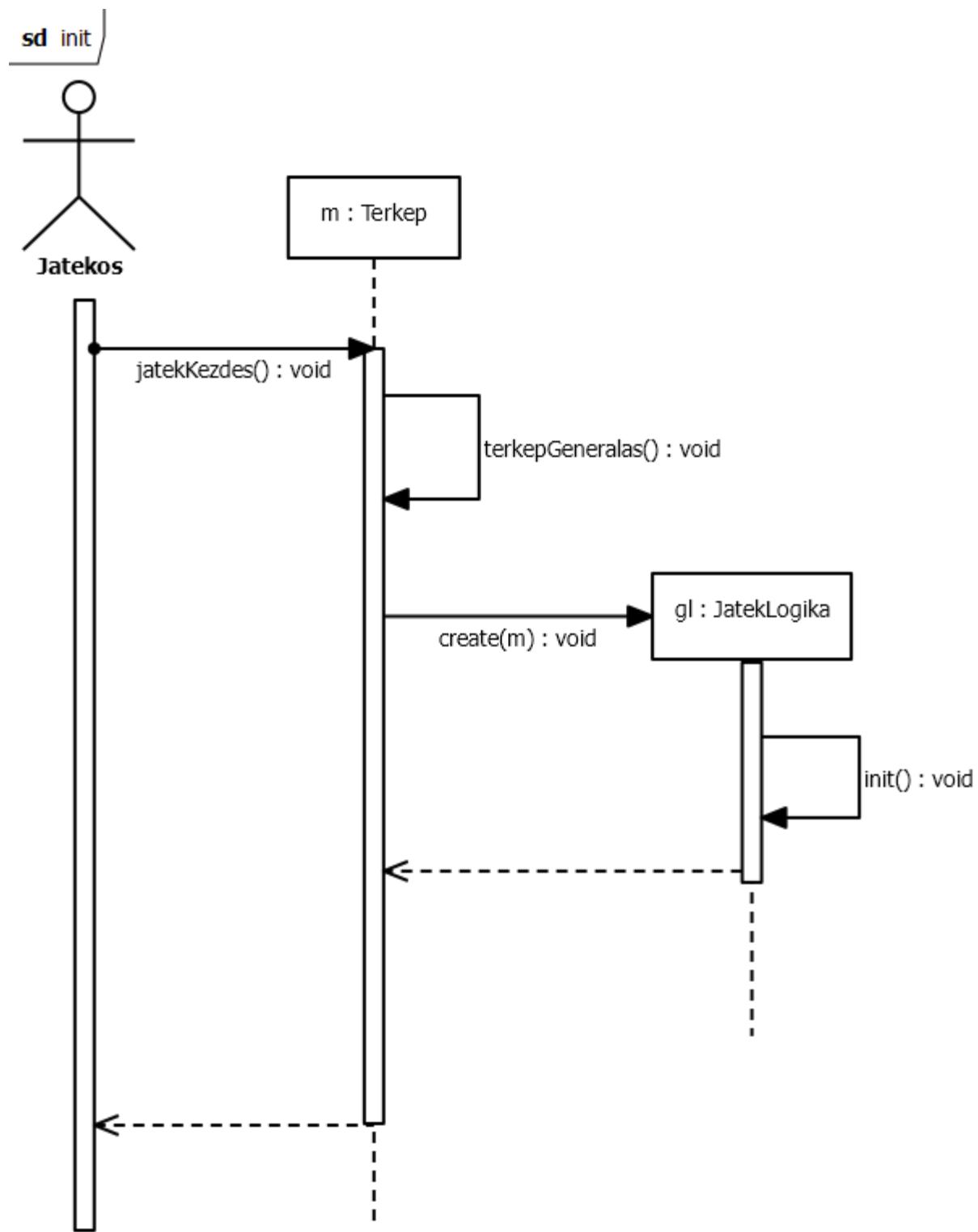


sd Gombafonal irányítás2

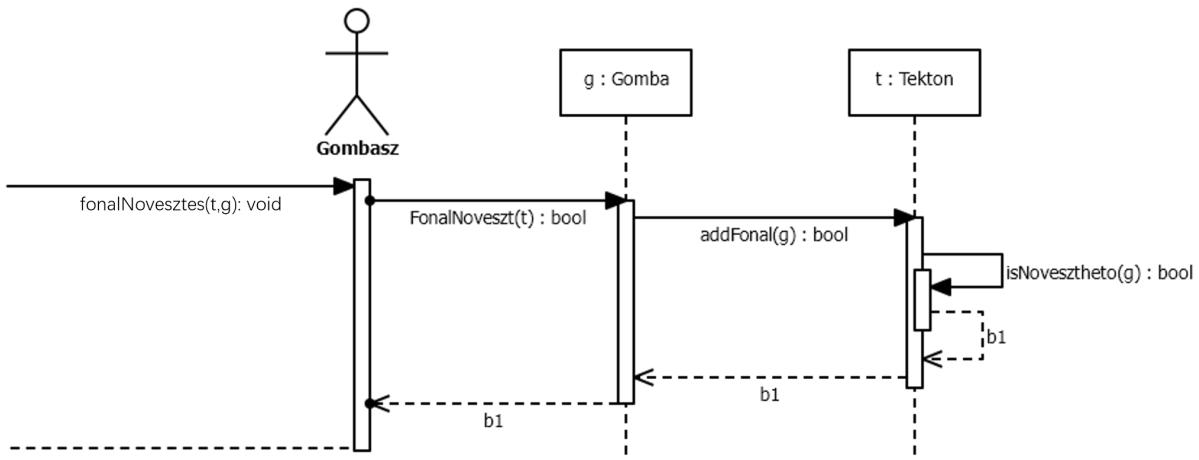


sd Új gombatest növesztése

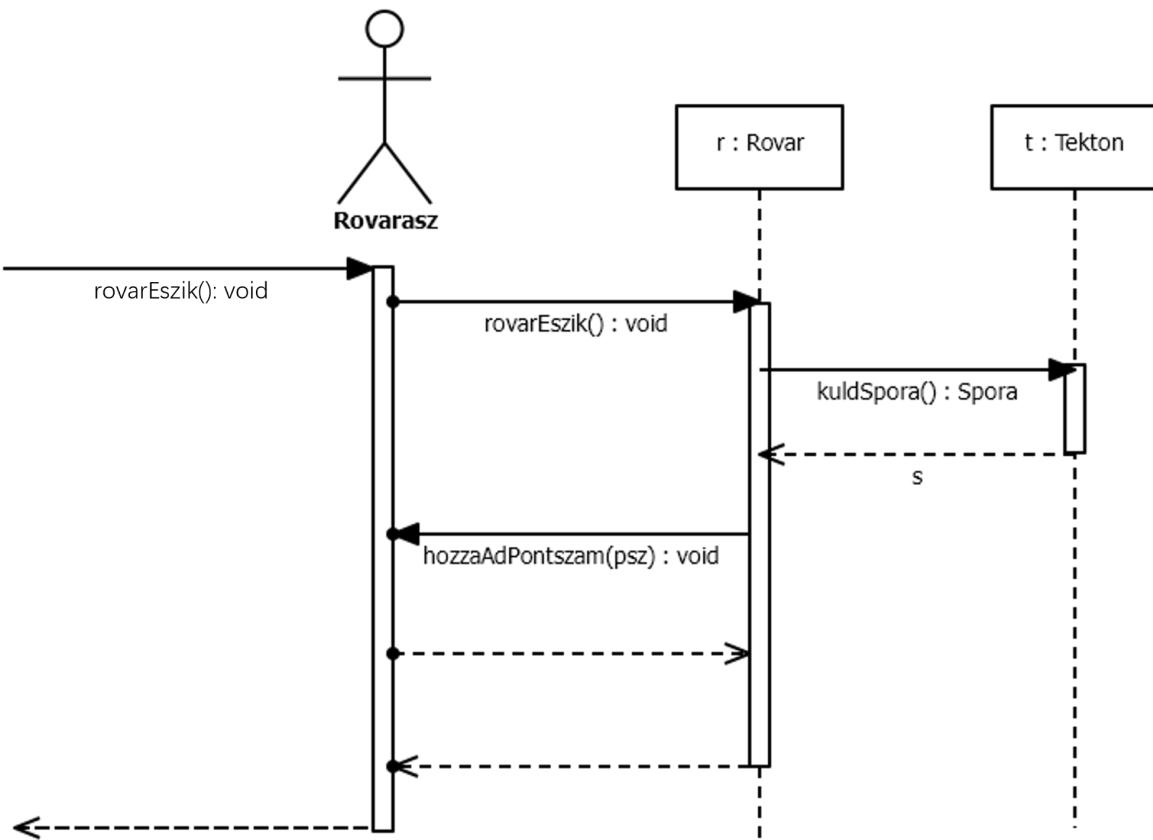




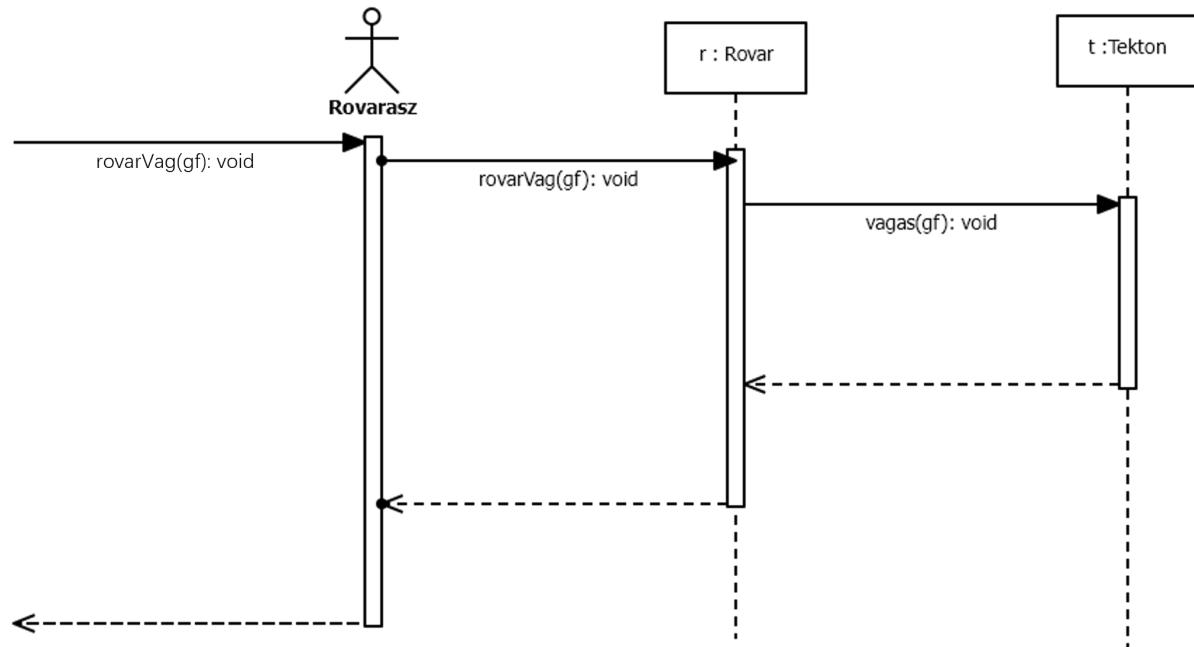
act Gombafonal irányítás



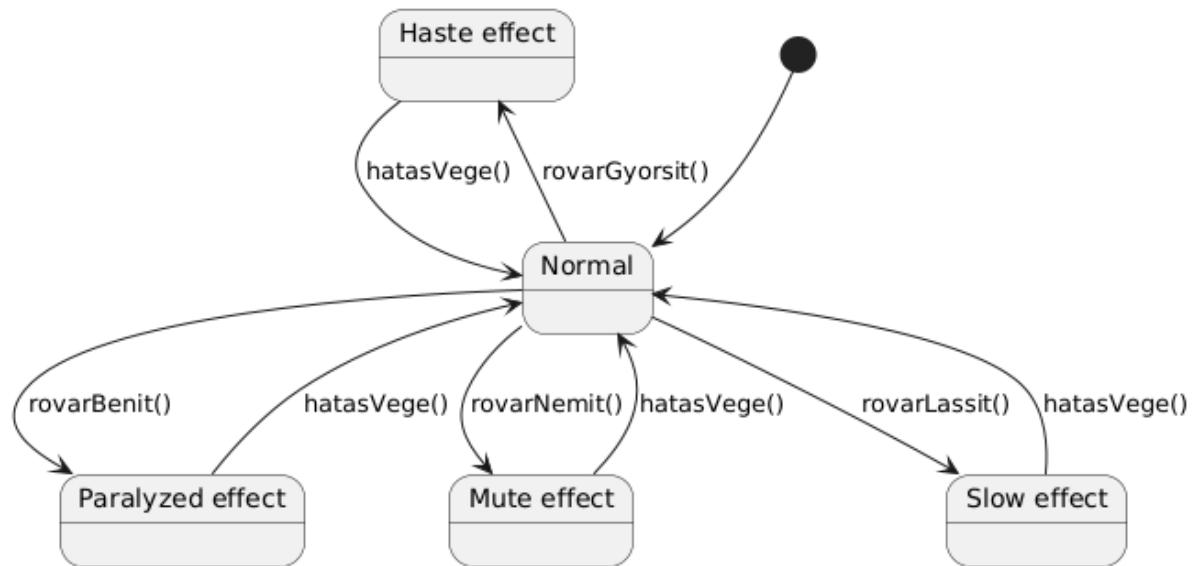
sd Rovar eszik



sd Rovar Vágás



3.5 State-chartok



3.6 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2025.02.26 19:30	2 óra	Yin Bene Kattan Ji Ji Yan Ji Yuang	Értekezlet. Döntés: Munka beosztás és class diagram részlet megbeszélés
2025.02.27 16:30	0.5 óra	Ji Jin Yan	Tevékenység: 3.1 Elkezdés, átgondolás
2025.02.27 15:00	4 óra	Kattan Ji Jin Yan	Értekezlet. 3.2 Átbeszélés
2025.02.28 10:00	2 óra	Yin	Tevékenység: Szekvencia diagramok készítése
2025.03.01 20:30	1 óra	Yin Bene	Értekezlet. Döntés: Szekvencia diagramok megbeszélés
2025.03.02 01:00	3 óra	Kattan	Tevékenység: Objektum Katalógus és Class Diagram elkészítése
2025.03.02 8:30	2 óra	Bene Ji Yuang	Értekezlet. Osztályok leírása
2025.03.02 16:00	3 óra	Bene	Tevékenység: 3.3 megfogalmazása
2025.03.02 13:00	2 óra	Ji Yuang	Tevékenység: Osztályok leírása
2025.03.02 20:00	0.5 óra	Yin	Tevékenység: State-chart elkészítése
2025.03.02 22:00	3 óra	Bene Yin	Értekezlet. Class Diagram és Szekvencia diagramok megbeszélés
2025.03.03 01:00	3 óra	Bene	Tevékenység: 3.2-3.3 javítása
2025.03.03 01:00	3 óra	Yin	Tevékenység: Szekvencia diagramok javítása

4. Analízis modell (II. változat)

82 – being_chilling

Konzulens:
Tóth Bence

Csapattagok

Yin Jun	TN1KZI	1171276297yin@gmail.com
Bene Máté	K4C4W1	matthew.be04@gmail.com
Kattan Ibrahim Gábor	ZU22JJ	kattanbarhum2@gmail.com
Ji Jin Yan	ZGCKAL	jijinyan0201@gmail.com
Ji Yuang	UI7V2S	yuangji688@gmail.com

2025.03.09.

4. Analízis modell kidolgozása

4.1 Objektum katalógus

4.1.1 Rovar:

A Rovar egy mozgó entitás a játékban, amely a térképen elhelyezkedő Tekton egységeken mozoghat. Különböző hatások érhetik, például lassulás, gyorsulás, vagy bénulás.
Felelősségei:

- Mozgás
- Spórafogyasztás
- Gombafonalak vágása
- Effektusok kezelése
- Elérhető tektonok listázása

4.1.2 Gombatest:

A Gombatest egy növényi entitás a játékban, amely spórákat szórhat és fonalakat növeszthet.
Felelősségei:

- Spórák szórása a környező Tekton-okra.
- Meglévő gombafonalak hosszabbítása.
- Új gombafonalat növeszteni.
- Szórható spórák mennyiség kezelése.

4.1.3 Normál spóra:

A normál spóra olyan játék elem, amely tápanyagként szolgál a rovarok számára, és vele tudnak a gombatestek nőni.

4.1.4 Bénító spóra:

A bénító spóra olyan játék elem, amely tápanyagként szolgál a rovarok számára, és vele tudnak a gombatestek nőni. Ha rovar elfogyasztja ezt a spórát, akkor bénító hatást kap, ami azt jelenti, hogy 2 körig nem vághat fonalat.

Felelőssége:

- Bénító effektus kezelése

4.1.5 Gyorsító spóra:

A gyorsító spóra olyan játék elem, amely tápanyagként szolgál a rovarok számára, és vele tudnak a gombatestek nőni. Ha rovar elfogyasztja ezt a spórát, akkor gyorsító hatást kap, ami azt jelenti, hogy 1 kör alatt 3 tektonnyi távolságot léphet.

Felelőssége:

- Gyorsító effektus kezelése

4.1.6 Némító spóra:

A némító spóra olyan játék elem, amely tápanyagként szolgál a rovarok számára, és vele tudnak a gombatestek nőni. Ha rovar elfogyasztja ezt a spórát, akkor némító hatást kap, ami azt jelenti, hogy 2 körig nem fogyaszthat spórát.

Felelőssége:

- Némító effektus kezelése

4.1.7 Lassító spóra:

A lassító spóra olyan játék elem, amely tápanyagként szolgál a rovarok számára, és vele tudnak a gombatestek nőni. Ha rovar elfogyasztja ezt a spórát, akkor lassító hatást kap, ami azt jelenti, hogy a következő körében csak 1 tektonnyi távolságot léphet.

Felelőssége:

- Lassító effektus kezelése

4.1.8 Gombafonal:

A Gombafonal a gombák növekedését és terjeszkedését segítő elem, amely összekapcsolja őket más Tektonokkal.

Felelősségei:

- Gombafonal növekedés, terjeszkedés.
- Gombafonal eltávolítás
- Felgyorsítás kezelés

4.1.9 Tekton:

A tektonok a játékmezők alapelemei, a térképen egy cellát reprezentálnak.

Felelőssége:

- Kettészakadás kezelése
- Szomszédos tektonok nyilvántartása
- Rajta lévő spórák, gombatest, gombafonal nyilvántartása

4.1.10 Fonalszívó tekton:

A fonalszívó tektonok a játékmezők alapelemei, a térképen egy cellát reprezentálnak. A fonalszító tektonon 1 fonal nőhet, de felszívódik 5 kör után.

Felelőssége:

- Kettészakadás kezelése
- Szomszédos tektonok nyilvántartása
- Rajta lévő spórák, gombatest, gombafonal nyilvántartása
- Felszívás kezelés

4.1.11 Többfonalas tekton:

A tektonok a játékmezők alapelemei, a térképen egy cellát reprezentálnak. A többfonalas tektonon 5 fonal nőhet.

Felelőssége:

- Kettészakadás kezelése
- Szomszédos tektonok nyilvántartása
- Rajta lévő spórák, gombatest, gombafonalak nyilvántartása

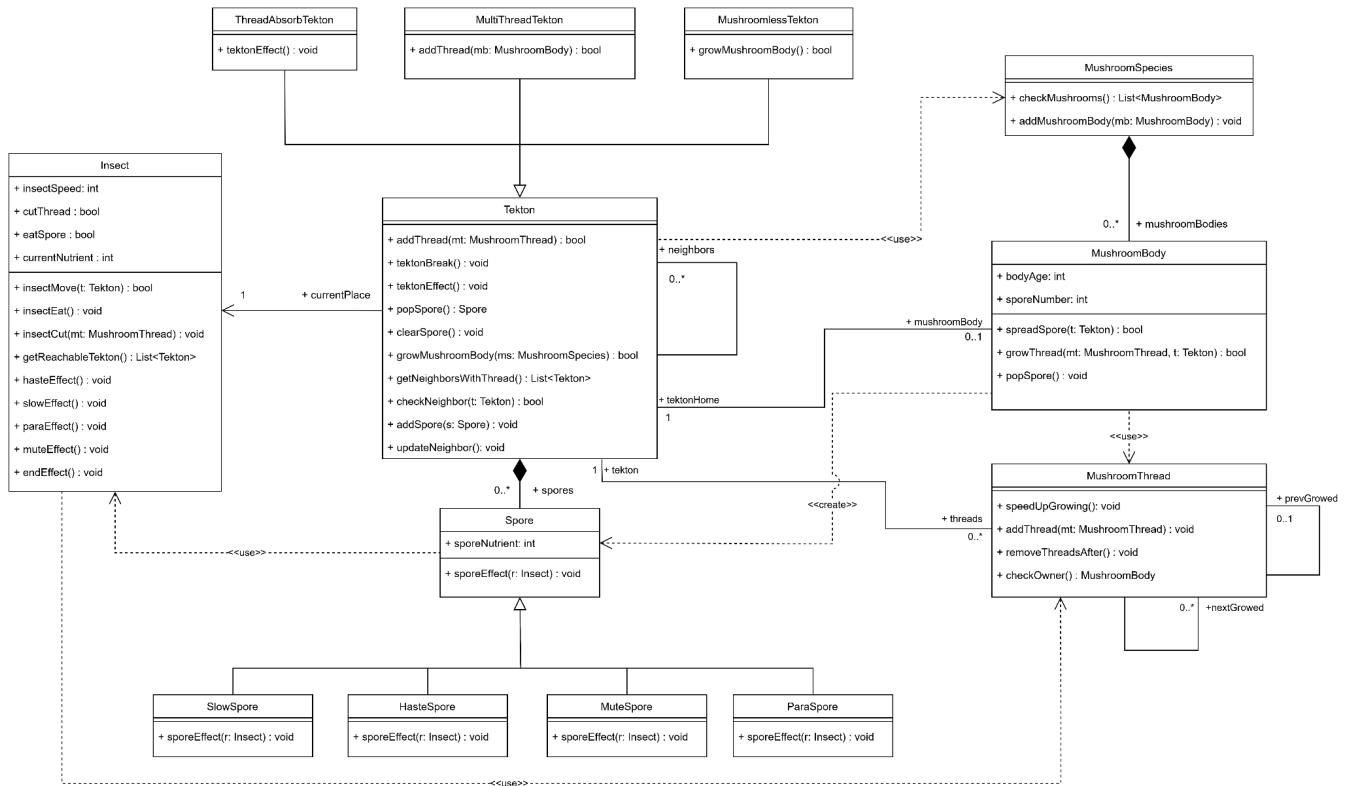
4.1.12 Gombanélküli tekton:

A tektonok a játékmezők alapelemei, a térképen egy cellát reprezentálnak. A gombanélküli tektonon, 1 fonal nőhet, és nem nőhet gomba.

Felelőssége:

- Kettészakadás kezelése
- Szomszédos tektonok nyilvántartása
- Rajta lévő spórák, gombatest, gombafonal nyilvántartása

4.2 Statikus struktúra diagramok



4.3 Osztályok leírása

4.3.1. HasteSpore

- **Felelősség**
 - Az osztály a gyorsító spórát valósítja meg.
- **Ősosztályok**
 - Spore
- **Metódusok**
 - **void sporeEffect(Insect r):** Felgyorsítja az adott rovart.

4.3.2. Insect

- **Felelősség**
 - Az osztály felelőssége a rovarászok által irányított rovarok funkcióinak megvalósítása.
- **Asszociációk**
 - **Egyszerű asszociáció (irányított):**
 - **Tekton:** A rovar eltárolja a pozíóját (vagyis a tektont).
 - **Függőség:**
 - **MushroomThread:** A rovar azonosítja, hogy mely fonalat kell elvágna
- **Attribútumok**
 - **int insectSpeed:** Ez a változó mutatja meg, hogy hány tektonnyit mozoghat a rovar egy kör alatt.
 - **bool cutThread:** Ez a változó mutatja meg, hogy vághat-e fonalat a rovar vagy sem.

- **bool eatSpore:** Ez a változó mutatja meg, hogy ehet-e spórát a rovar vagy sem.
- **int currentNutrient:** Mennyi tápanyagot evett meg a rovar.
- **Metódusok**
 - **bool insectMove(Tekton t):** Egy adott tektonra mozgatja a rovart.
 - **void insectEat():** Megesz egy, a tektonján elhelyezett spórát.
 - **void insectCut(MushroomThread mt):** Rovar elvágja az adott gombafonalat.
 - **List<Tekton> getReachableTekton():** Visszaadja az összes, gombafonalakkal elérhető Tektont.
 - **void hasteEffect():** Felgyorsítja a rovart, gyorsító spóra megevése után.
 - **void slowEffect():** Lelassítja a rovart, lassító spóra megevése után.
 - **void paraEffect():** Megbénítja a rovart, bénító spóra megevése után. A bénítás hatása alatt azt értjük, hogy a rovar nem ehet, nem mozoghat és nem vághat gombafonalat.
 - **void muteEffect():** Elnémítja a rovart, némitő spóra megevése után. A némitás hatás alatt azt értjük hogy a rovar kettő körig nem vághat.
 - **void endEffect():** Ez a metódus felelős arra, hogy a spóra hatását levegye a rovarról az időtartam lefutása után.

4.3.3. MultiThreadTekton

- **Felelősség**
 - Az osztály a többfonalas tekton megvalósítása, amely egy specializált tekton. A többfonalas tektonon több játékos fonala is tartózkodhat egyszerre.
- **Ősosztályok**
 - Tekton
- **Metódusok**
 - **void addThread(MushroomThread mt):** Felülírja a tekton metódusát, mivel itt több fonal is növekedhet.

4.3.4. MushroomBody

- **Felelősség**
 - Az osztály a Tektikon lévő gombatestek funkcióit valósítja meg.
- **Attribútumok**
 - **int bodyAge:** Megmondja, hogy hány kör óta él a gombatest. Fontos tudni, mivel a gombatestek egy idő után fejlődnek.
 - **int sporeNumber:** Megmondja, hogy hány spórája maradt a gombatestnek.
- **Asszociációk**
 - **Függőség:**
 - Spore: Ez az osztály hozza létre a spórát és adja át a tektonnak.
 - MushroomThread: Paraméterként veszi át az osztály több függvénye a MushroomThread-et, ez azonosítja a fonalat a játékosokkal.
 - **Egyeszerű asszociáció:**
 - **Tekton:** Osztály eltárolja, hogy melyik tektonon tartózkodik.
- **Metódusok**
 - **bool spreadSpore(Tekton t):** Spórát szór egy adott tektonra.
 - **bool growThread(MushroomThread mt, Tekton t):** Egy adott tektonra növeszt gombafonalat egy gombatest, attól függően, hogy van-e a tekton szomszédján ugyanannak a gombatestnek fonala. Ha nem a gombafonal végét növeszti, akkor egy új ágat hoz létre arra a tektonra.
 - **void popSpore():** Csökkenti a spórának számát eggyel.

4.3.5. MushroomlessTekton

- **Felelősség**
 - Az osztály a gombánélküli tekton megvalósítása, amely egy specializált tekton. A tekton speciális tulajdonsága, hogy nem nőhet rajta gombatest.
- **Ősosztályok**
 - Tekton
- **Metódusok**
 - **bool growMushroomBody():** Felülírja a tekton metódusát: nem csinál semmit, csak false-t ad vissza, mivel nem nőhet gombatest az ilyen tektonokon.

4.3.6. MushroomSpecies

- **Felelősség**
 - Az osztály egy gombafaj összes gombatest példányát tárolja el.
- **Asszociációk**
 - **Kompozíció:**
 - **MushroomBody:** Nyilvántartja, egy gombafaj összes létező egyedét.
- **Metódusok**
 - **List<MushroomBody> checkMushrooms():** Visszaadja az összes létező példányt.
 - **void addMushroomBody(MushroomBody mb):** Felvesz egy új gombatestet.

4.3.7. MushroomThread

- **Felelősség**
 - Az osztály a gombatestek által létrehozott gombafonalak funkciójait valósítja meg.
- **Asszociációk**
 - **Egyszerű asszociáció:**
 - Tekton: Eltárolja melyik tektonon van a gombafonal.
 - MushroomThread: Eltárolja az előtte lévő gombafonalat.
 - MushroomThread: Eltárolja az utána lévő gombafonalakat.
- **Metódusok**
 - **void speedUpGrowing():** Ha fonal tektonjára spóra kerül, akkor egy ideig magától hosszabulni fog a fonal.
 - **void addThread(MushroomThread mt):** A listához hozzáadja az adott fonalat.
 - **void removeThreadsAfter():** A fonal elvágása után, törli az utána lévő fonalakat.
 - **MushroomBody checkOwner():** Megnézi, hogy melyik gombatesthez tartozik a jelen lévő gombafonal.

4.3.8. MuteSpore

- **Felelősség**
 - Az osztály a némító spórát valósítja meg.
- **Ősosztályok**
 - Spore
- **Metódusok**
 - **void sporeEffect(Insect r):** Lenémítja az adott rovart.

4.3.9. ParaSpore

- **Felelősség**
 - Az osztály a bénító spórát valósítja meg.
- **Ősosztályok**
 - Spore
- **Metódusok**
 - **void sporeEffect(Insect r):** Megbénítja az adott rovart.

4.3.10. SlowSpore

- **Felelősség**
 - Az osztály a lassító spórát valósítja meg.
- **Ősosztályok**
 - Spore
- **Metódusok**
 - **void sporeEffect(Insect r):** Lelassítja az adott rovart.

4.3.11. Spore

- **Felelősség**
 - Egy szórásnyi spóra hatásaiért és tulajdonságaiért felelős osztály.
- **Asszociációk**
 - **Függőség:**
 - **Insect:** Ez azonosítja, hogy melyik rovarra kell a spóra hatását biztosítani.
- **Attribútumok**
 - **int sporeNutrient:** Eltárolja, hogy mennyi tápanyagot tartalmaz egy szórásnyi spóra.
- **Metódusok**
 - **void sporeEffect(Insect r):** A leszármazó osztályok a hatásait ezzel a metódussal valósítják meg.

4.3.12. Tekton

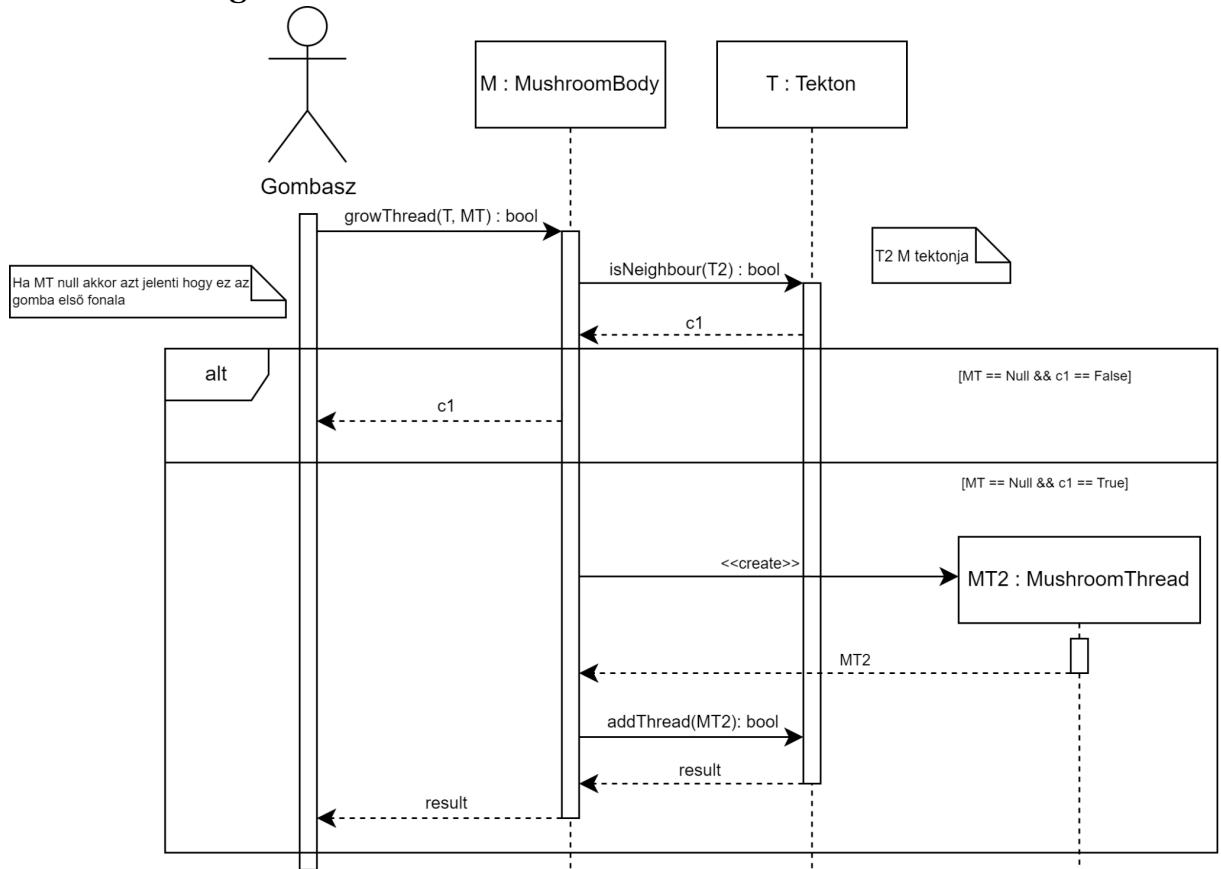
- **Felelősség**
 - Az osztály a játék alapmezőjét valósítja meg, a tektont. Legfőbb felelőssége a játéktérrel kapcsolatos, de fonalak és spórák működését is elősegíti.
- **Metódusok**
 - **void addThread(MushroomThread mt):** Egy játékos a saját fonalát a tektonon keresztül terjesztheti. Ez a metódus hozzáadja a fonalat a tektonhoz.
 - **void tektonBreak():** Ez a metódus a tekton félbetörését valósítja meg. Olyan tekton nemtörhet ketté, amin rovar, vagy gombatest van. A tektonon lévő fonalak elszakadnak, spórák eltűnnék.
 - **void tektonEffect():** Ez a metódus a fonalszívó tekton számára létrehozott, a Tekton osztályban nincs hatása.
 - **void addSpore(Spore s):** Ez a metódus hívódik meg, ha egy gombatest spórát szór a tektonra, és hozzáad egy spórát a tektonhoz.
 - **bool growMushroomBody(MushroomSpecies ms):** Ez a metódus egy gombafaj gombatestét növeszti a tektonra. Csak egy gombatest lehet tektononként.

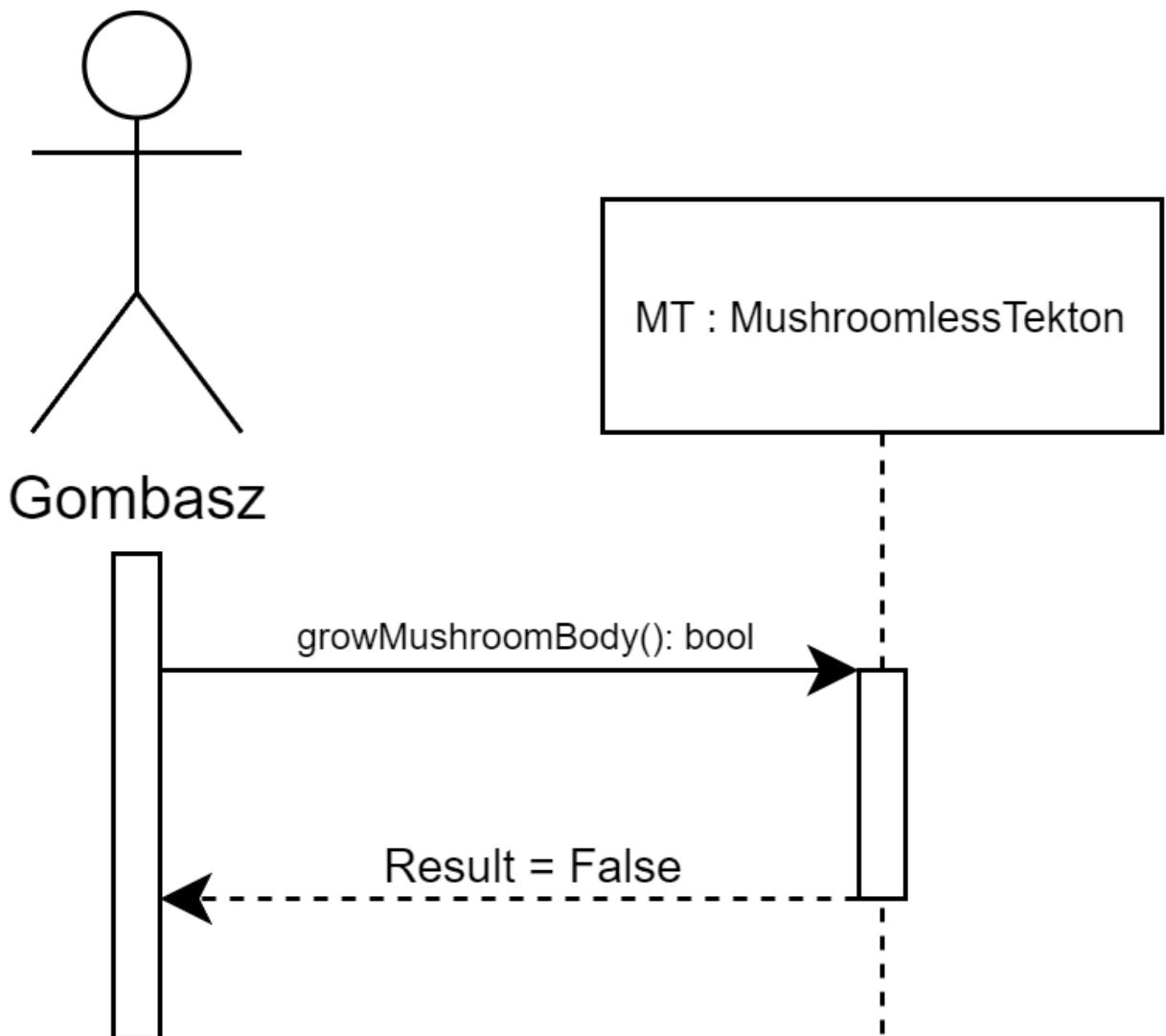
- **List<Tekton> getNeighborsWithThread():** Visszaadja azokat a tektonokat, amelyekkel fonalon keresztül össze van kötve a tekton.
- **bool checkNeighbor(Tekton t):** Ez a metódus visszaadja, hogy egy adott tekton szomszédos-e a tektonnal.
- **Spore popSpore():** Ez a metódus töröl egy rászort spórát a tektonról, és azt a spórát visszatéríti.
- **void updateNeighbor(List<Tekton> newAdd, List<Tekton> delete):** A függvény hozzáadja az új tektonokat tartalmazó listát az aktuális (hívó) tektonhoz, és eltávolítja a törlendő tektonokat tartalmazó listában lévőket.
- **void clearSpore():** A tektonról letörli az összes spórát.
- **void deleteNeighbor(Tekton t):** a parameterként kapott t tekton kitörlése.
- **void addNeighbor(Tekton t):** a parameterként kapott t tekton hozzáadása.
- **Asszociációk**
 - Egyszerű asszociációk:
 - **Tekton:** A tektonok tartalmazzák, hogy mely más tektonokkal szomszédosak.
 - **MushroomThread:** A tektonok tartalmazhatnak gombafonalat (többfonalas esetén többet is).
 - **MushroomBody:** A tekton tartalmazhat egy gombatestet, ha egy játékos növesztett egyet rajta.
 - Kompozíciók:
 - **Spore:** A tektonok tartalmazhatnak spórákat, ezek felgyorsítják a gombafonalak növekedését.

4.3.13. ThreadAbsorbTekton

- **Felelősség**
 - Az osztály a fonalszívó tekton megvalósítása, amely egy specializált tekton. A fonalszívó tekton időközönként felszínya (eltünteti) a rajta lévő gombafonalakat.
- **Ősosztályok**
 - Tekton
- **Metódusok**
 - **void tektonEffect():** Ez a metódus felszínya a tektonon lévő fonalakat

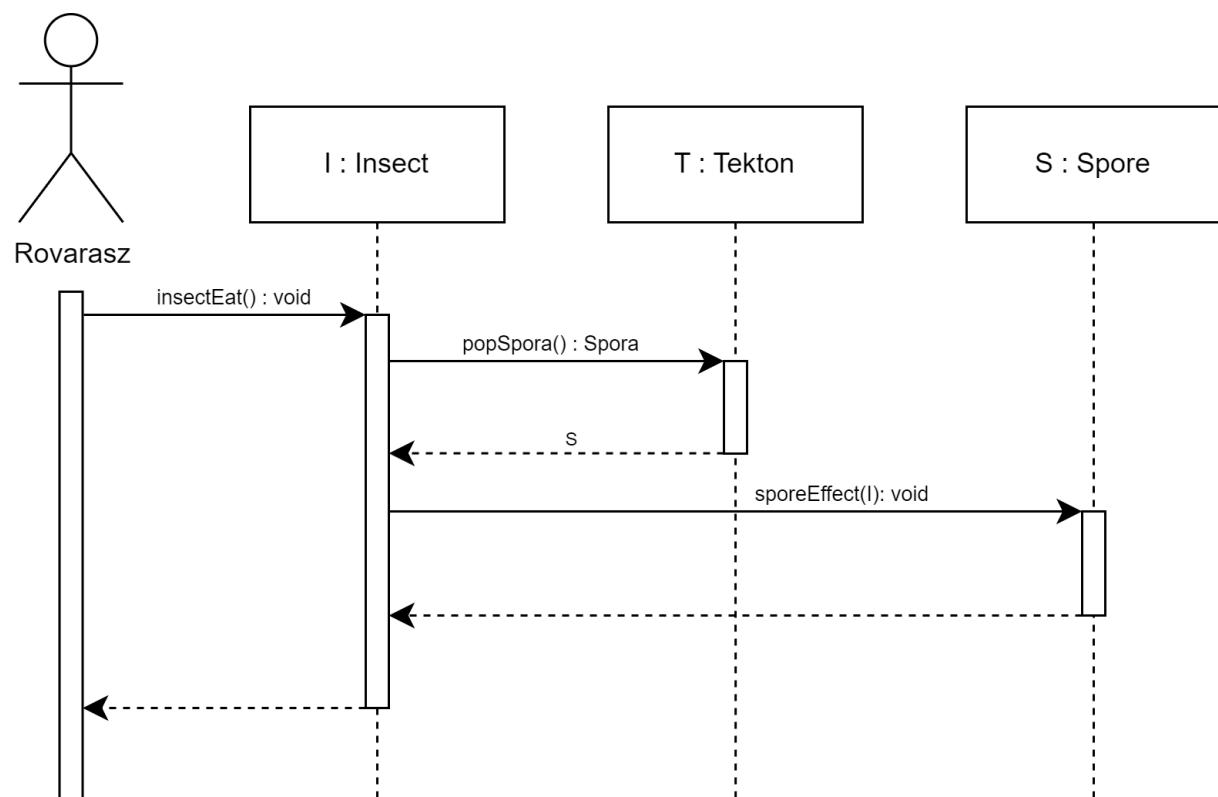
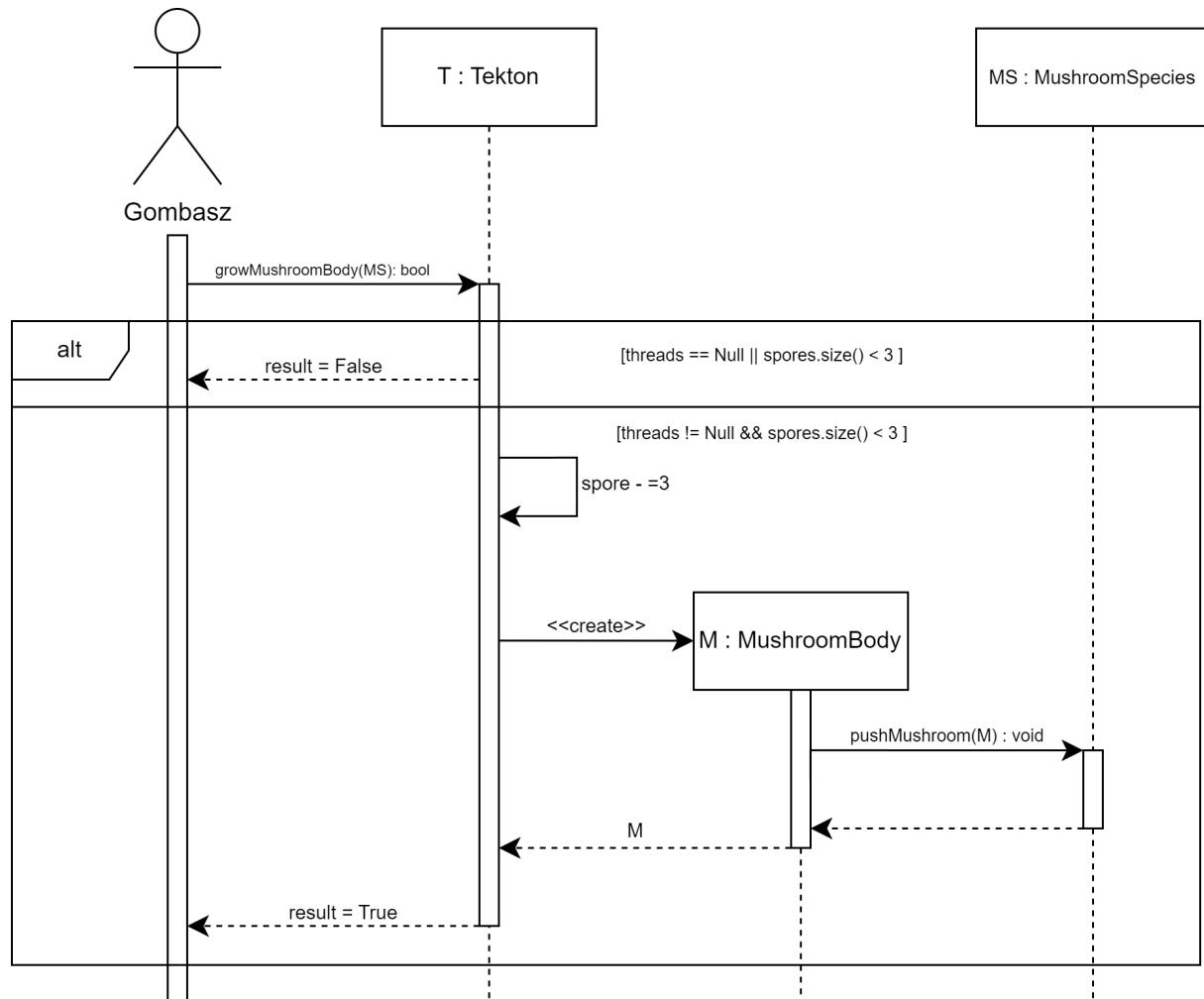
4.4 Szekvencia diagramok

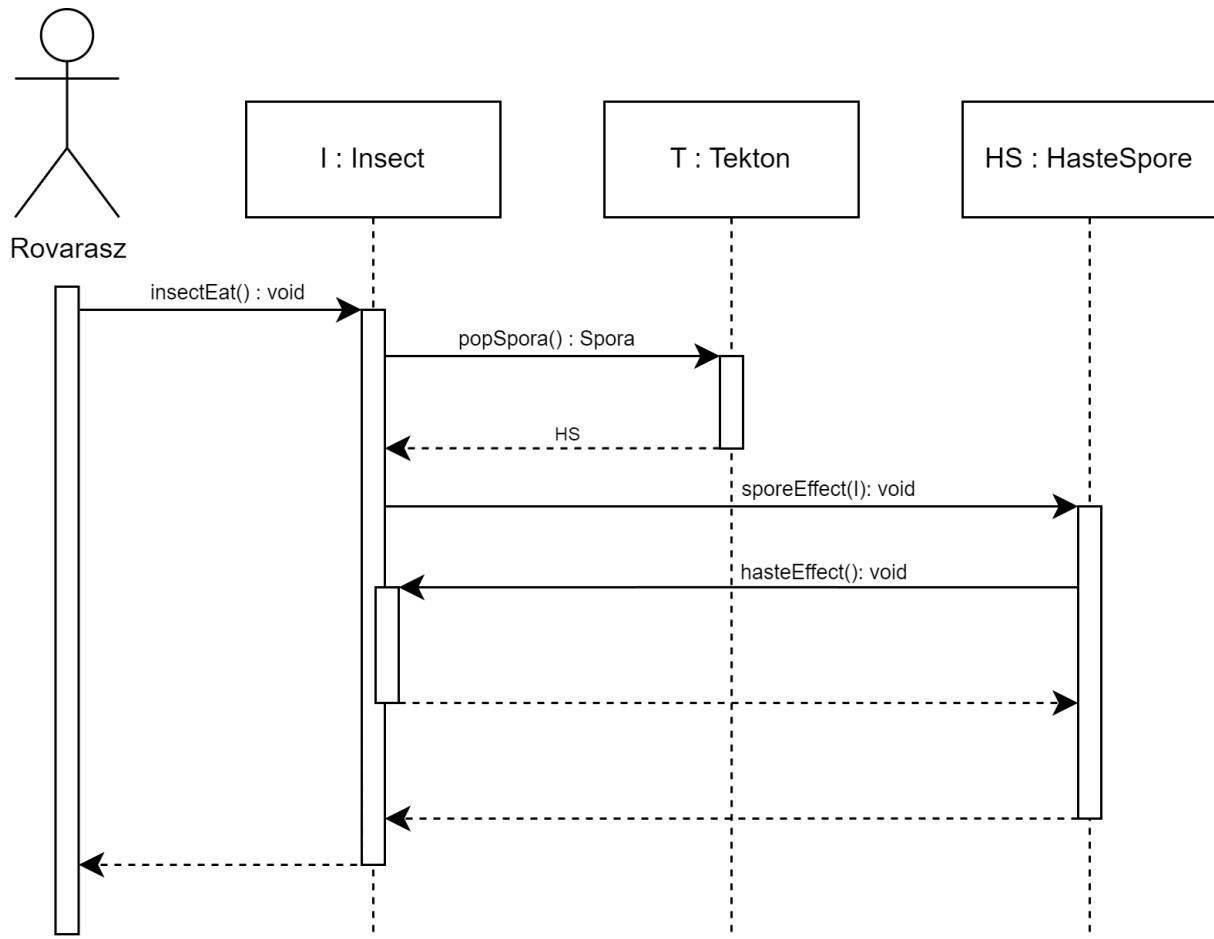


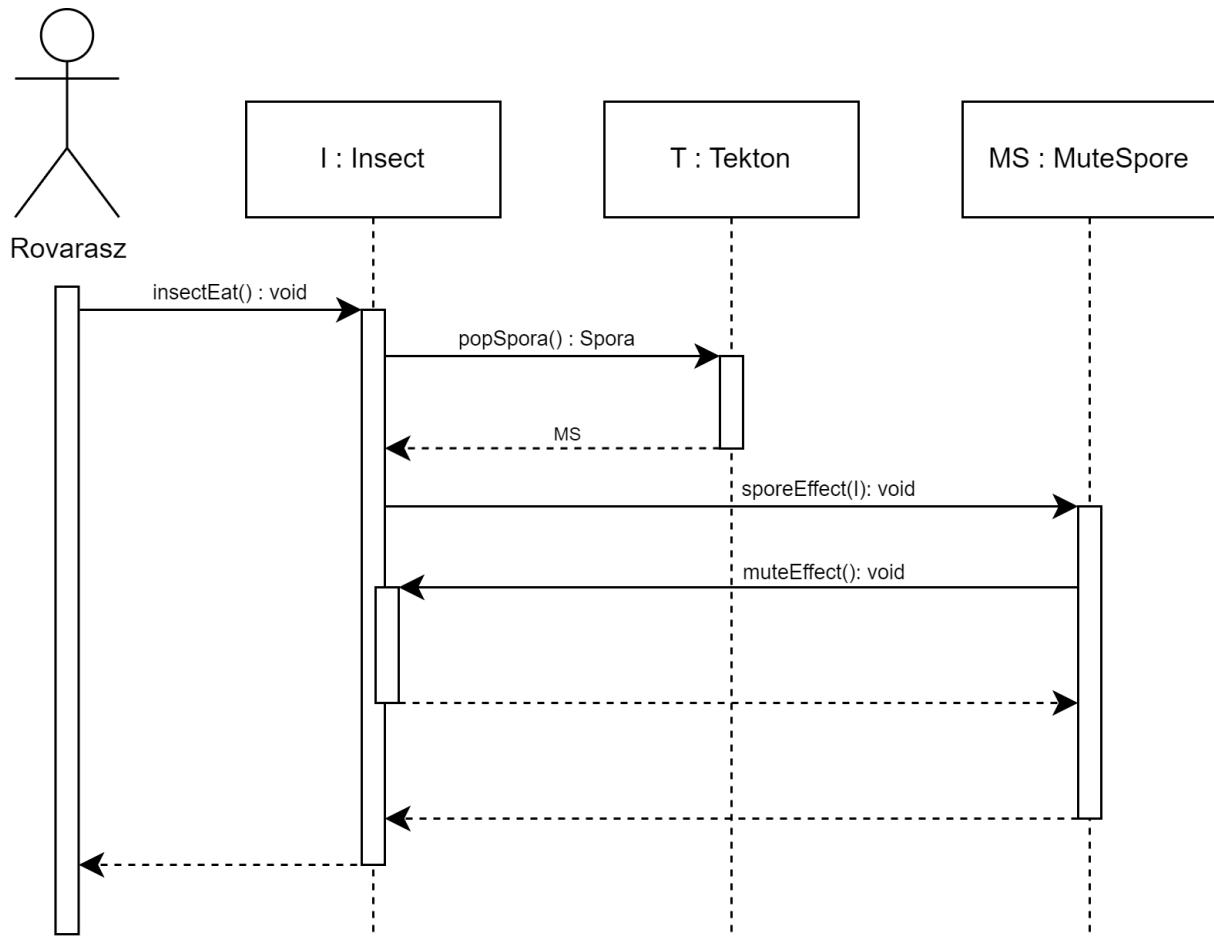


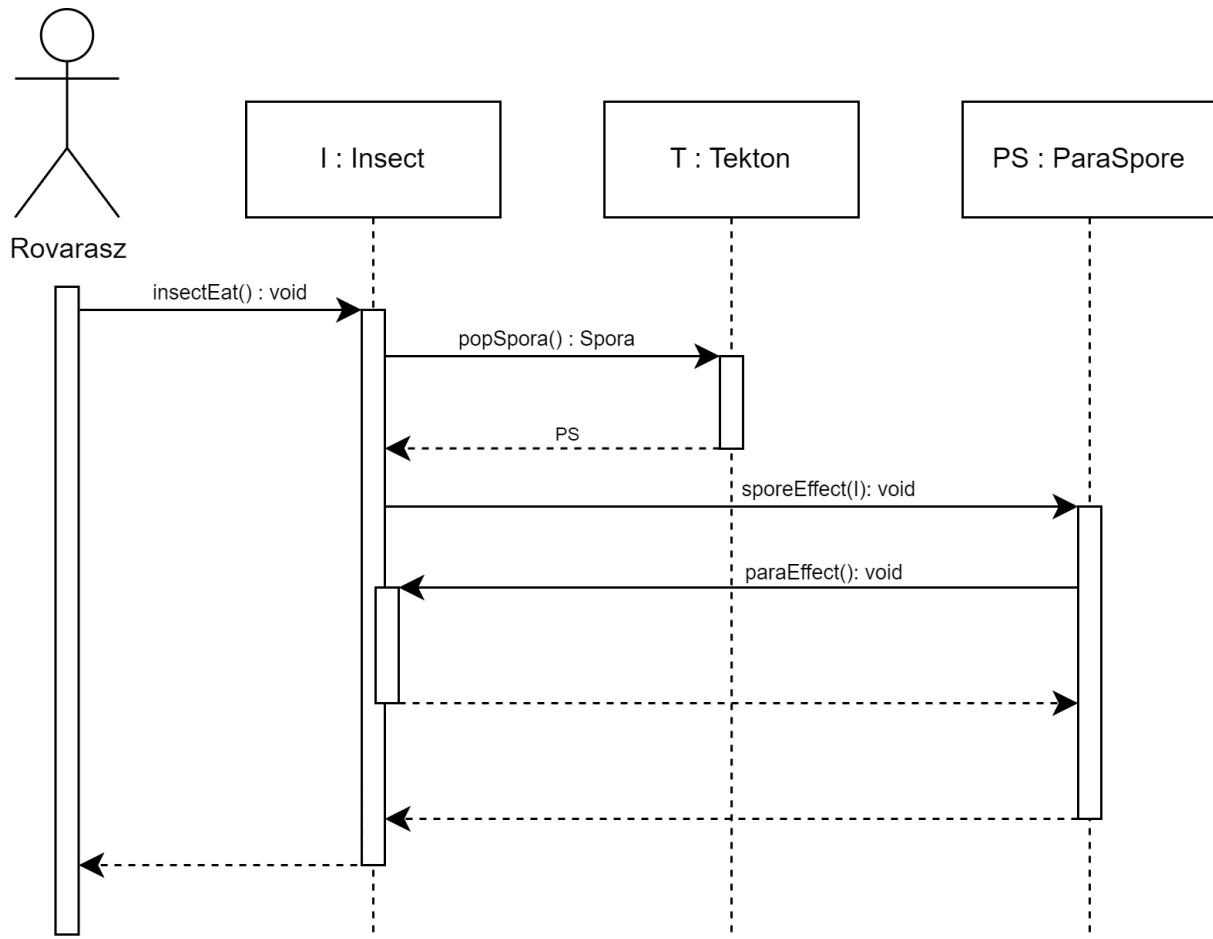
4. Analízis modell kidolgozása

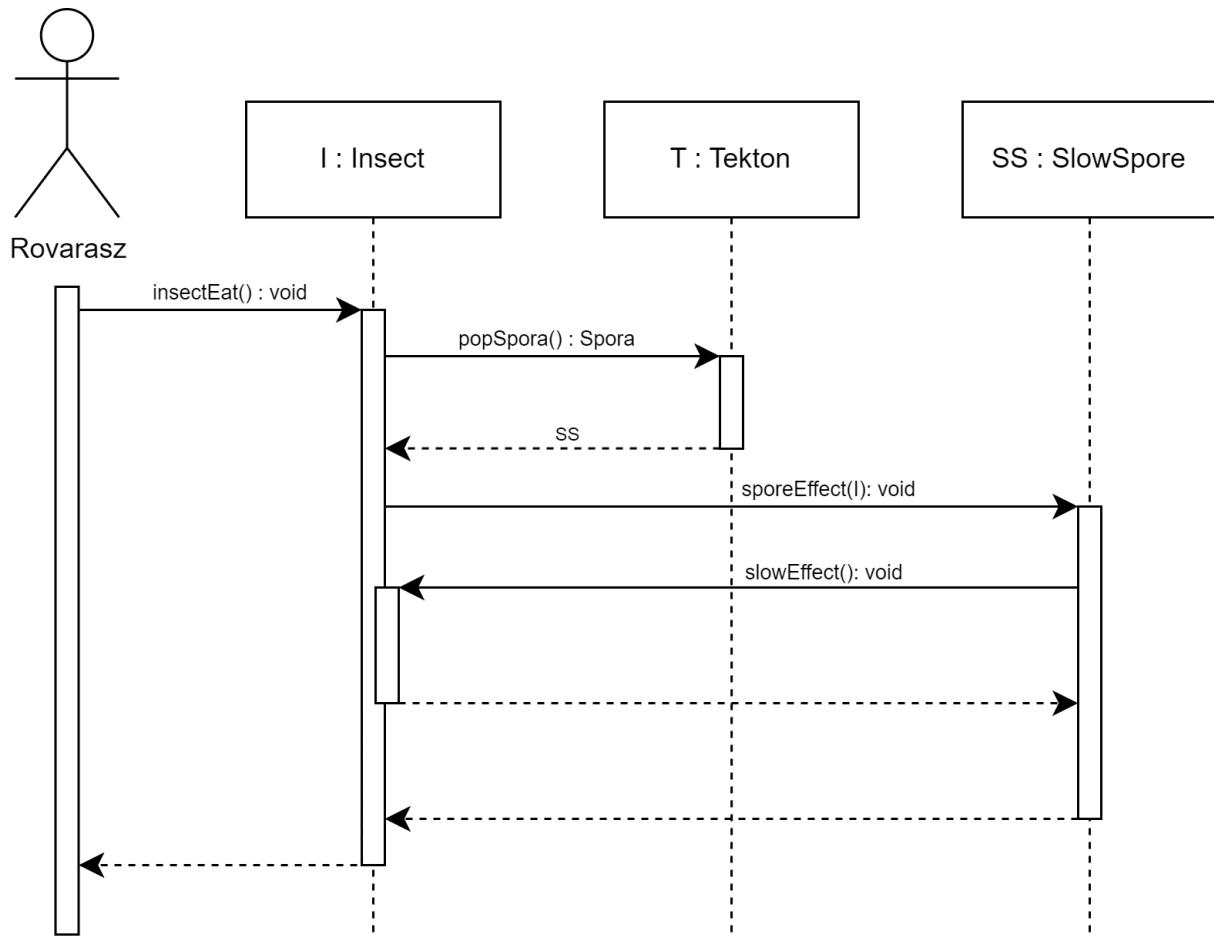
being_chilling

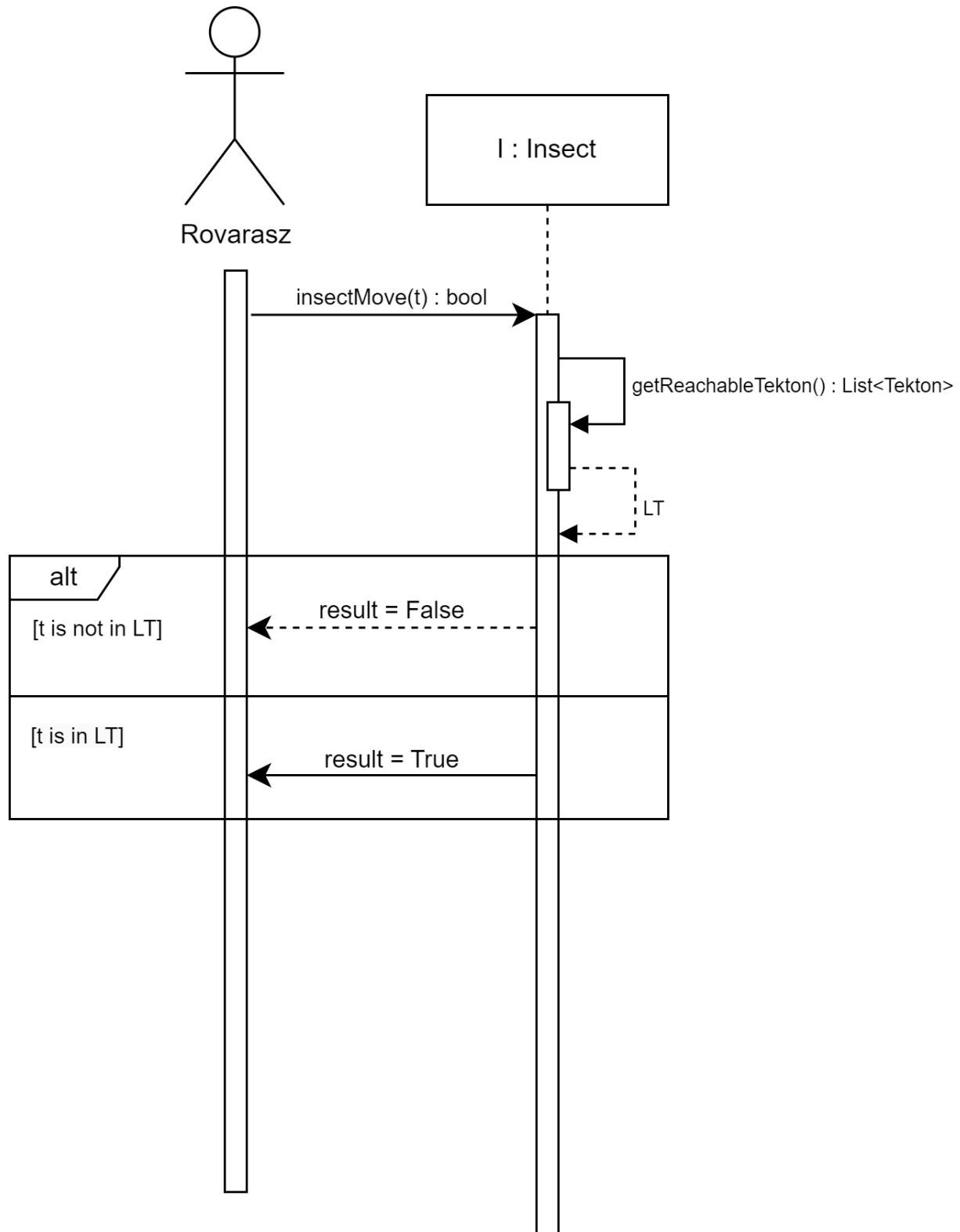


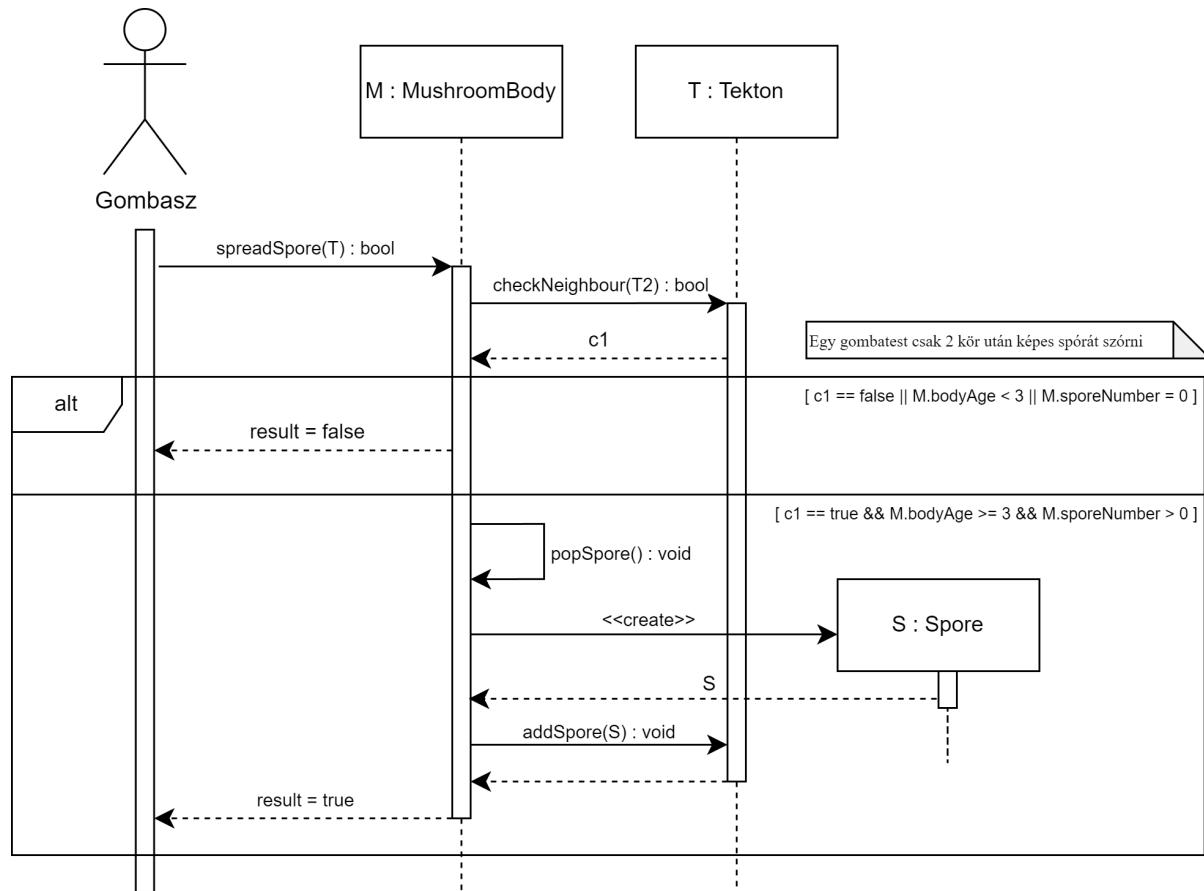


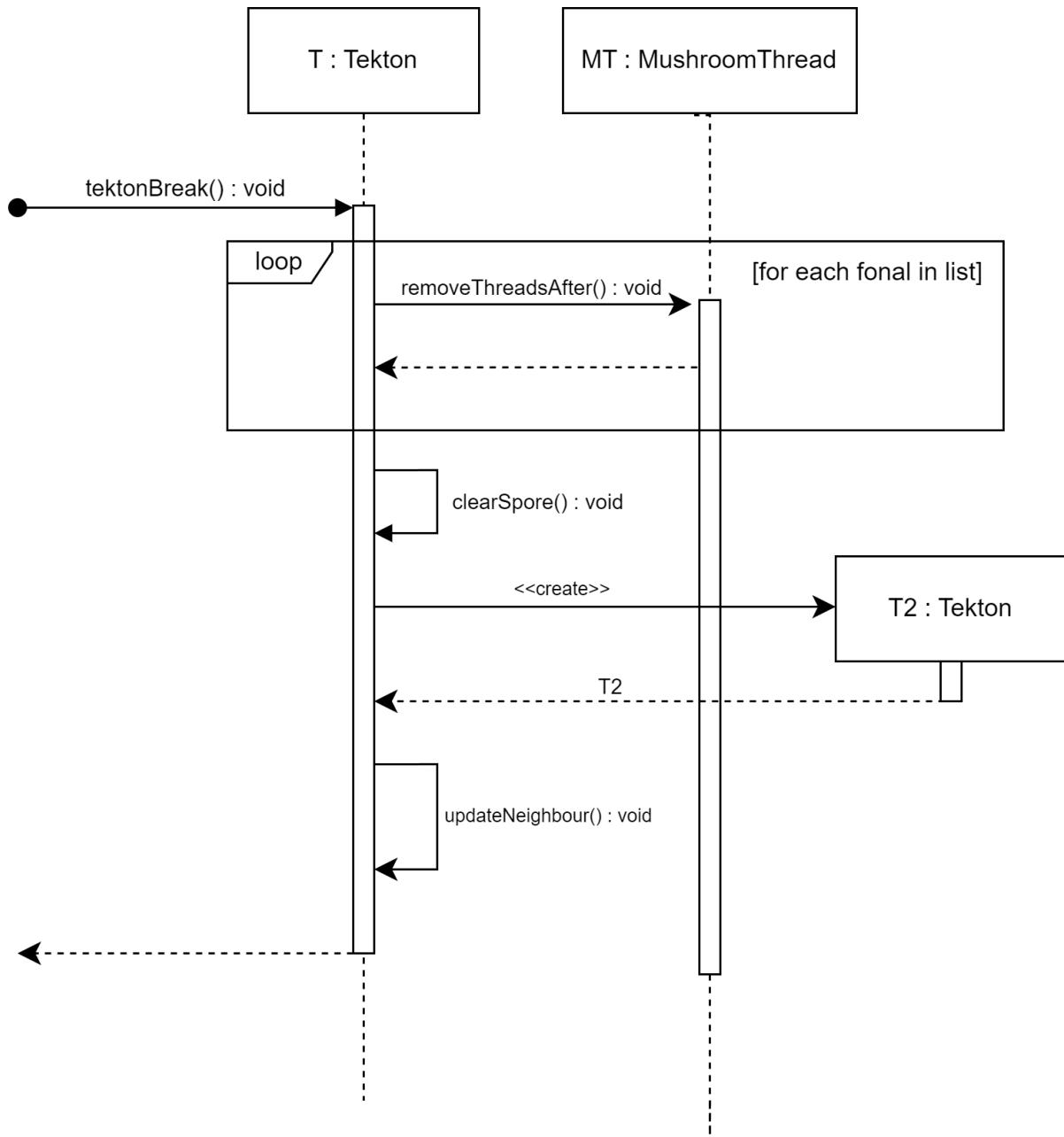


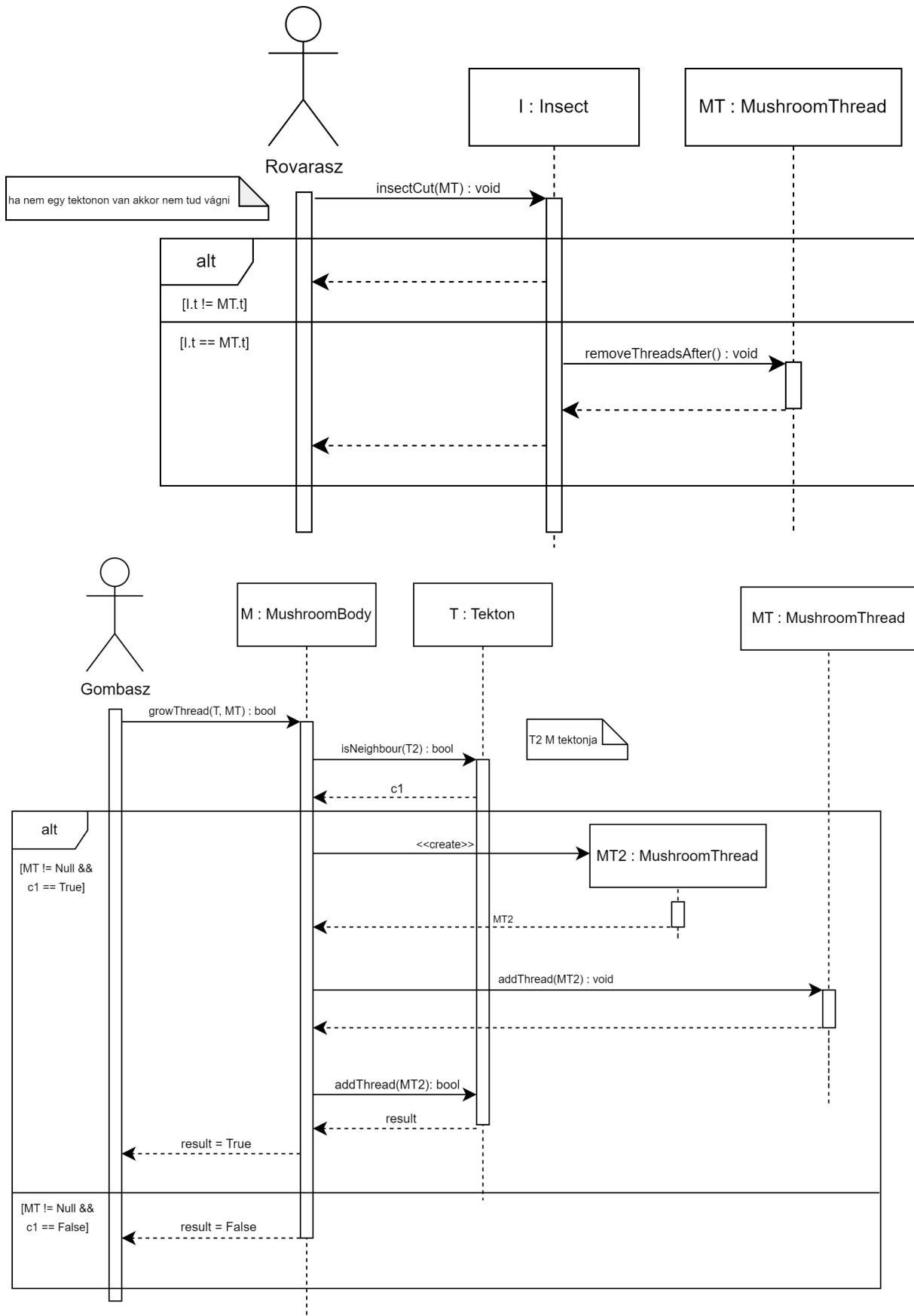




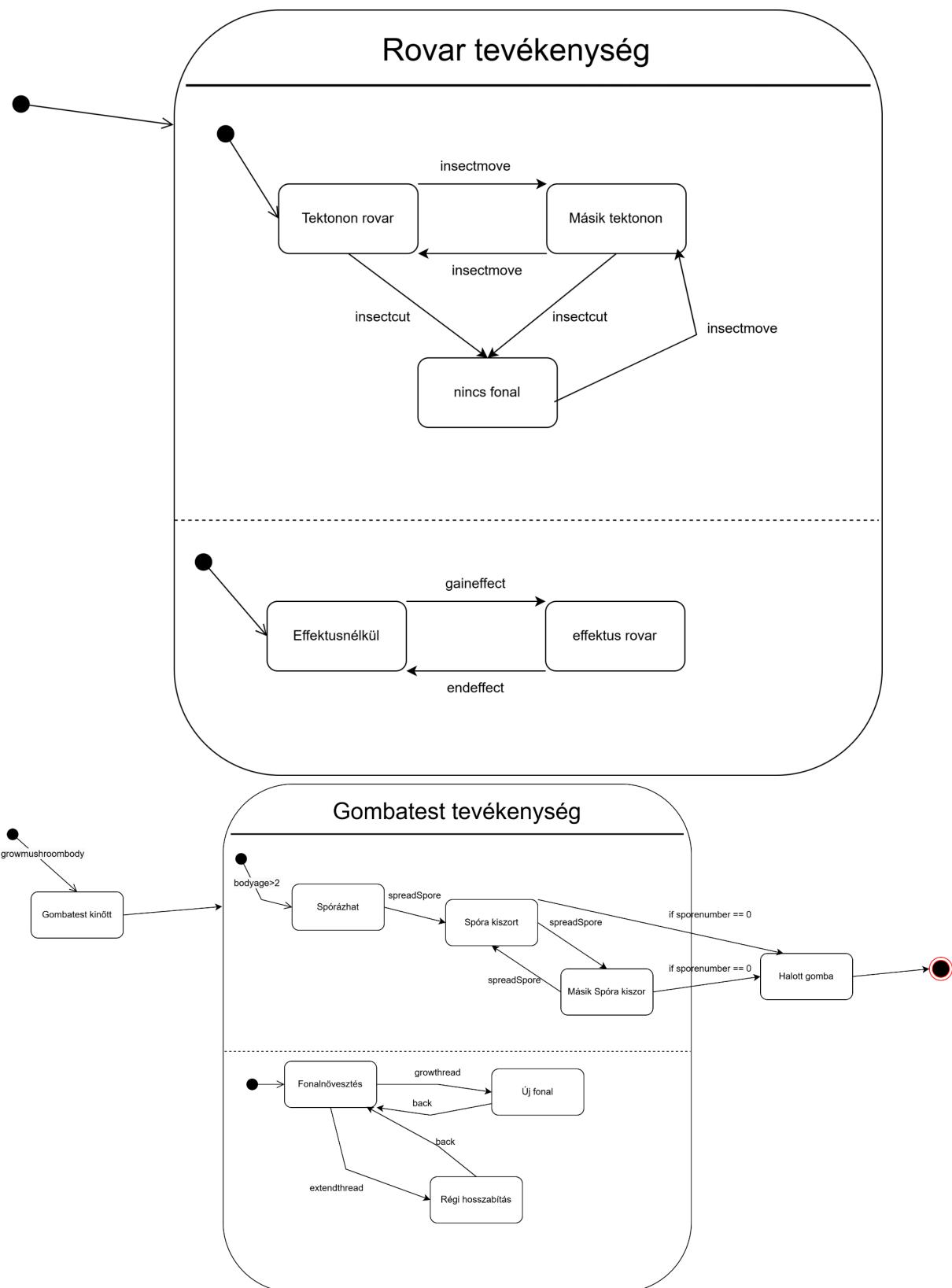








4.5 State-chartok



4.6 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2025.03.06. 15:00	2 óra	Bene Ji Jin Yan Ji Yuang Yin Kattan	Értekezlet. A mindenki feladatának megbeszélése, felosztása.
2025.03.07. 16:00	3 óra	Bene	Tevékenység: UML vázlatos elkészítése
2025.03.07. 17:00	3 óra	Yin	Tevékenység: 2 szekvencia diagram készítése
2025.03.07 12:00	2 óra	Ji Jin Yan	Tevékenység: 4.1 elkészítés
2025.03.07. 18:45	1 óra	Kattan Ji Yuang	Értekezlet: 4.3 elkészítéséről megbeszélés, mit hogyan kell csinálni
2025.03.07. 19:45	3 óra	Kattan	Tevékenység: 4.3 egy részének megírása
2025.03.07. 19:45	3 óra	Ji Yuang	Tevékenység: 4.3-ban saját rész megírása
2025.03.07. 15:00	4 óra	Bene	Tevékenység: 4.2 részletes elkészítése
2025.03.08 13:00	3 óra	Ji Jin Yan	Tevékenység: 4.5 elkészítés
2025.03.08 8:00	4 óra	Yin	Tevékenység: 4.4 elkészítés
2025.03.09. 17:00	2 óra	Bene	Tevékenység: Kis hibák javítása az UML-ben.

5. Szkeleton tervezése

82 – being_chilling

Konzulens:
Tóth Bence

Csapattagok

Yin Jun	TN1KZI	1171276297yin@gmail.com
Bene Máté	K4C4W1	matthew.be04@gmail.com
Kattan Ibrahim Gábor	ZU22JJ	kattanbarhum2@gmail.com
Ji Jin Yan	ZGCKAL	jijinyan0201@gmail.com
Ji Yuang	UI7V2S	yuangji688@gmail.com

2025.03.16.

5. Szkeleton tervezése

5.1 A szkeleton modell valóságos use-case-ai

5.1.1 Use-case diagram

5.1.2 Use-case leírások

Use-case neve	Rovar mozgás effektus nélkül
Rövid leírás	Rovar tektonok között mozog effektus nélkül
Aktorok	Tesztelő
Forgatókönyv	1. Játékos kiválaszt egy tektont, ahová lépni szeretne 2. A rovar a tektonra lép
Előfeltétel	1. Létezik egy rovar, és nincs rajta effektus. 2. A tektonok között van fonal. 3. 2 tektonnyi távolság belül van a kiválasztott tekton
Diagram	sd: insectMove

Use-case neve	Rovar mozgás lassító effektussal
Rövid leírás	Rovar tektonok között mozog lassító effektussal
Aktorok	Tesztelő
Forgatókönyv	1. Játékos kiválaszt egy tektont, ahová lépni szeretne 2. A rovar a tektonra lép
Előfeltétel	1. Létezik egy rovar, és van rajta lassító hatás. 2. Tektonok között van fonal. 3. 1 tektonnyi távolságon belül van a kiválasztott tekton
Diagram	sd: insectMove

Use-case neve	Rovar mozgás gyorsító effektussal
Rövid leírás	Rovar tektonok között mozog gyorsító effektussal
Aktorok	Tesztelő
Forgatókönyv	1. Játékos kiválaszt egy tektont, ahová lépni szeretne 2. Rovar odalép a tektonhoz
Előfeltétel	1. Van egy rovar, és van rajta gyorsító hatás. 2. Tektonok között van fonal. 3. 3 tektonnyi távolságon belül van a kiválasztott tekton
Diagram	sd: insectMove

Use-case neve	Rovar mozgás bénító effektussal
Rövid leírás	Rovar tektonok között nem tud mozogni bénító effektussal
Aktorok	Tesztelő
Forgatókönyv	1. Játékos kiválaszt egy tektont, ahová lépni szeretne 2. Rovar nem lép oda a tektonhoz, és helyben marad.
Előfeltétel	1. A Rovaron van bénító hatás.
Diagram	sd: insectMove

Use-case neve	Rovar mozgás sikertelen
Rövid leírás	Rovar nem tud átmenni a másik tektonra
Aktorok	Tesztelő
Forgatókönyv	1. Játékos kiválaszt egy tektont, ahová lépni szeretne 2. Rovar helyben marad.
Előfeltétel	1a) Tektonok között nem létezik gombafonal. 1b) Nem szomszédos tekton lett kiválasztva.
Diagram	1a) sd: insectMoveUnsec 1b) sd: insectMoveUnsec

Use-case neve	Rovar vágás sikertelen
Rövid leírás	Rovar nem tudja elvágni a kiválasztott fonalat
Aktorok	Tesztelő
Forgatókönyv	1. A játékos kiválaszt egy gombafonalat 2. A rovar nem képes elvágni a gombafonalat, a gombafonal megmarad.
Előfeltétel	1. Egy tektonon van a Rovar és a kiválasztott Fonal 2. Rovar van némitó hatás
Diagram	sd: InsectCutUnsec

Use-case neve	Rovar vágás
Rövid leírás	Rovar elvágja a kiválasztott fonalat
Aktorok	Tesztelő
Forgatókönyv	1. Játékos kiválaszt egy fonalat 2. Rovar elvágja a fonalat 3. A elvágott fonal utáni fonalak felszívódnak
Előfeltétel	1. Egy tektonon van a Rovar és a kiválasztott Fonal 2. Rovar nincs némitó hatás alatt
Diagram	sd: InsectCut

Use-case neve	Gombatest növesztés
Rövid leírás	Egy tektonon egy gombatest növesztése
Aktorok	Tesztelő
Forgatókönyv	1. Kiválaszt egy tektont 2. Kinő a gombatest 3. MushroomSpecies felvészeti a gombatestet
Előfeltétel	1. A tekton nem gombatest nélküli tekton 2. Tektonon van elég spóra 3. Tektonon van gombafonal
Diagram	sd GrowMushroom

Use-case neve	Gombatest növesztése sikertelen
Rövid leírás	Egy tektonon egy gombatest növesztése sikertelen
Aktorok	Tesztelő
Forgatókönyv	1. Kiválaszt egy tektont 2. Nem nő ki gombatest
Előfeltétel	1a) A kiválasztott tektonon nincs elég spóra. 1b) A kiválasztott tektonon nincs fonal. 1c) A kiválasztott tekton gombanélküli.
Diagram	sd: 1a) MushroomGrowUnsec 1b) MushroomGrowUnsec 1c) GrowMushroomOnMushroomless

Use-case neve	Spóraszórás
Rövid leírás	Gombatest kiszórja a spóráját
Aktorok	Tesztelő
Forgatókönyv	1. Játékos kiválaszt egy tekton 2. Gombatest kiszórja a spóráját a tektonra
Előfeltétel	1. Van elég spóra a gombatestben 2. Kiválasztott tekton hatótávolságban belül van 3. Tud spórát szórni (életkora ≥ 2)
Diagram	sd: Spóraszórás

Use-case neve	Spóraszórás sikertelen
Rövid leírás	Gombatestből spórát szórni egy tektonra nem sikerült
Aktorok	Tesztelő
Forgatókönyv	1. Játékos kiválaszt egy tekton 2. Gombatest nem tud spórát szórni.
Előfeltétel	1a). A gomba testben lévő spóra nem elegendő 1b). Kiválasztott tekton nincs hatótávolságban belül
Diagram	sd: Spóraszórás Sikertelen

Use-case neve	Gombafonal növesztése
Rövid leírás	Gombatest kinöveszti a fonalat egy tektonra
Aktorok	Tesztelő
Forgatókönyv	1. Játékos kiválaszt egy tekton 2. Gombatest kinöveszti a fonalat arra a tektonra
Előfeltétel	1. A kiválasztott tekton több fonalas, vagy más fajta tekton, de nincs rajta fonal
Diagram	

Use-case neve	Gombafonal növesztés sikertelen
Rövid leírás	Gombatestből egy gombafonal kinövesztése egy tektonra sikertelen.
Aktorok	Tesztelő
Forgatókönyv	1. Játékos kiválaszt egy tekton. 2. Gombatest nem növeszti ki a gombafonalat.
Előfeltétel	1) Egy nem többfonalas tektonon már létezik egy darab gombafonal
Diagram	

Use-case neve	Gombafonal gyorsítás
Rövid leírás	Tektonon van spóra, és a fonál a spórázott tektonra nő
Aktorok	Tesztelő
Forgatókönyv	1. Játékos növeszt egy fonalat a kiválasztott tektonra 2. Tektonon nő fonal

	3. Random irányban növeszt még egy fonalat
Előfeltétel	1. Tektonon (amelyikre növesztendő a fonal) van spóra 2. Gombatest arra a tektonra növeszt fonalat 3. A random növesztett fonal egy szomszédos tektonra (ami a tesztelő által kiválasztott tekton szomszéda) nő
Diagram	sd: SpeedUpThreadGrowing

Use-case neve	Tekton kettétörése
Rövid leírás	A tekton kettétörök
Aktorok	Tesztelő
Forgatókönyv	1. Tektonon lévő fonal törlődik 2. Az új tekton a régi tekton szomszéd listából egy részhalmazát kap 3. Töröljük a már nem szomszédos tekton a régiből
Előfeltétel	1a). Tektonon nincsen rovar. 1b). Tektonon nincsen gombatest.
Diagram	sd: tektonBreak cd: tektonBreak

Use-case neve	Tekton kettétörése sikertelen
Rövid leírás	A tekton egy idő utáni kettétörése sikertelen
Aktorok	Tesztelő
Forgatókönyv	1. Egy bizonyos idő elteltével, a tekton ketté törne, csak nem lehetséges
Előfeltétel	1a). Tektonon van rovar. 1b). Tektonon van gombatest.
Diagram	sd: breakUnsec

Use-case neve	Rovar Eszik (gyorsító Spóra)
Rövid leírás	Gyorsító spóra hatásai
Aktorok	Tesztelő
Forgatókönyv	1. Rovar megeszik egy spórát a tektonon. 2. Spóra megadja a rovarnak a hatást 3. Spóra törlődik a tektonról
Előfeltétel	1. Rovar megegett egy gyorsító spórát
Diagram	sd: HasteSpore

Use-case neve	Rovar Eszik(lassító Spóra)
Rövid leírás	Lassító spóra hatásai
Aktorok	Tesztelő
Forgatókönyv	1. Rovar megeszik egy spórát a tektonon. 2. Spóra megadja a rovarnak a hatást 3. Spóra törlődik a tektonról
Előfeltétel	1. Rovar megegett egy lassító spórát

Diagram	sd: SlowSpore
----------------	---------------

Use-case neve	Rovar Eszik(bénító Spóra)
Rövid leírás	Bénító spóra hatásai
Aktorok	Tesztelő
Forgatókönyv	<ol style="list-style-type: none"> 1. Rovar megeszik egy spórát a tektonon. 2. Spóra megadja a rovarnak a hatást 3. Spóra törlődik a tektonról
Előfeltétel	1. Rovar megevett egy bénító spórát
Diagram	sd: ParaSpore

Use-case neve	Rovar Eszik(némító Spóra)
Rövid leírás	Némítő spóra hatásai
Aktorok	Tesztelő
Forgatókönyv	<ol style="list-style-type: none"> 1. Rovar megeszik egy spórát a tektonon. 2. Spóra megadja a rovarnak a hatást 3. Spóra törlődik a tektonról
Előfeltétel	1. Rovar megevett egy némítő spórát
Diagram	sd: MuneSpore

Use-case neve	Rovar Eszik
Rövid leírás	Rovar megeszi a tektonon lévő spórát
Aktorok	Tesztelő
Forgatókönyv	<ol style="list-style-type: none"> 1. Rovar megeszik egy spórát a tektonon. 2. Spóra törlődik a tektonról
Előfeltétel	1. Van tektonon spóra.
Diagram	sd: NormalSpore

5.2 A szkeleton kezelői felületének terve, dialógusok

A szkeleton felhasználói felülete menüvezérelt módon fog történni. A tesztelő számokkal fog tudni navigálni a menüpontok között. minden menüpont egy tesztesetet, vagy egy inicializálást fog elindítani.

Elvárt bemenetek a tesztelőről:

A tesztelő: számok, betűk beírásával tud lépegetni a menüpontok között. Ha esetleg más input lenne, mint ami lehetséges, nem fog történni semmi.

A megjelenő kimenetek:

Miután a tesztelő elindított egy tesztesetet, a konzolon a teszteset lefutása fog lejátszódni. Ezt a konzolra történő kiírások fogják mutatni.

A konzolra a teszteset neve fog először megjelenni, majd új sorba a függvényhívások. A függvényhívások egymás után 1-el következő tabulátorral jelennek meg. A metódusokat ha meghívíták őket, akkor > szimbólum, ha véget ért < szimbólum jelzi. A sorban először a Class neve szerepel, majd a Class függvénye paraméterekkel. A visszatérési érték, ha van a

visszafele szimbólum (<) után fog megjelenni.

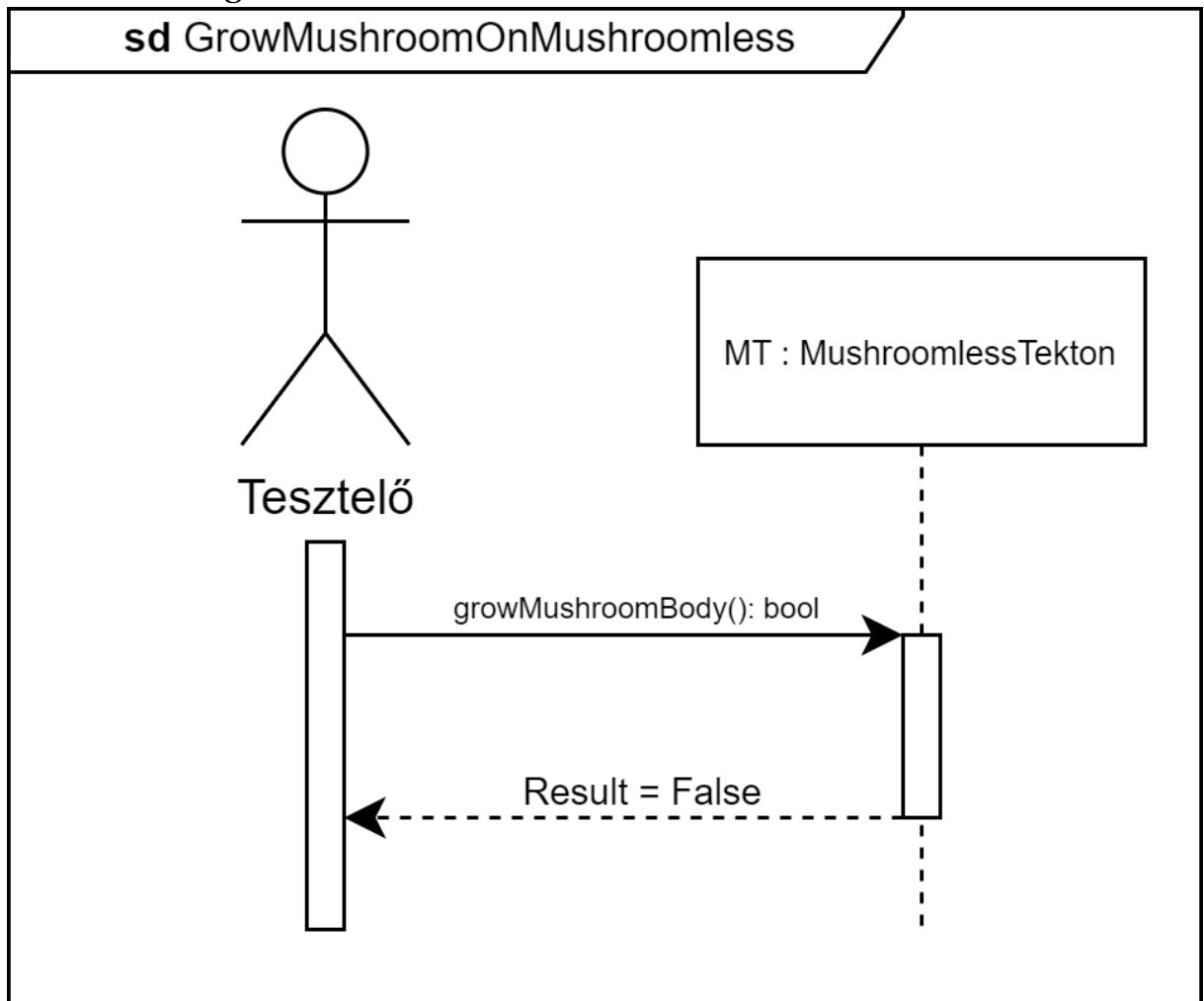
Például:

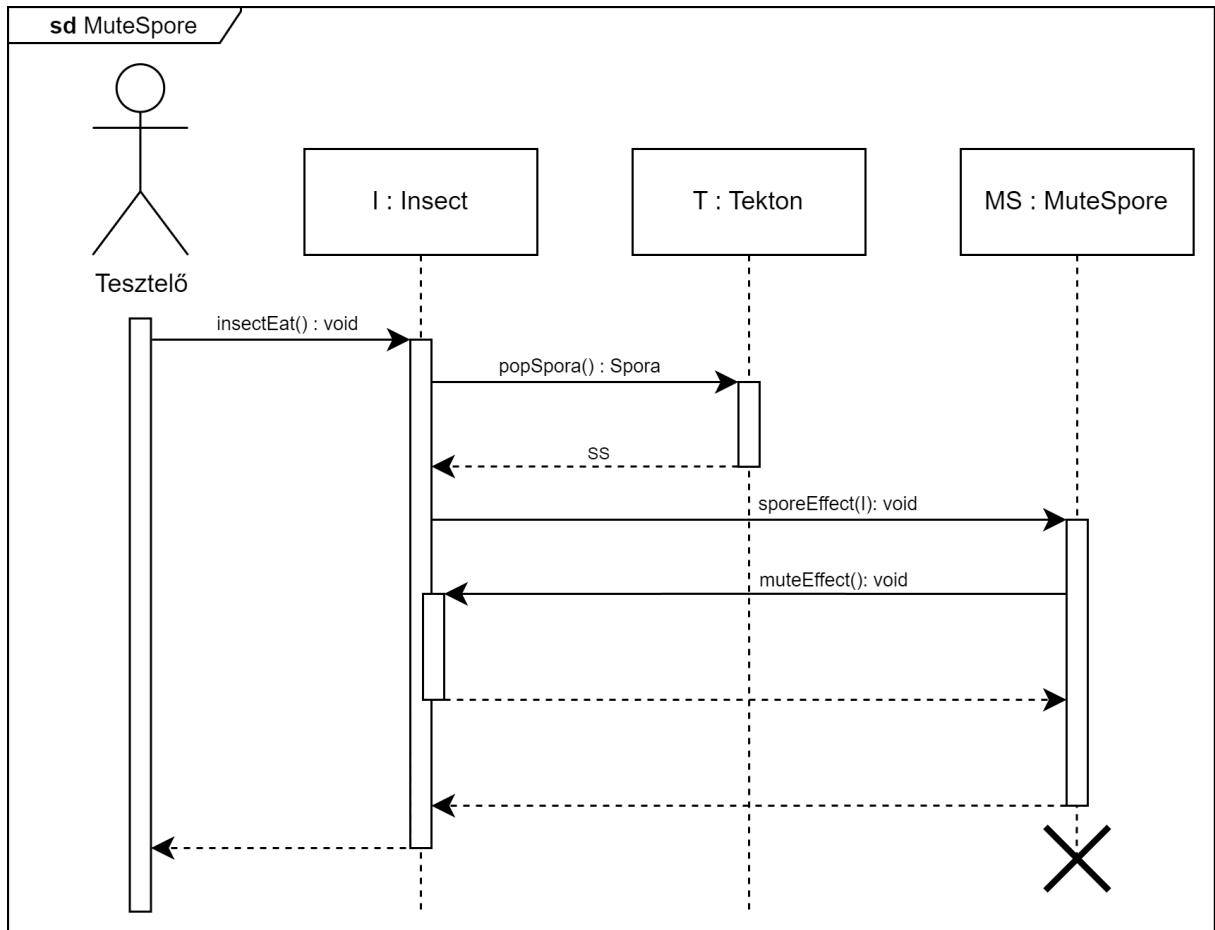
Rovar Eszik:

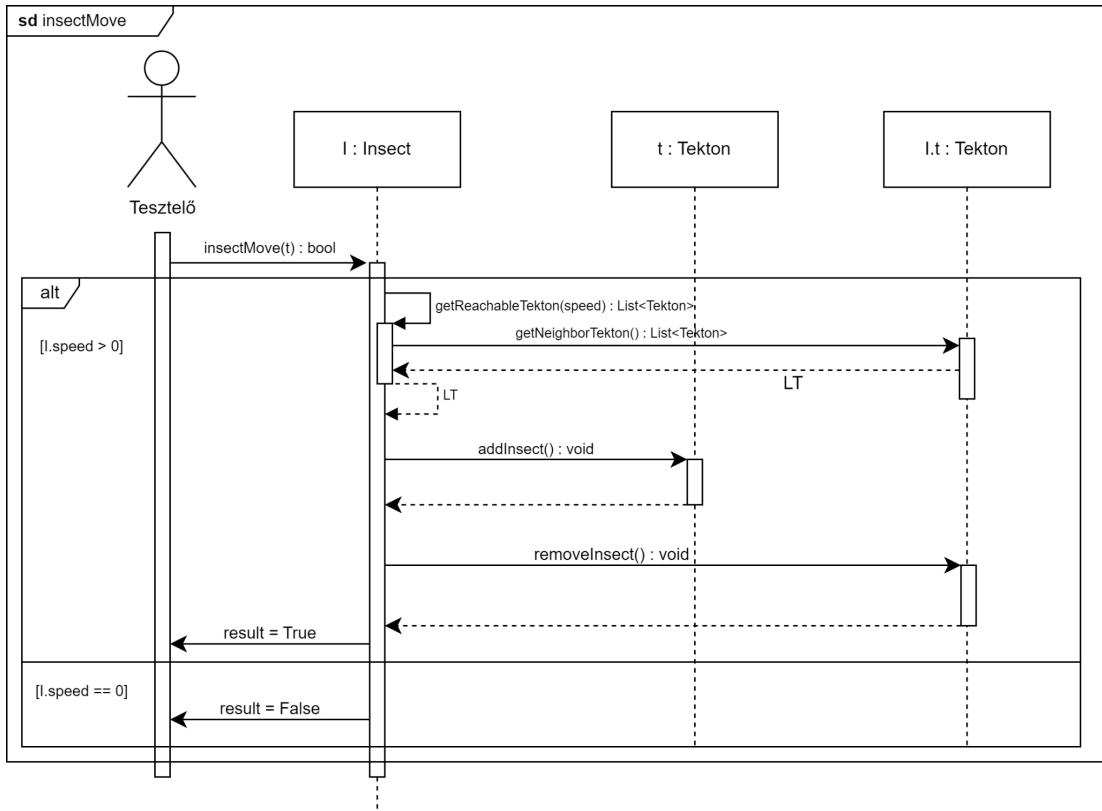
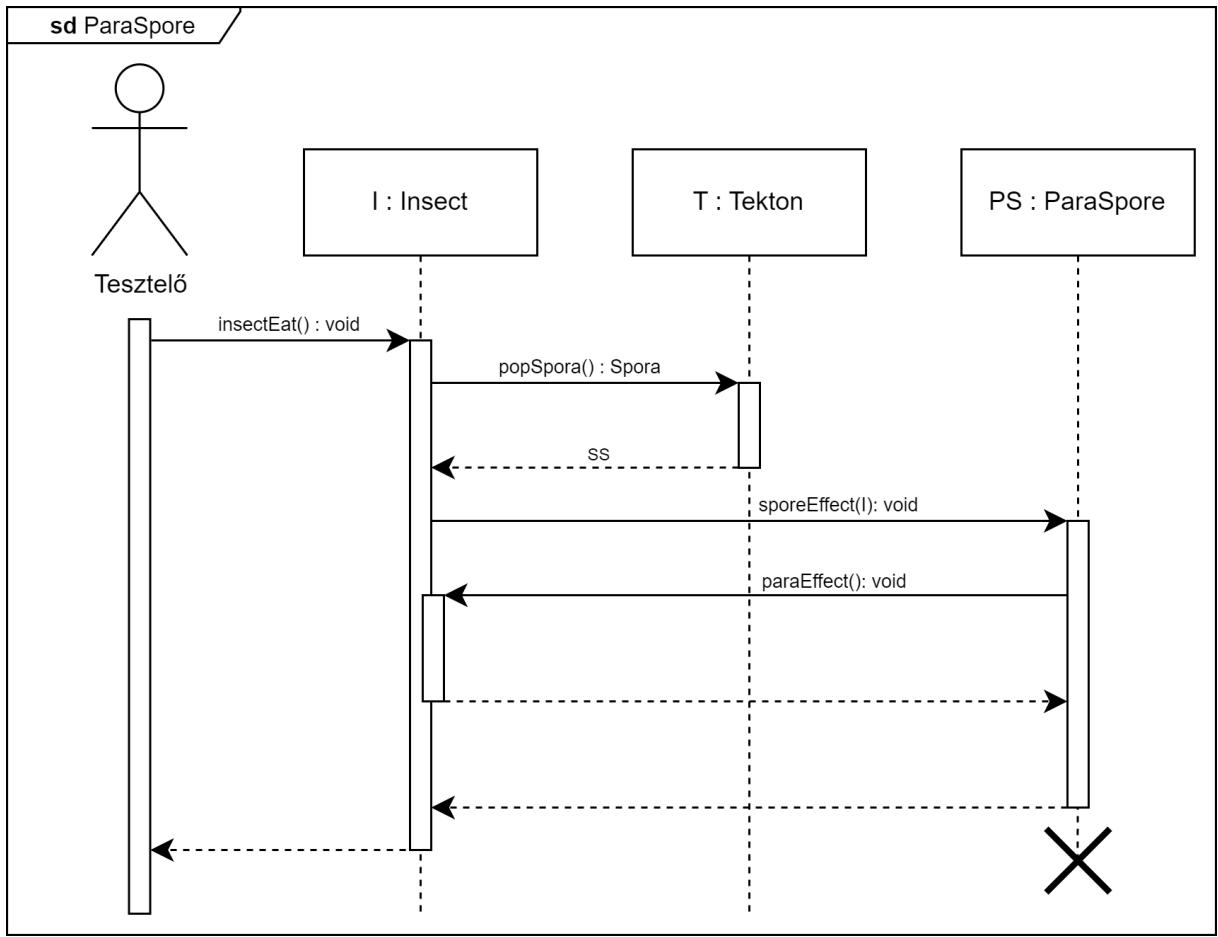
```
> Insect.insectEat() : void
    > Tekton.popSpore(): Spora
    < s : Spore
    > sporeEffect(i: Insect): void
    <
<
```

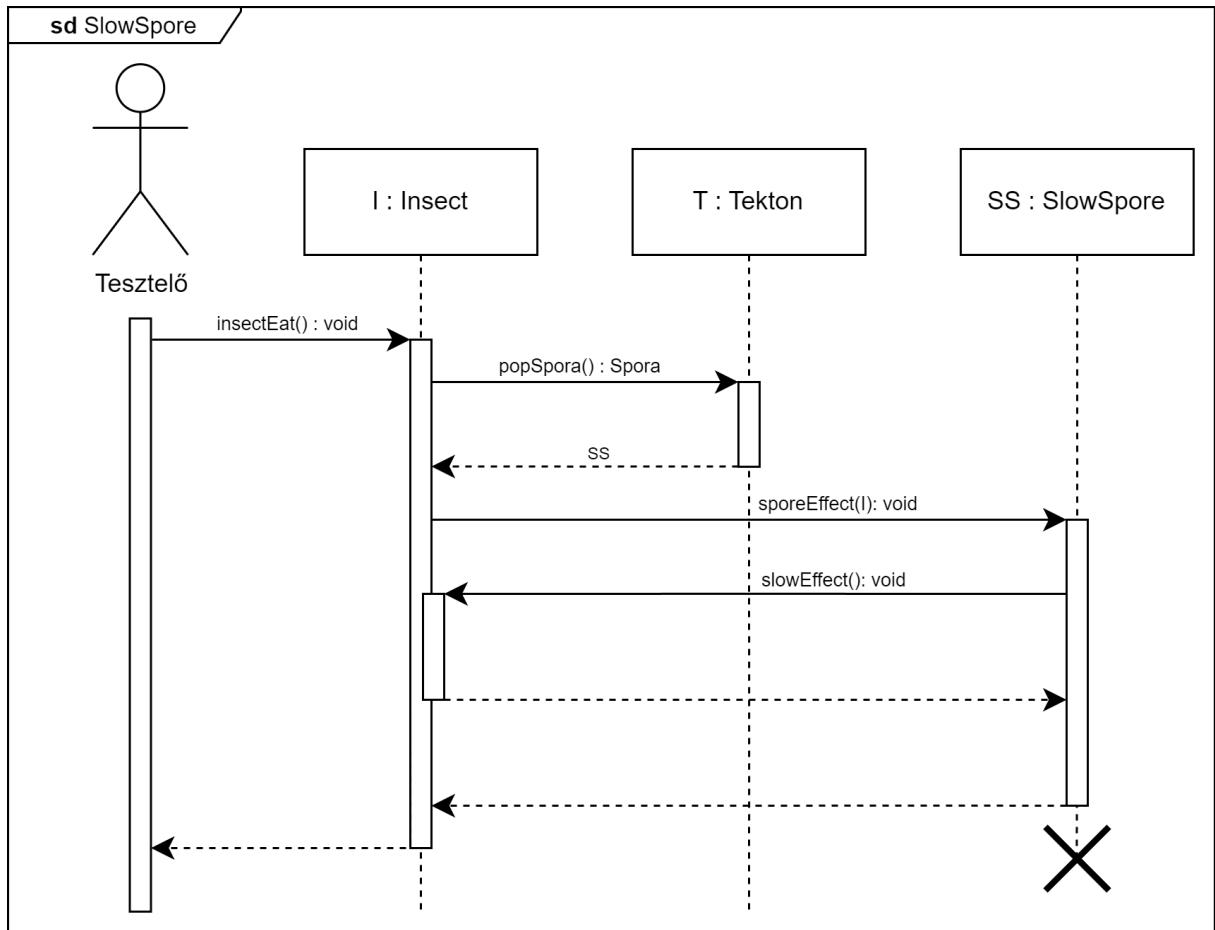
A menüelemek:

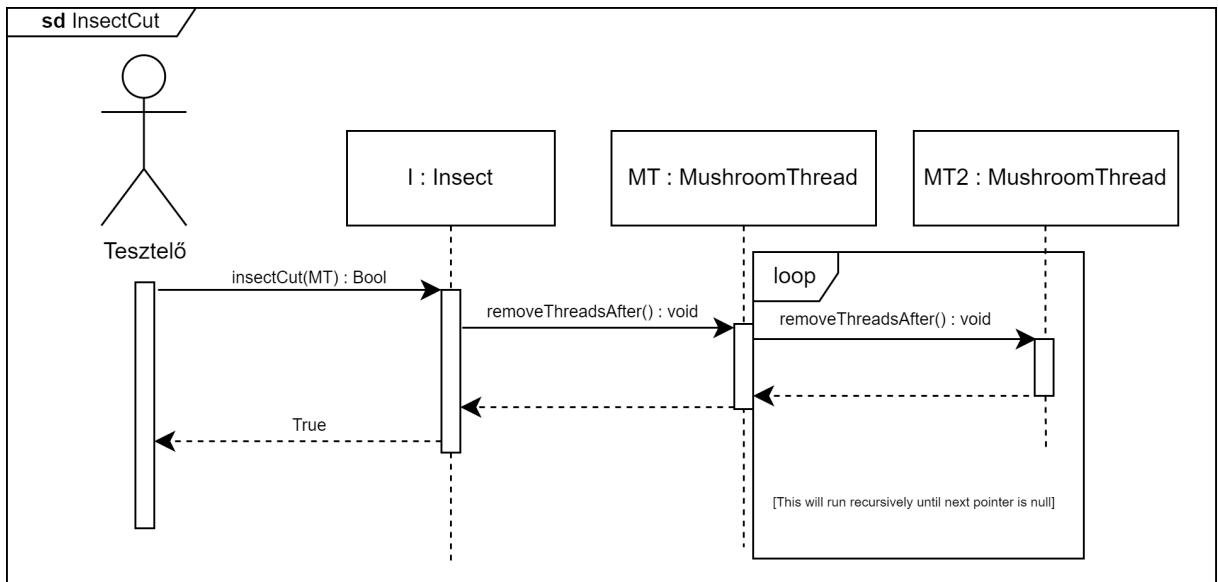
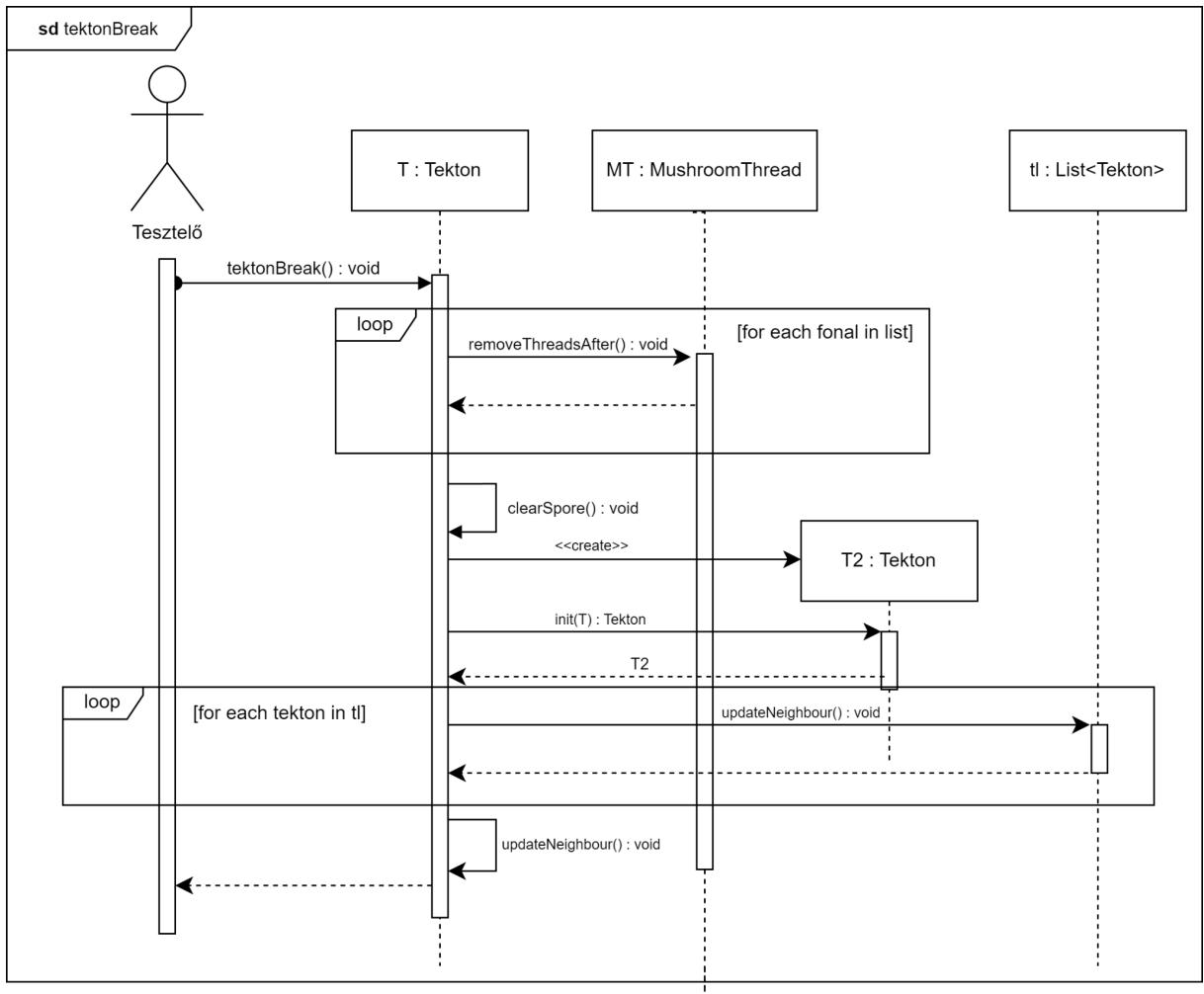
1. Rovar mozgás effektus nélkül
2. Rovar mozgás lassító effektussal
3. Rovar mozgás gyorsító effektussal
4. Rovar mozgás bénító effektussal
5. Rovar mozgás sikertelen
6. Rovar vágás sikertelen
7. Rovar vágás
8. Gombatest növesztés
9. Gombatest növesztése sikertelen
10. Spóraszórás
11. Spóraszórás sikertelen
12. Gombafonal növesztése
13. Gombafonal növesztés sikertelen
14. Gombafonal gyorsítás
15. Tekton kettétörése
16. Tekton kettétörése sikertelen
17. Rovar Eszik (gyorsító Spóra)
18. Rovar Eszik (lassító Spóra)
19. Rovar Eszik (bénító Spóra)
20. Rovar Eszik (némító Spóra)
21. Rovar Eszik
0. Exit

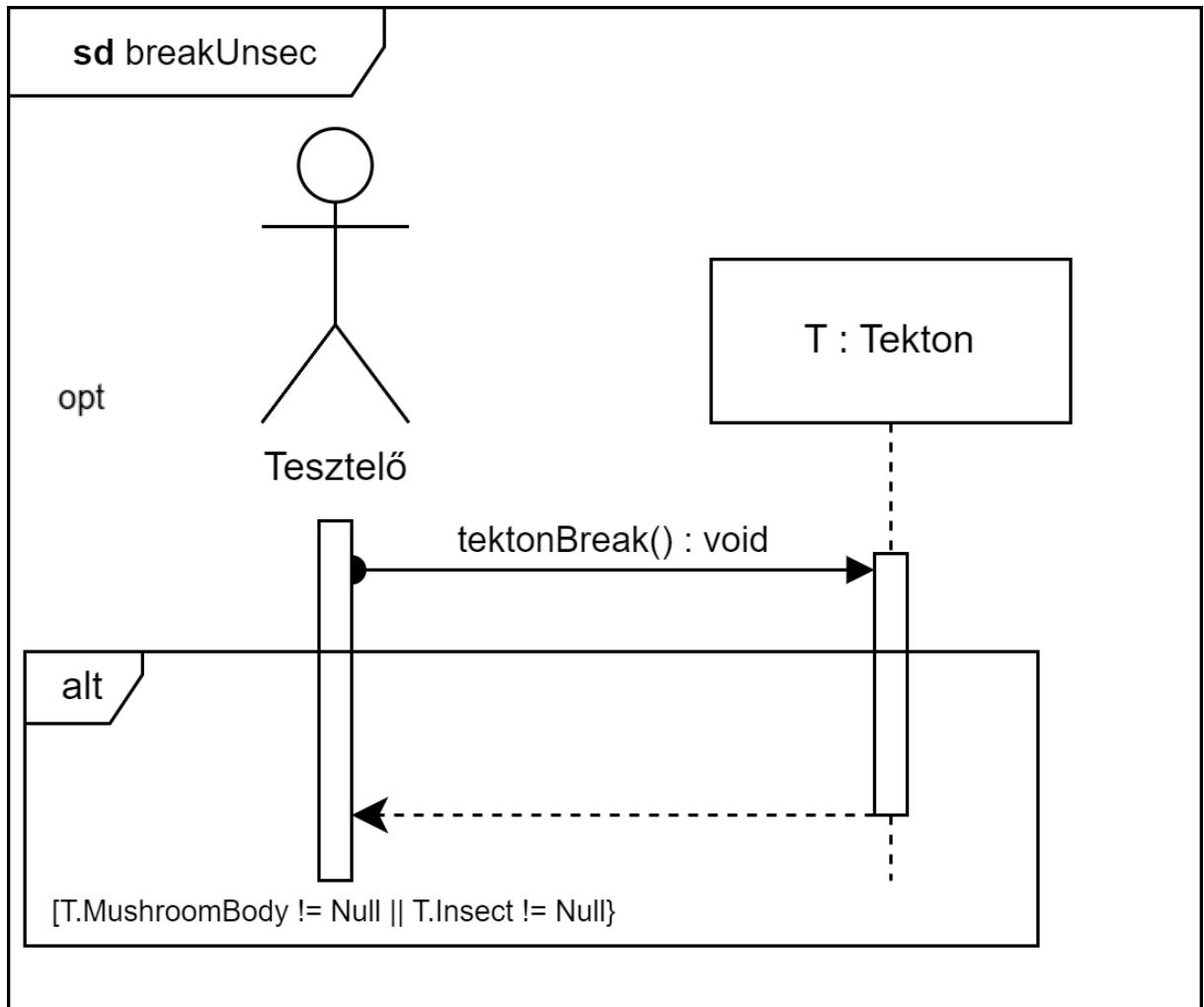
5.3 Szekvencia diagramok a belső működésre

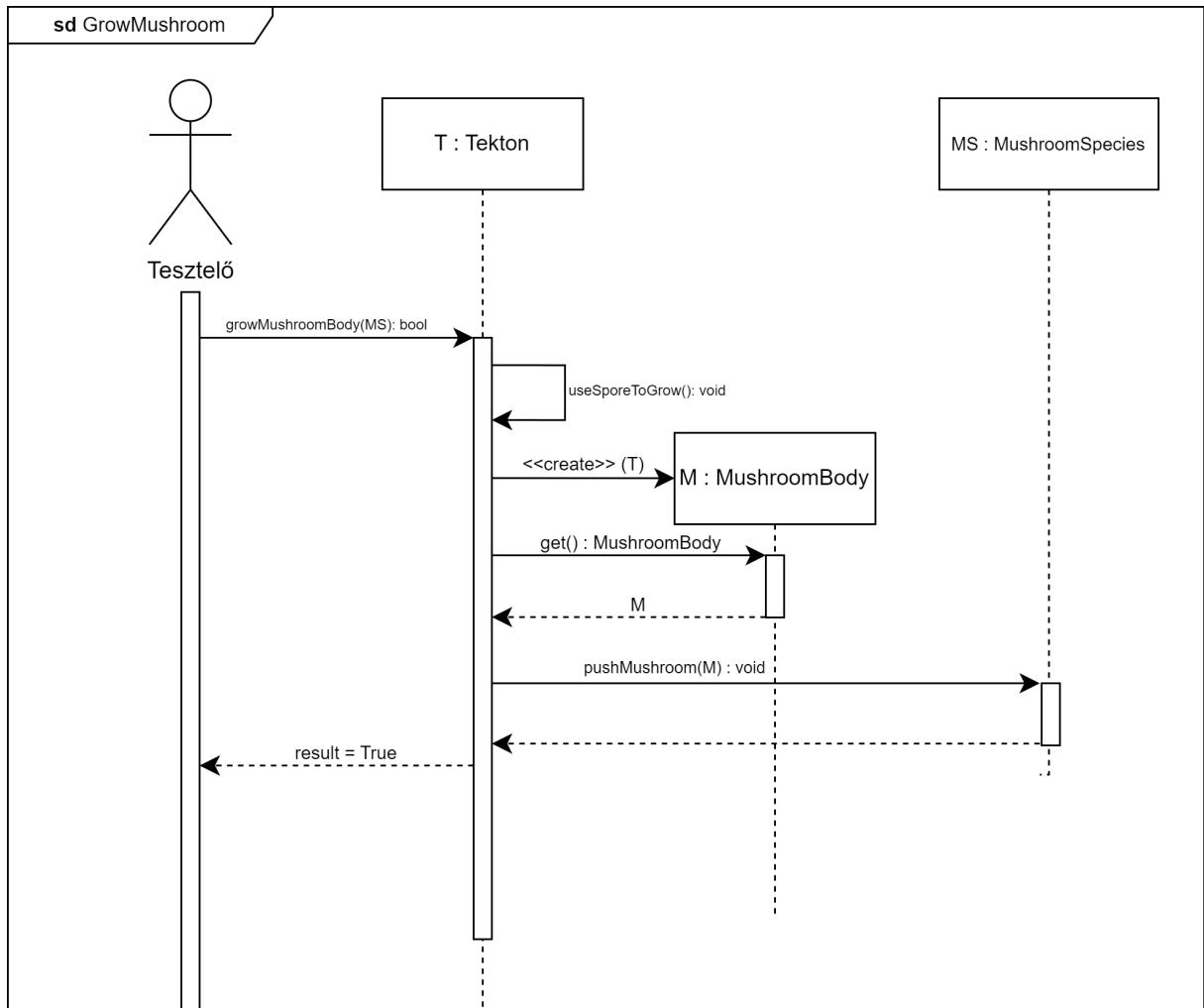


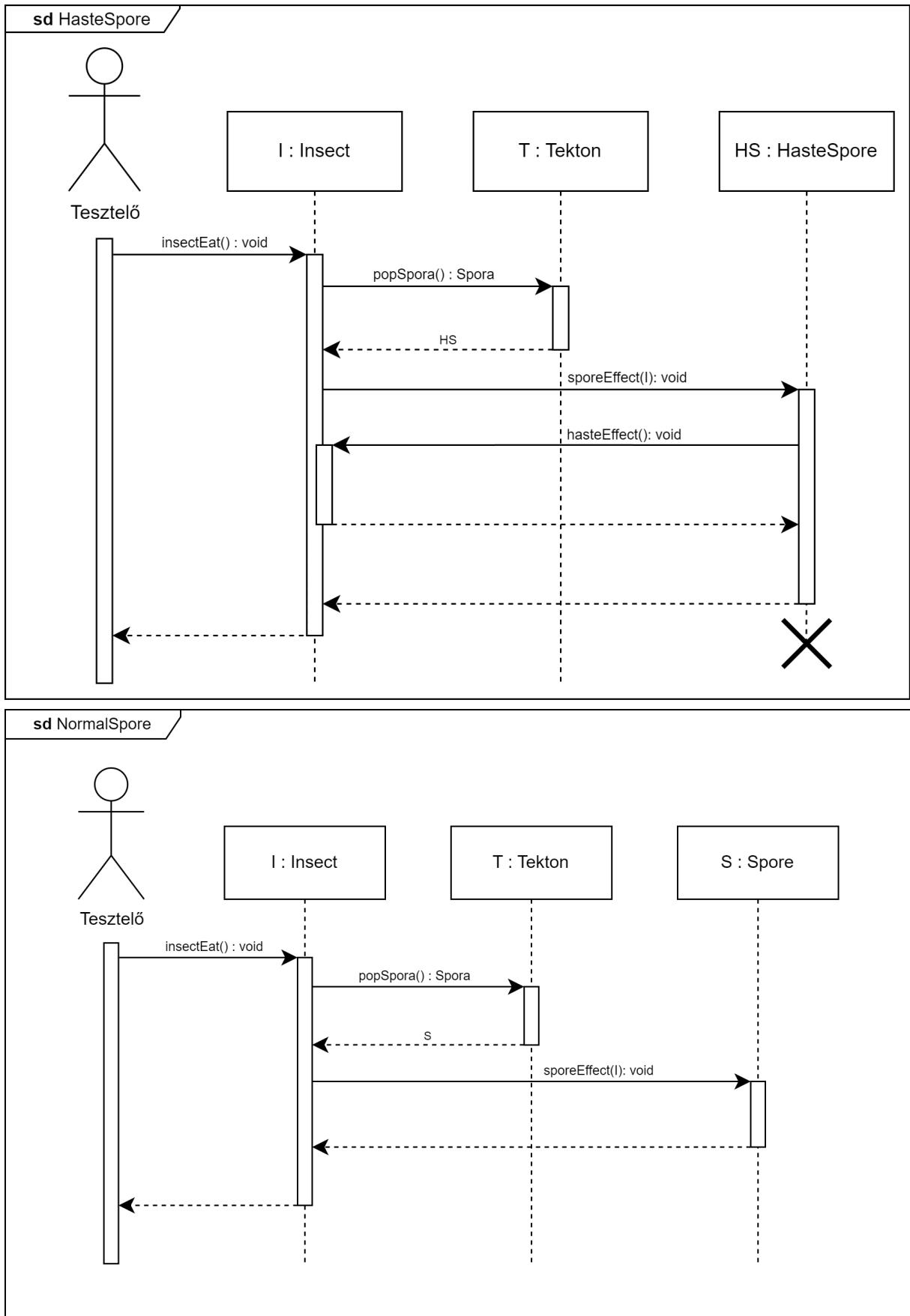


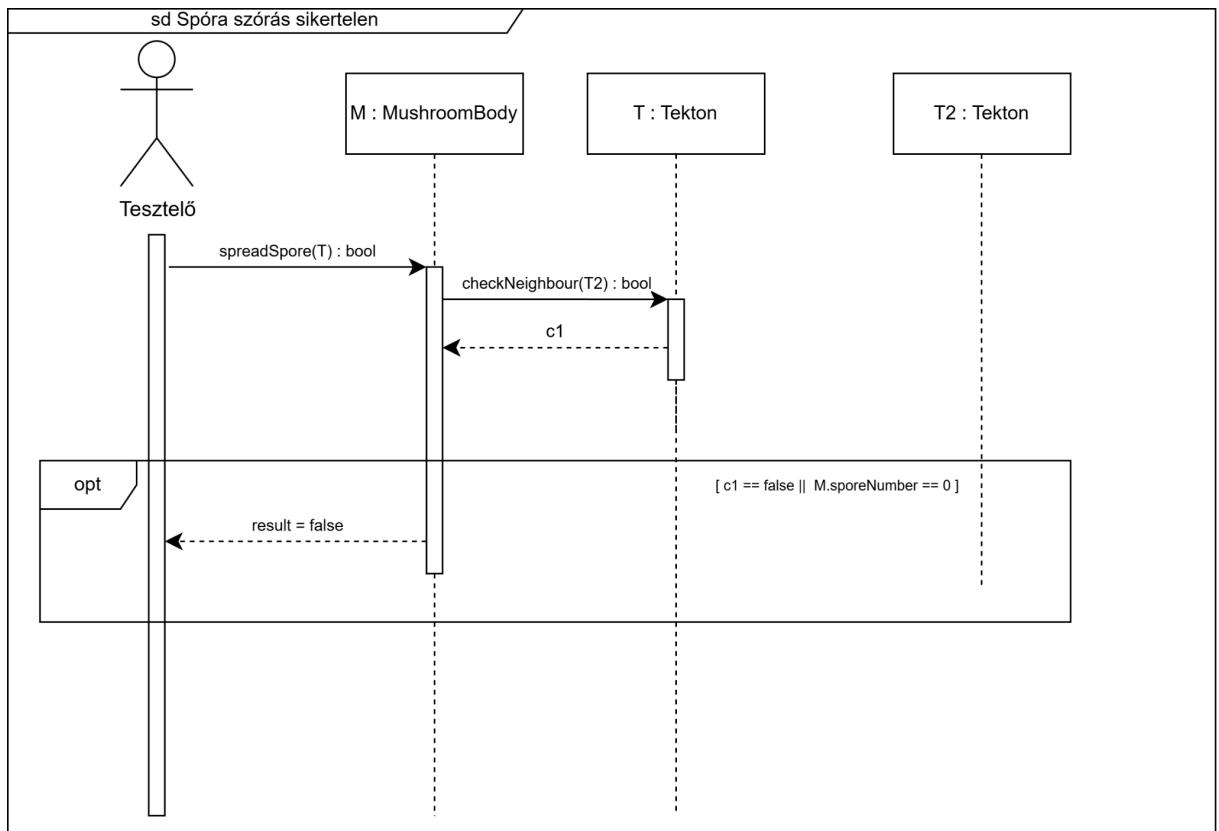


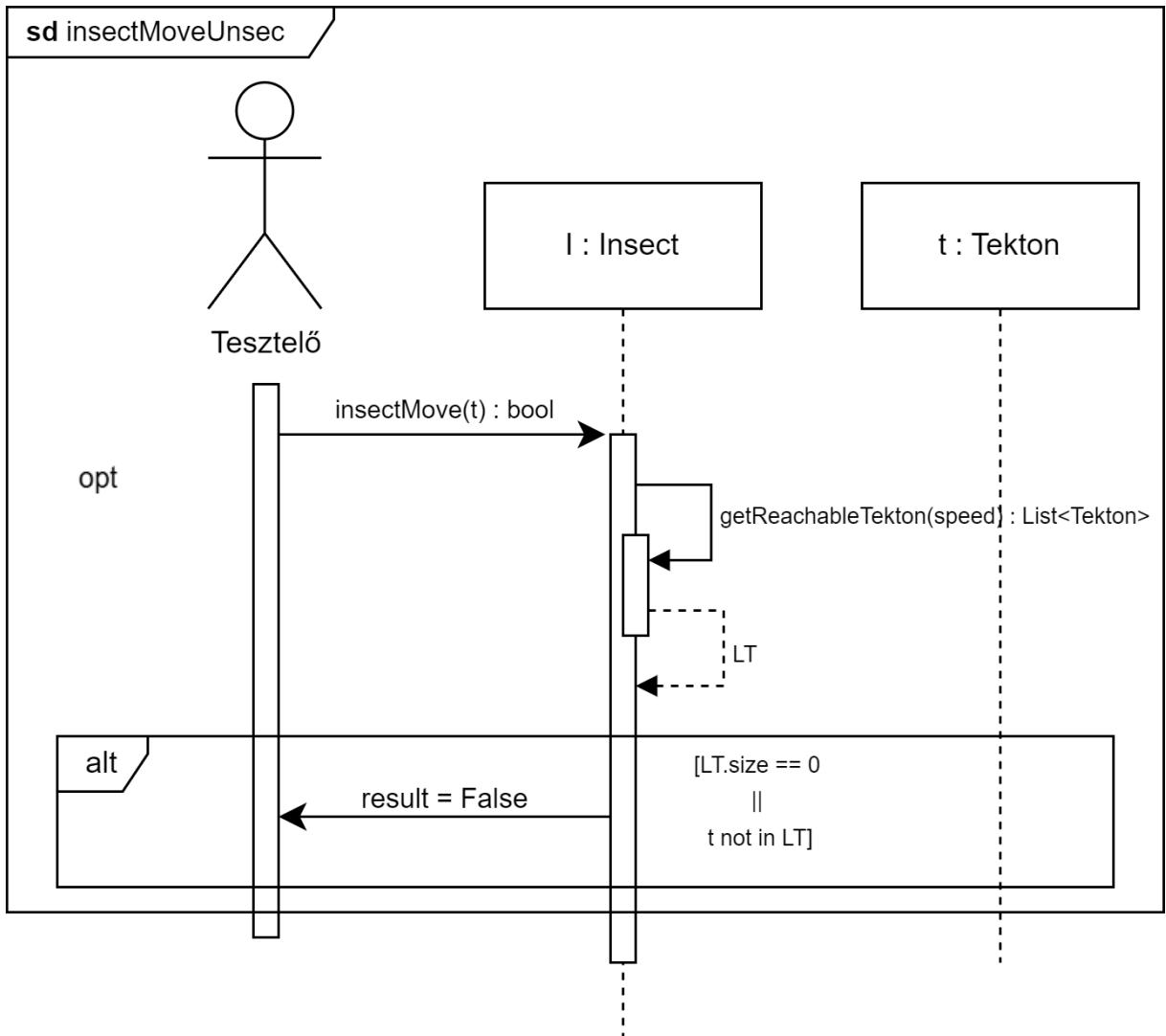


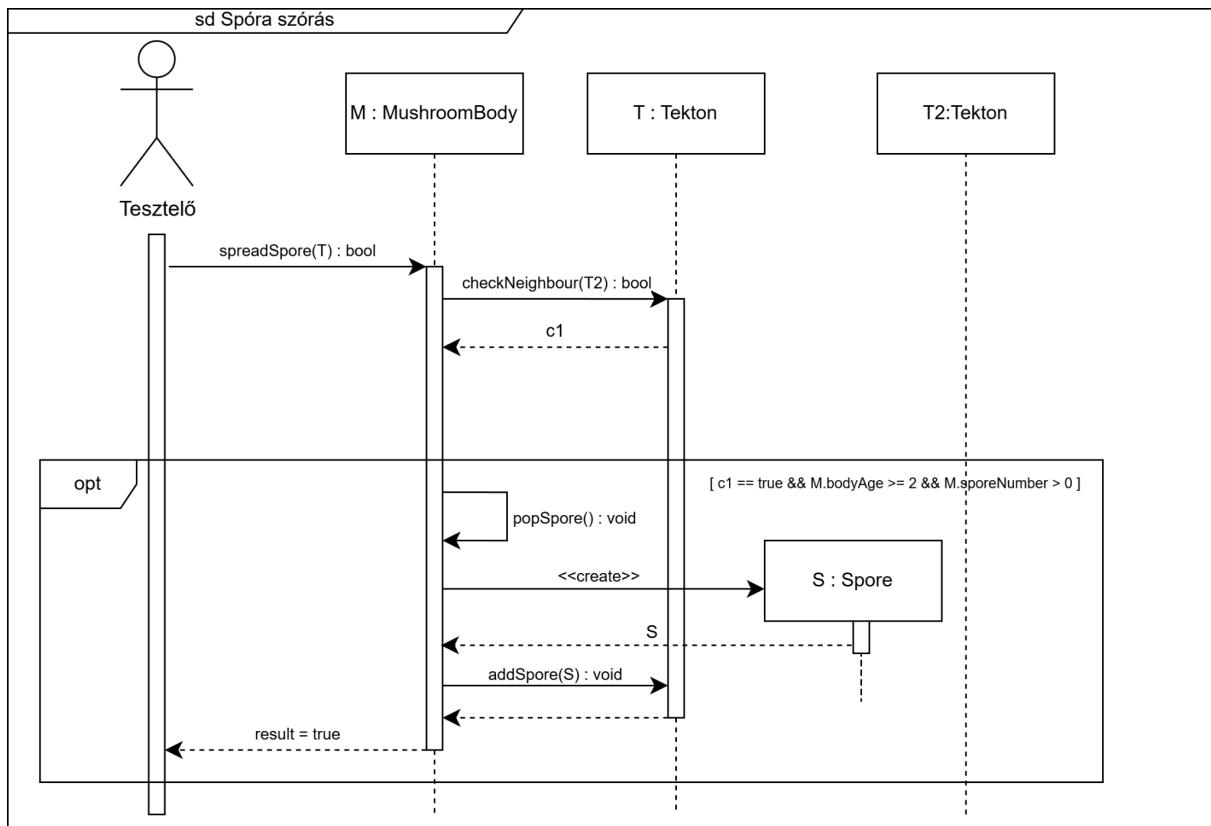






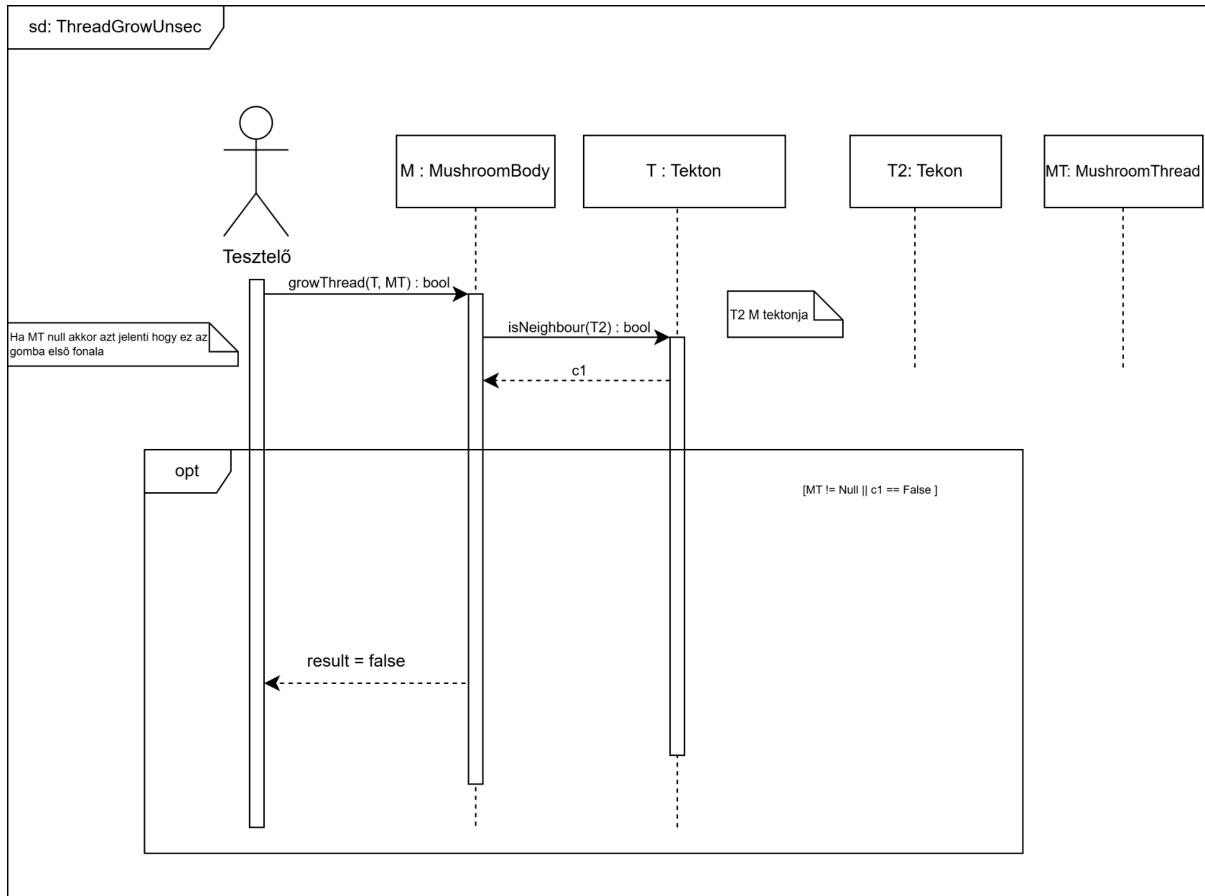
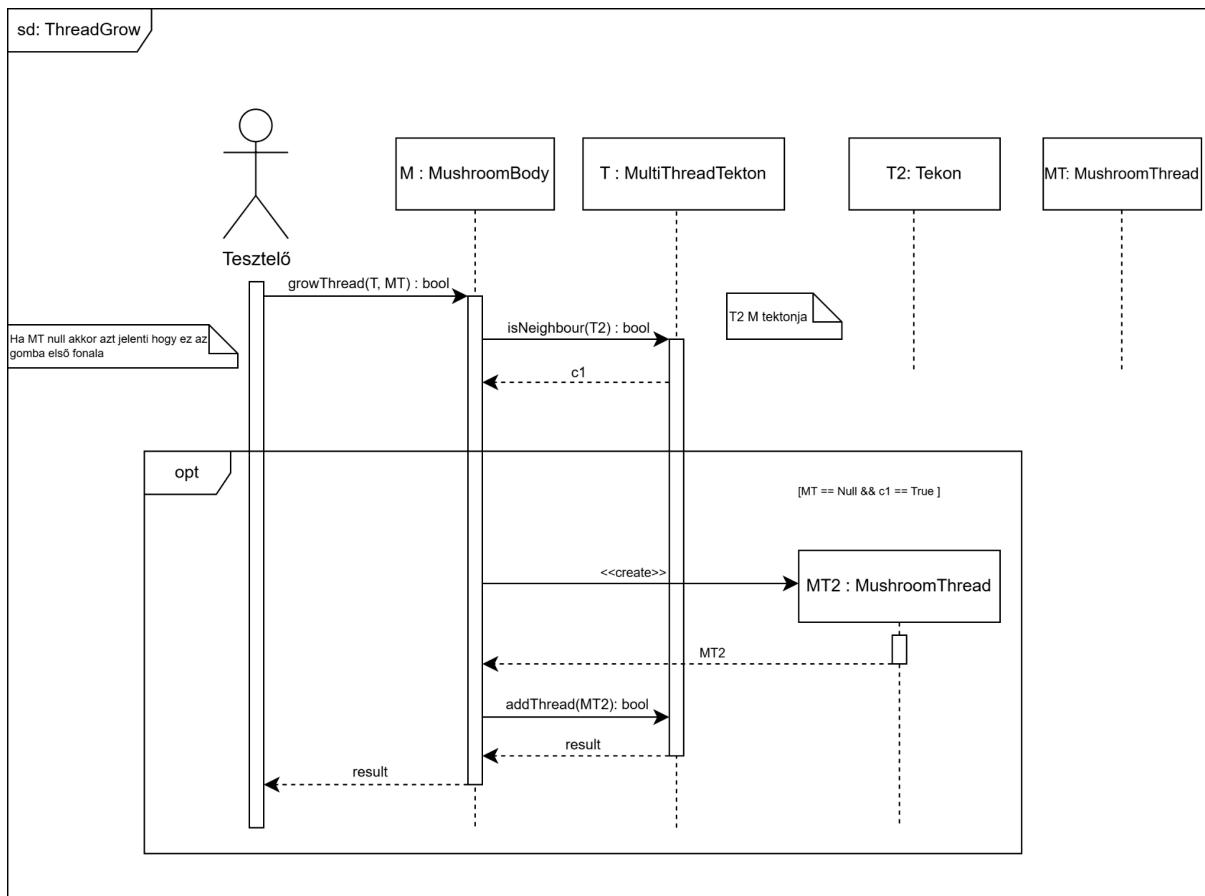




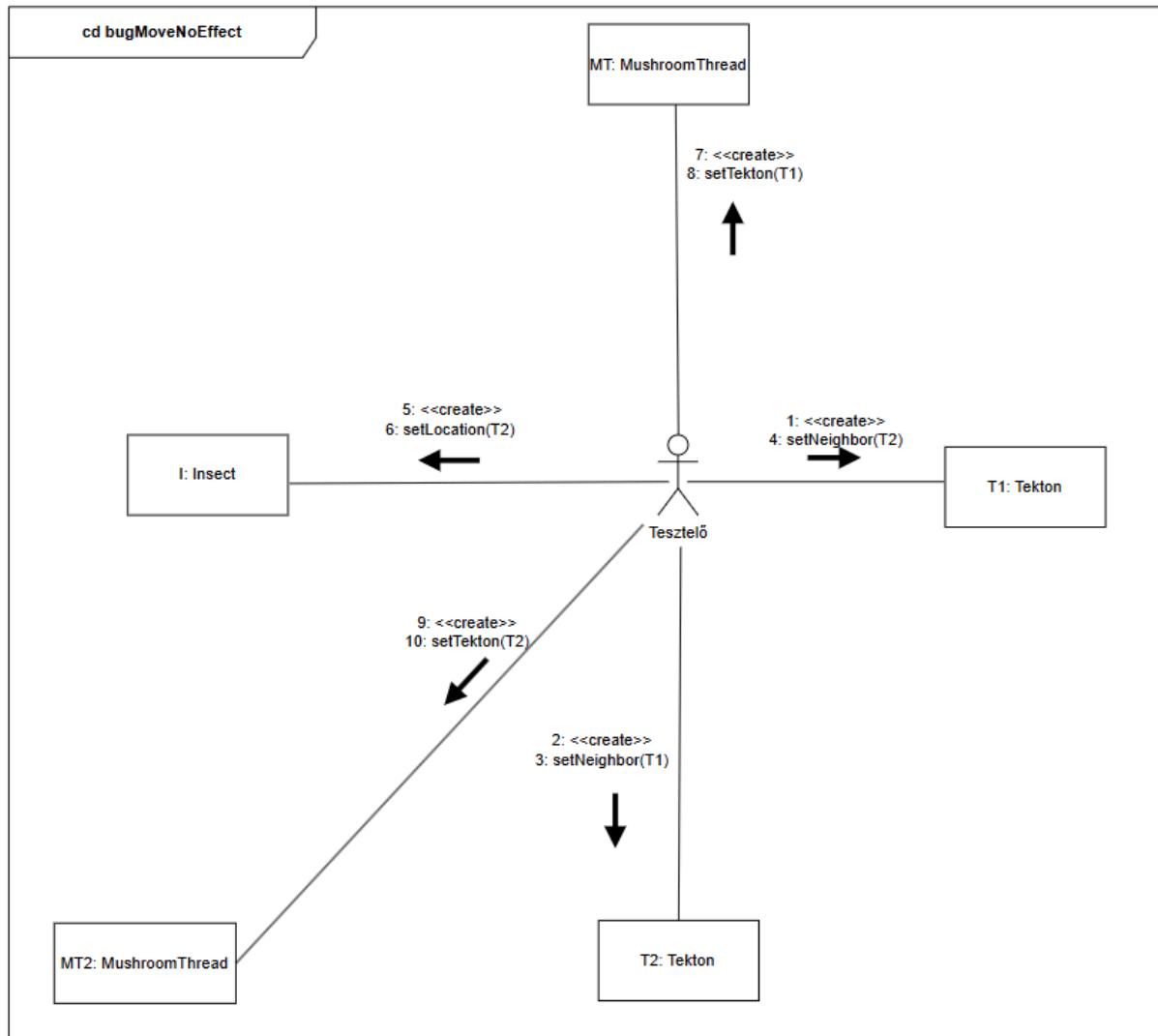


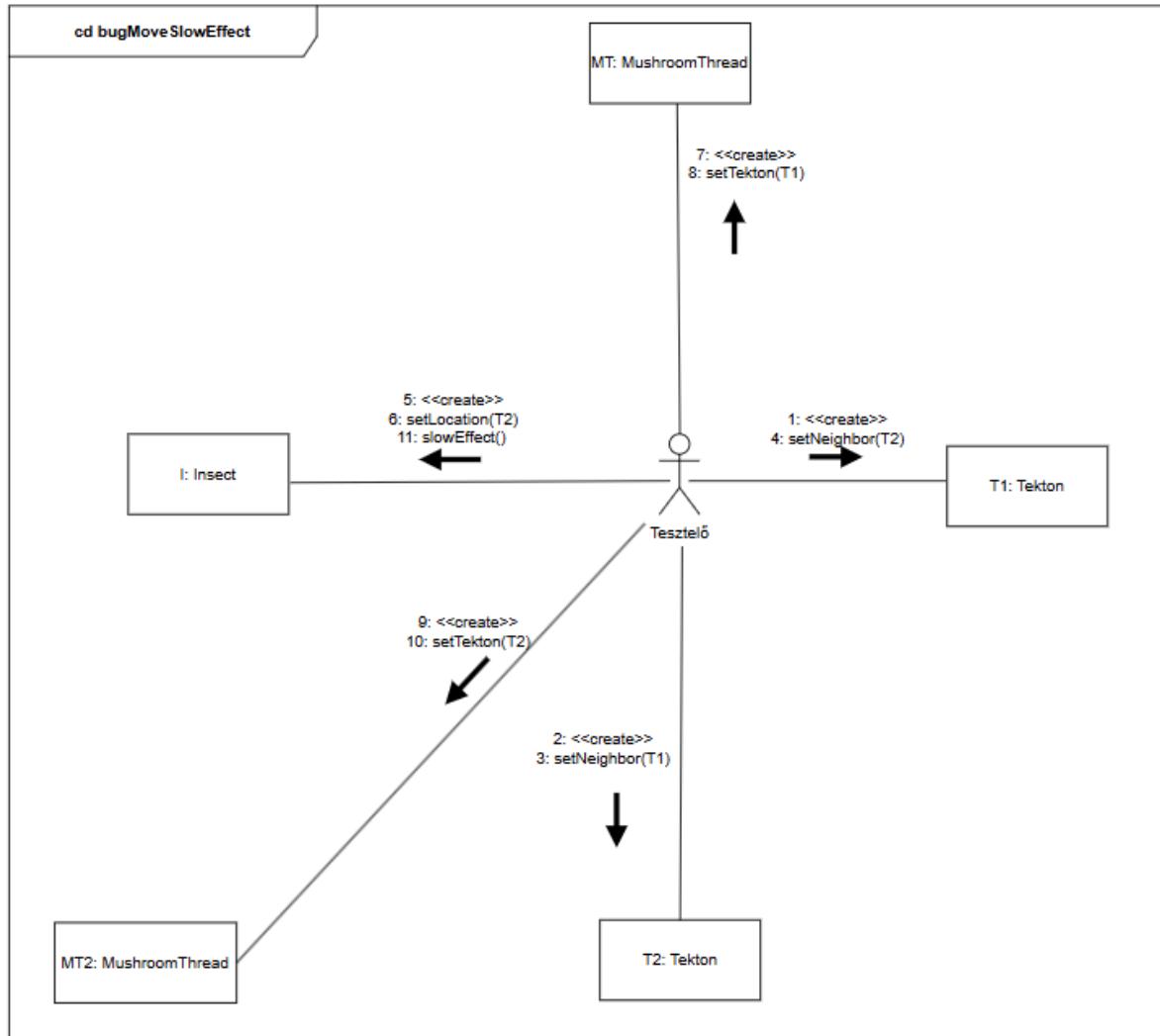
5. Szkeleton tervezése

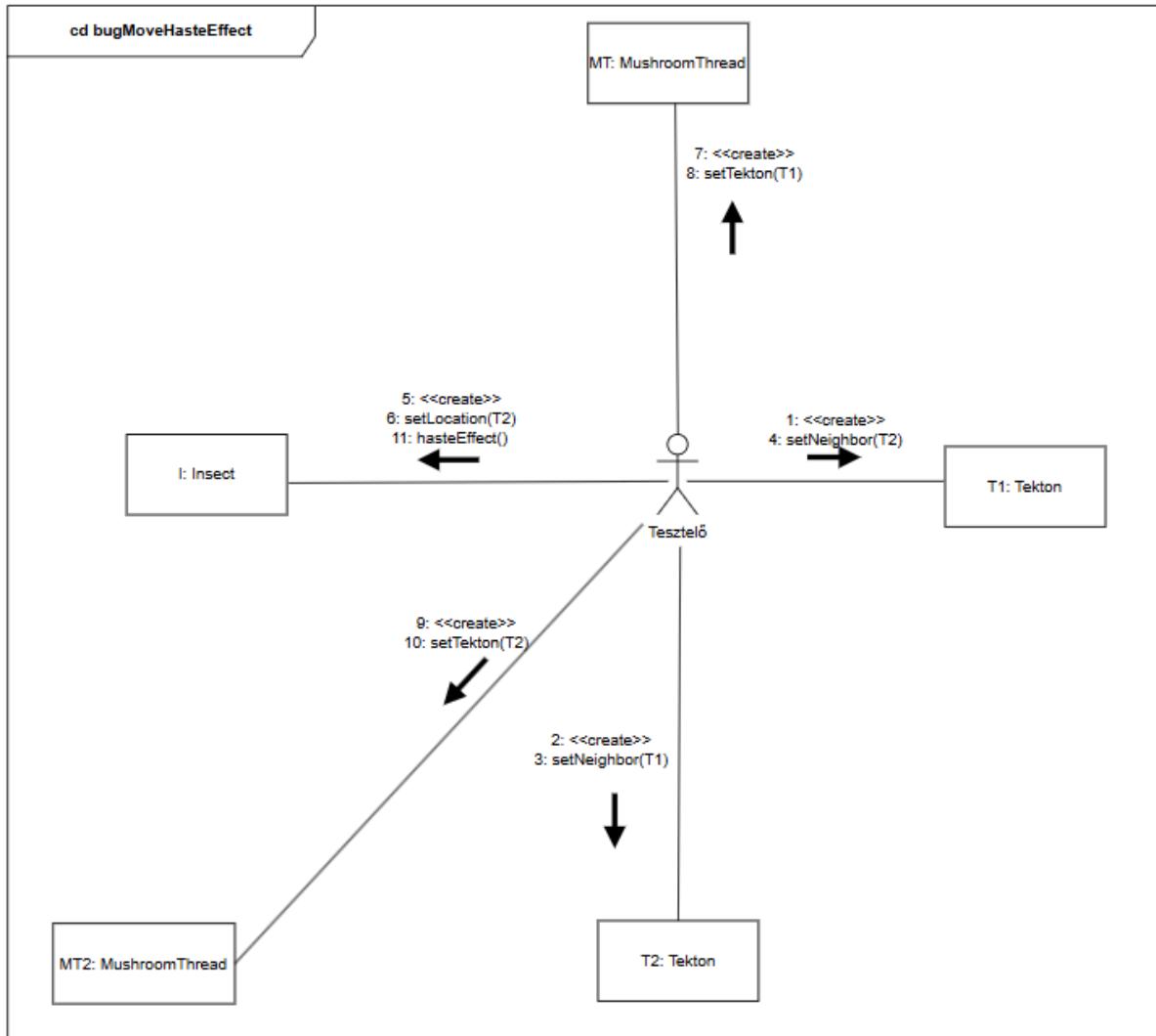
Being_Chilling

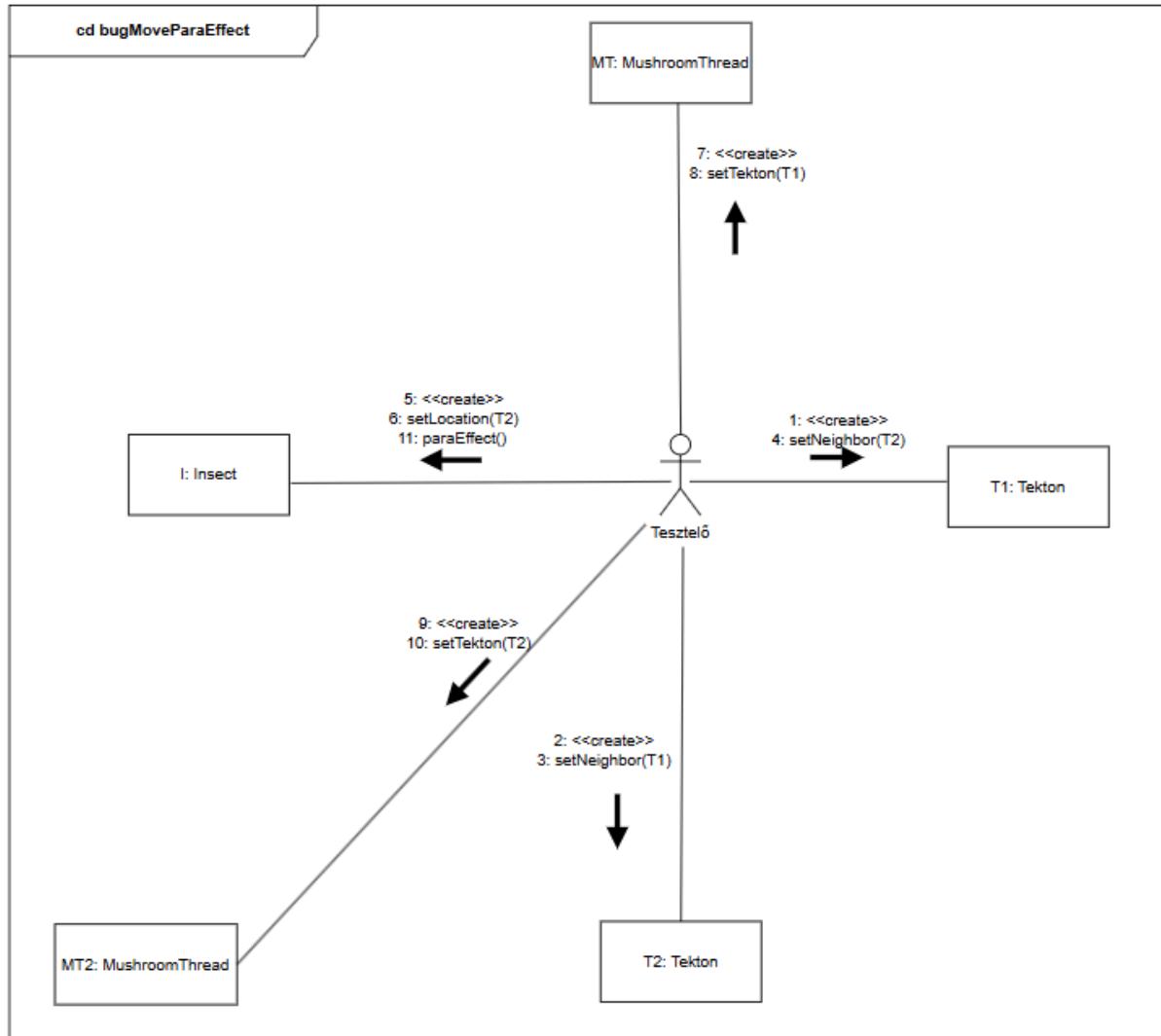


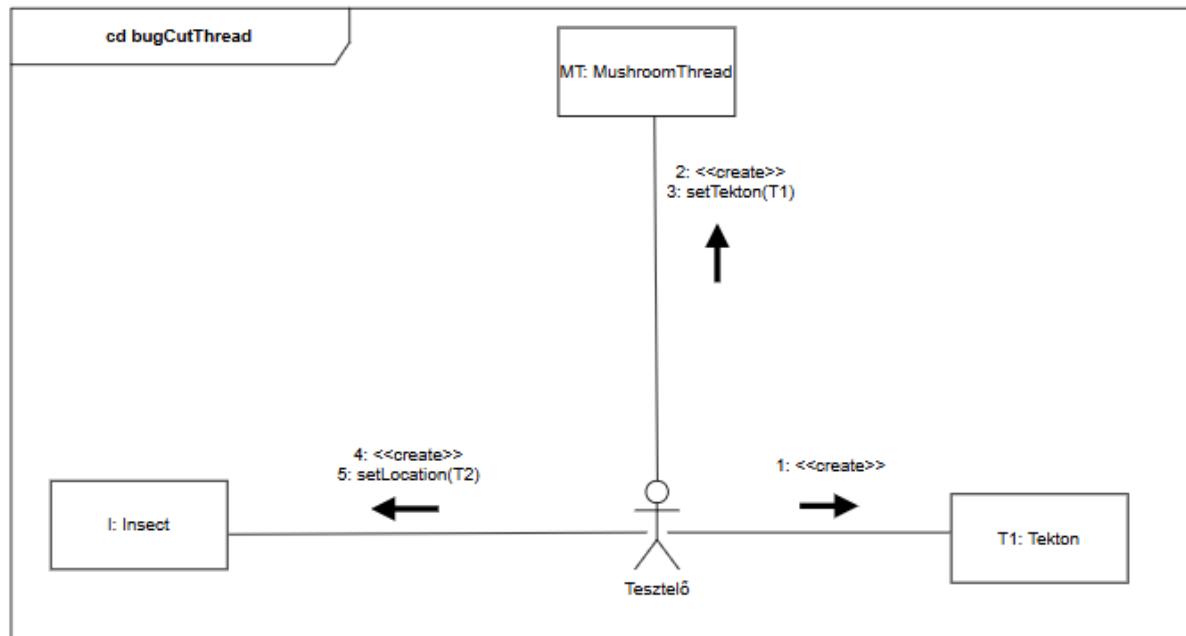
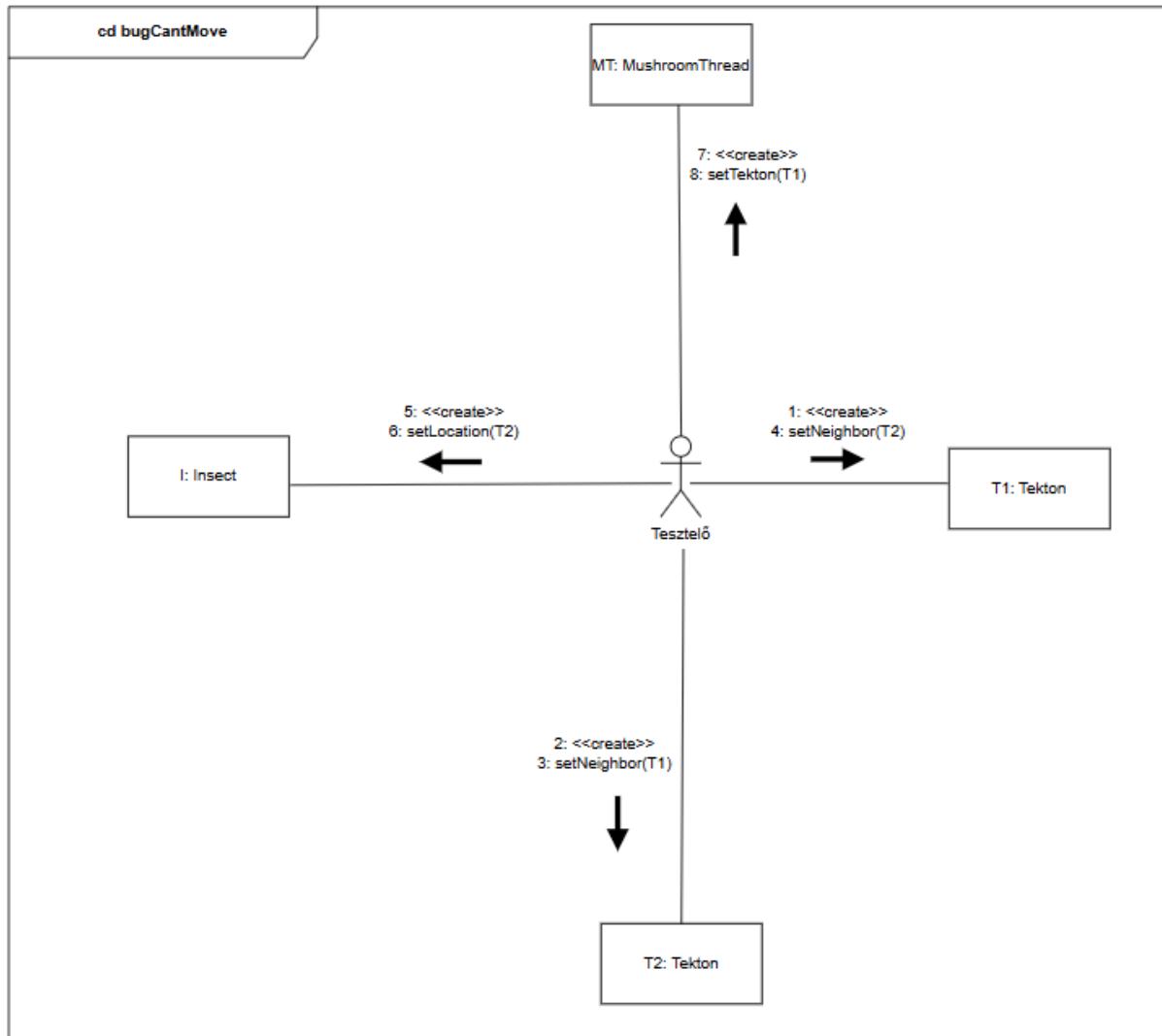
5.4 Kommunikációs diagramok

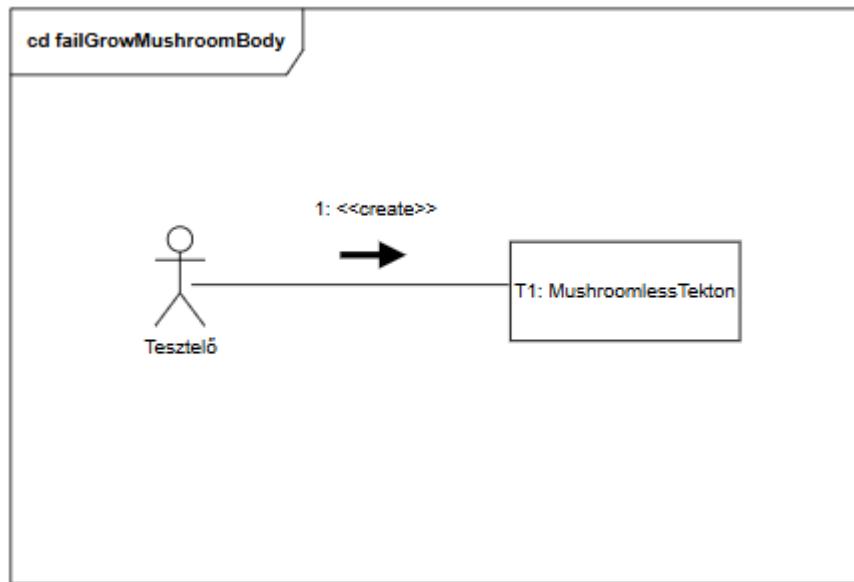
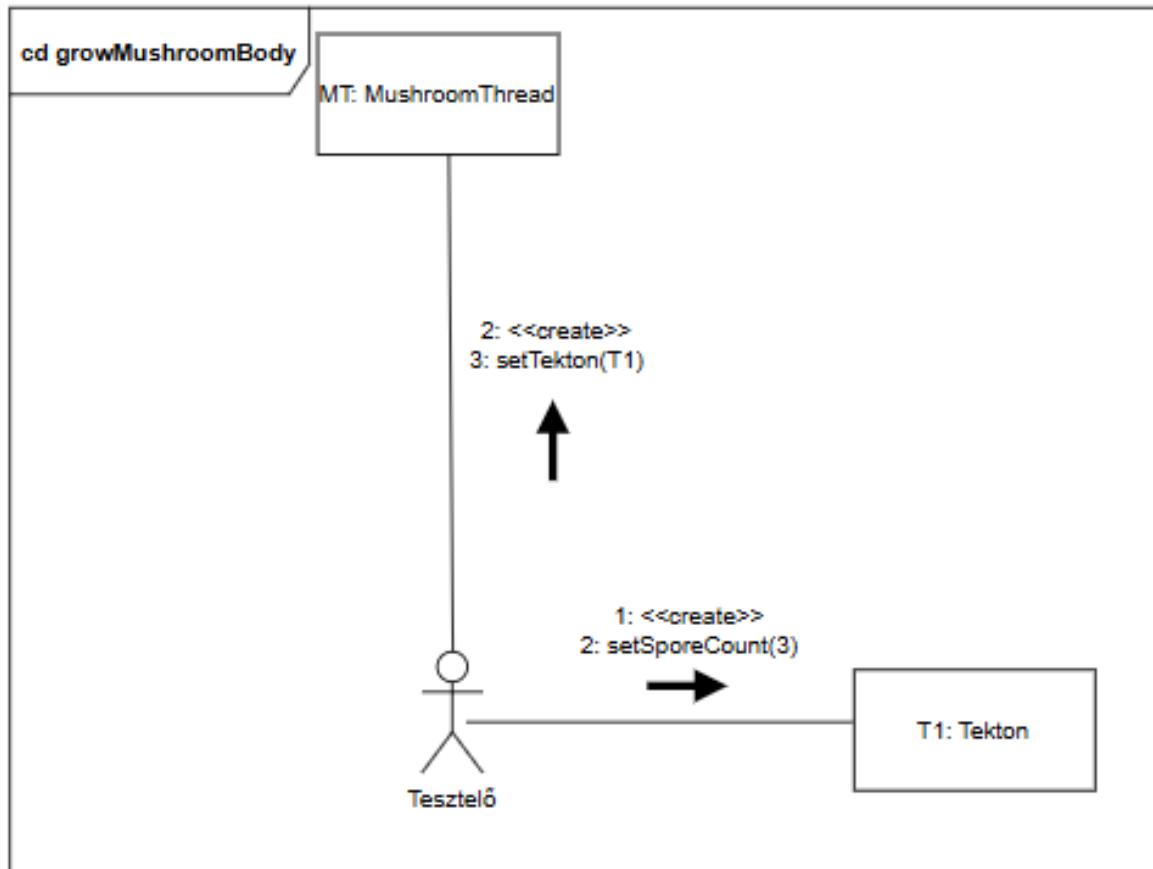


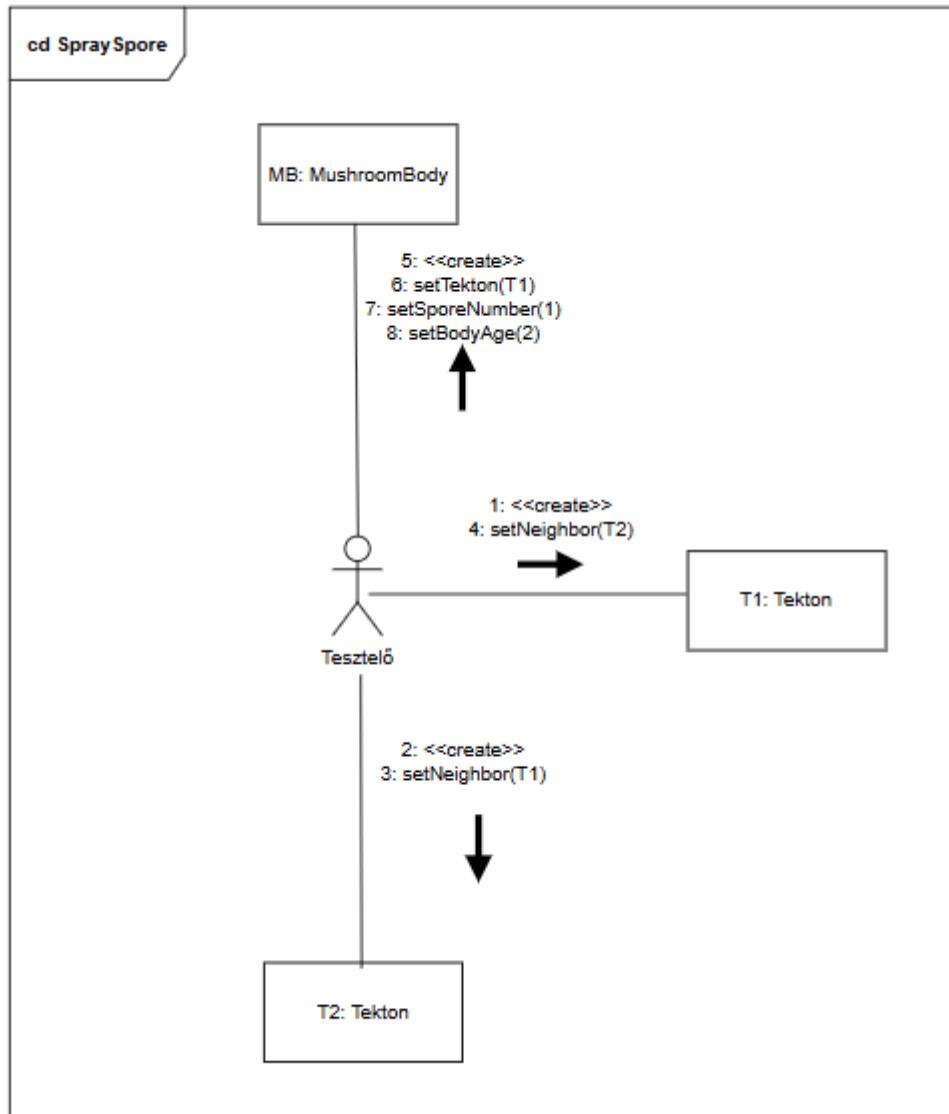


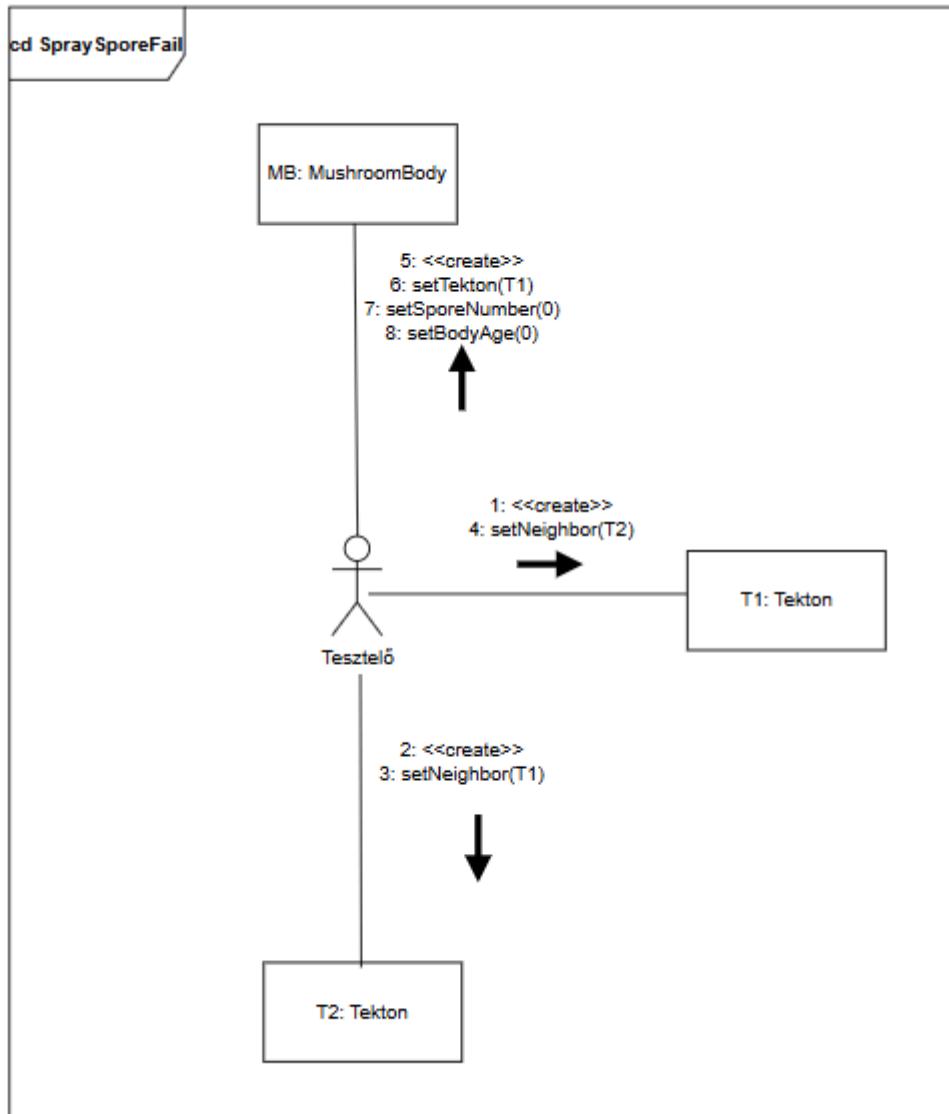


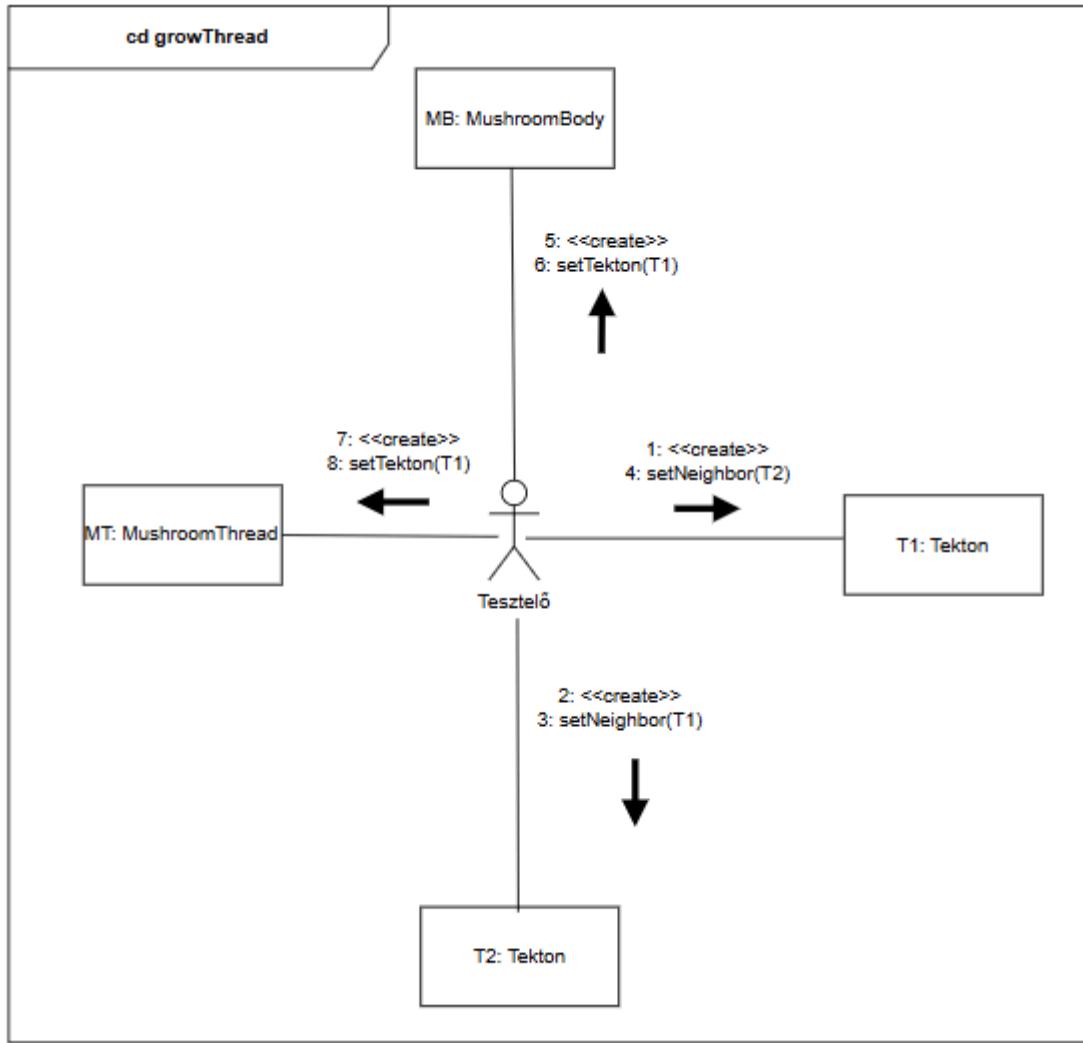


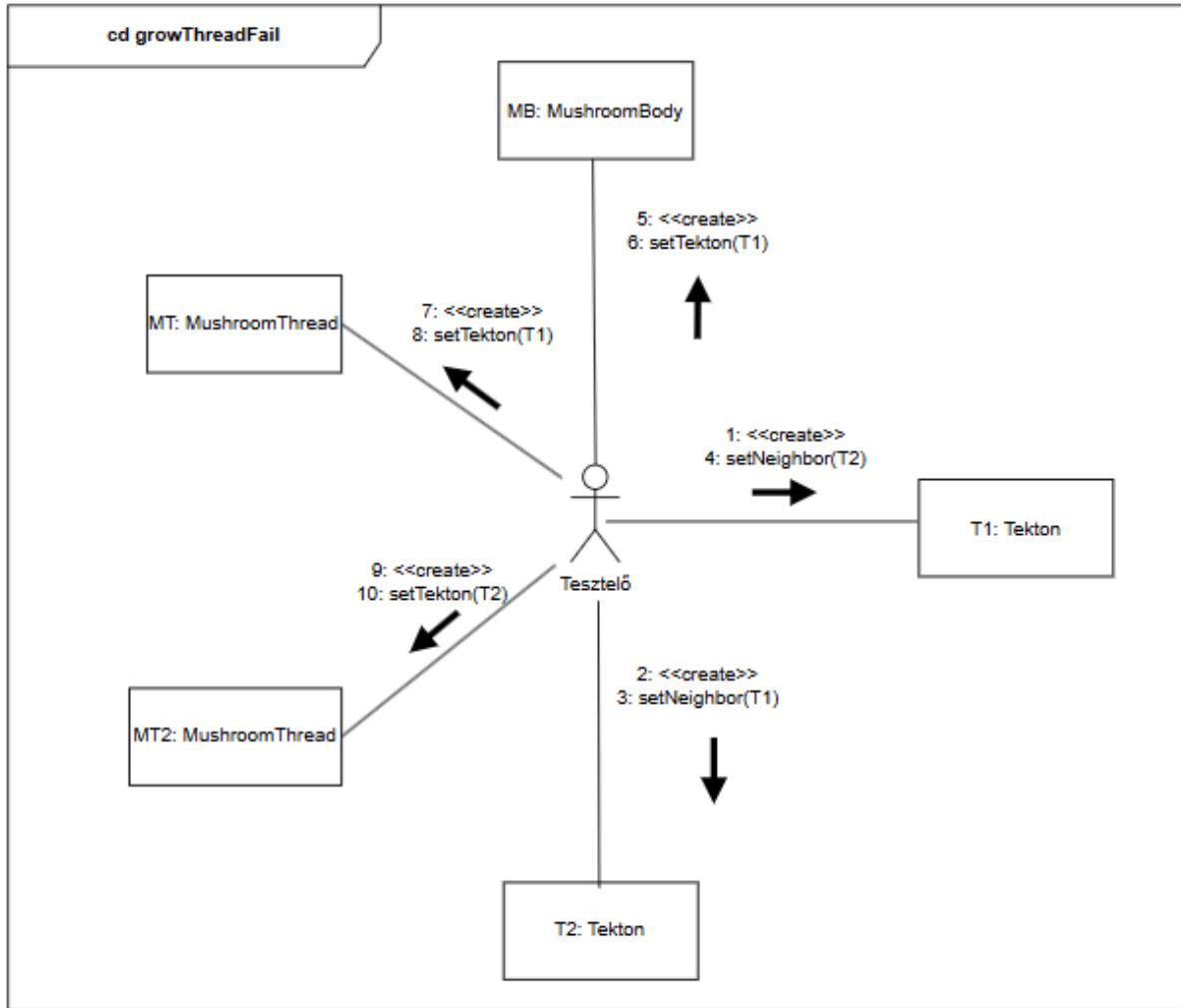


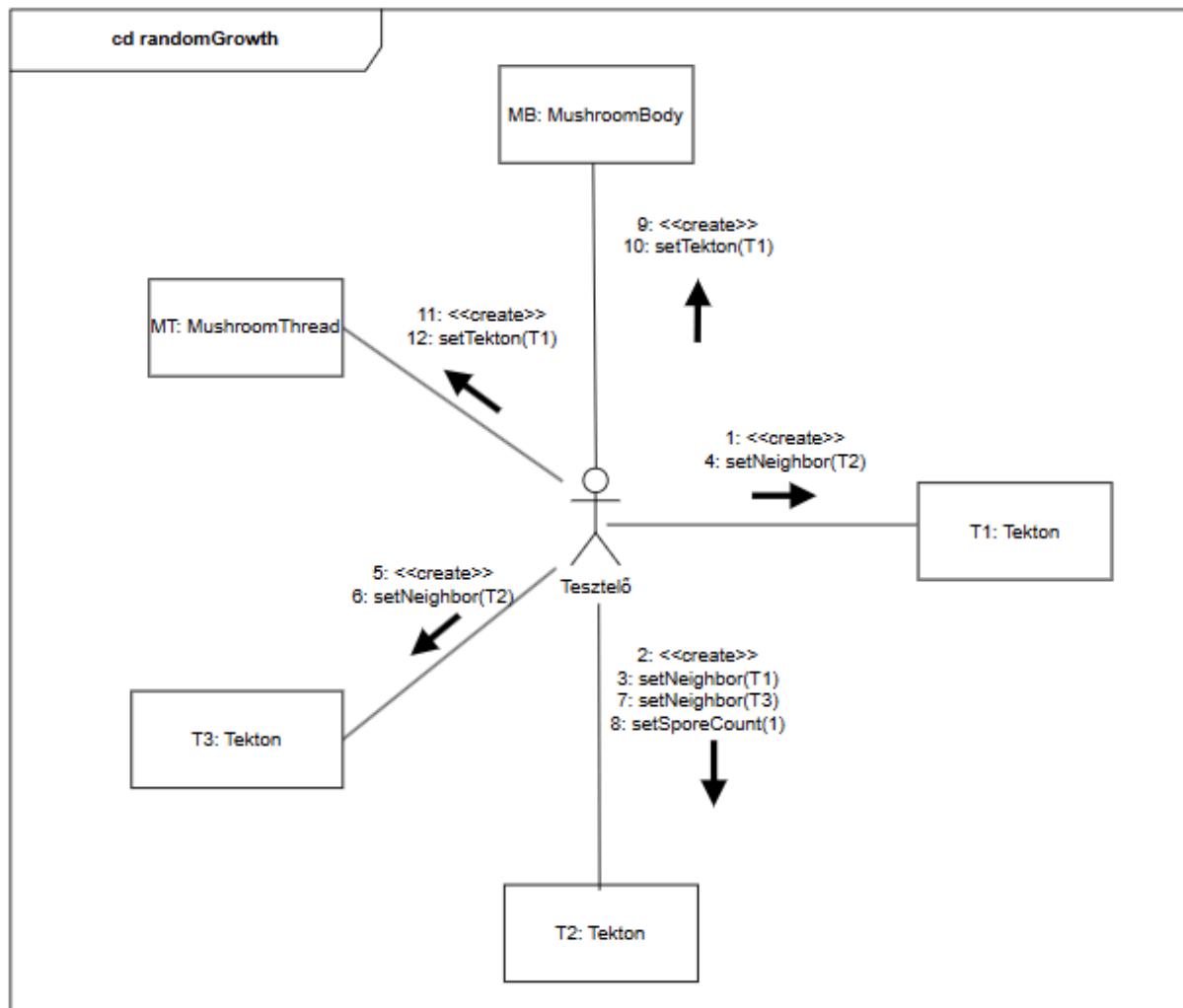


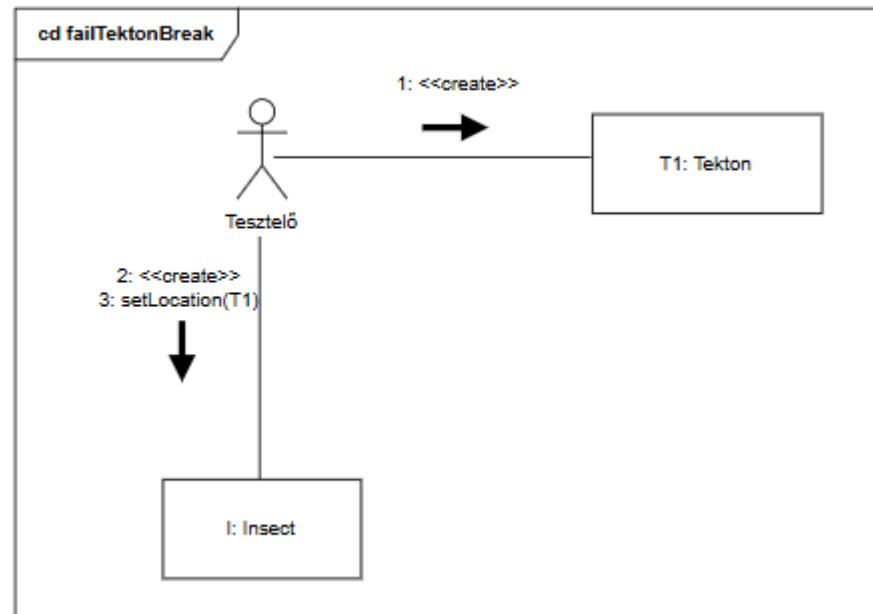
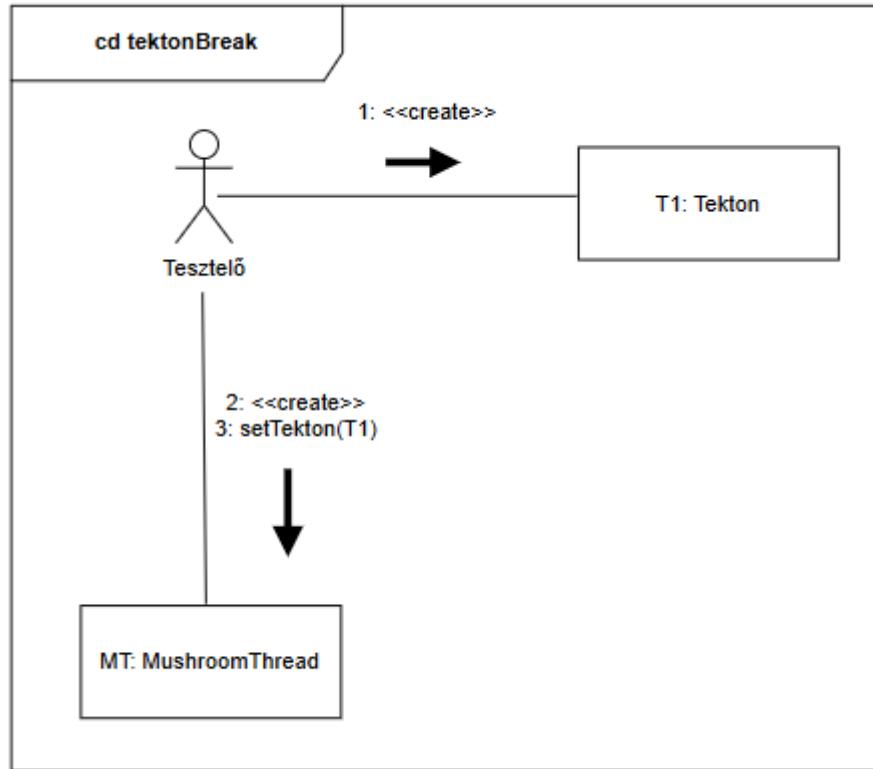


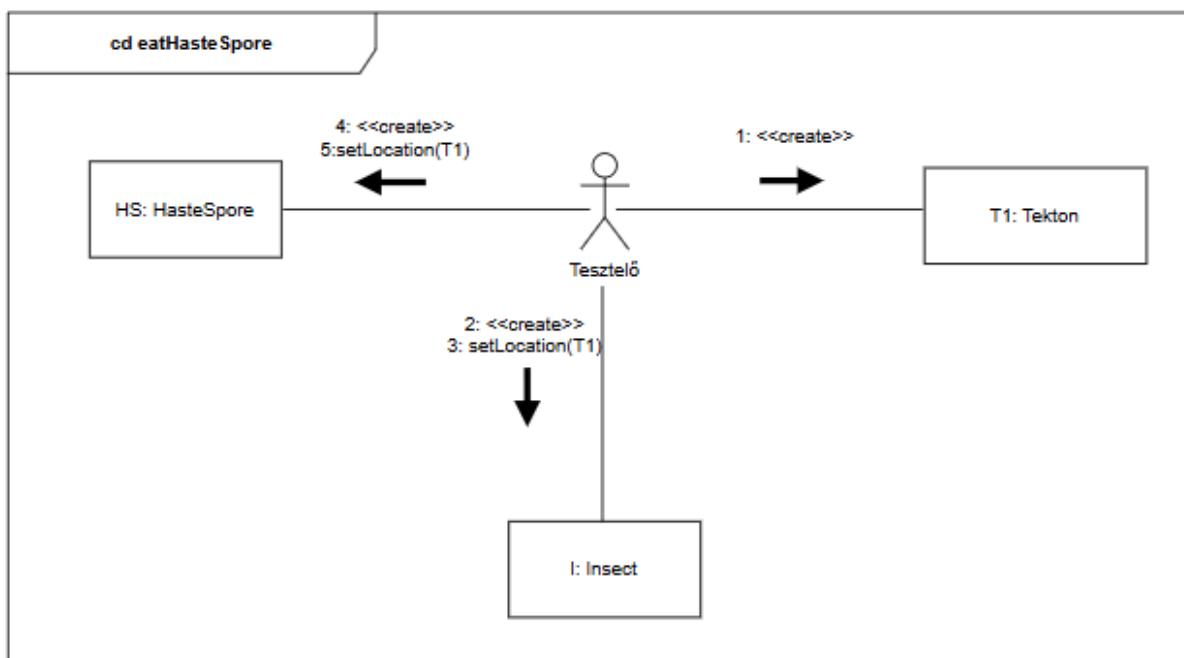
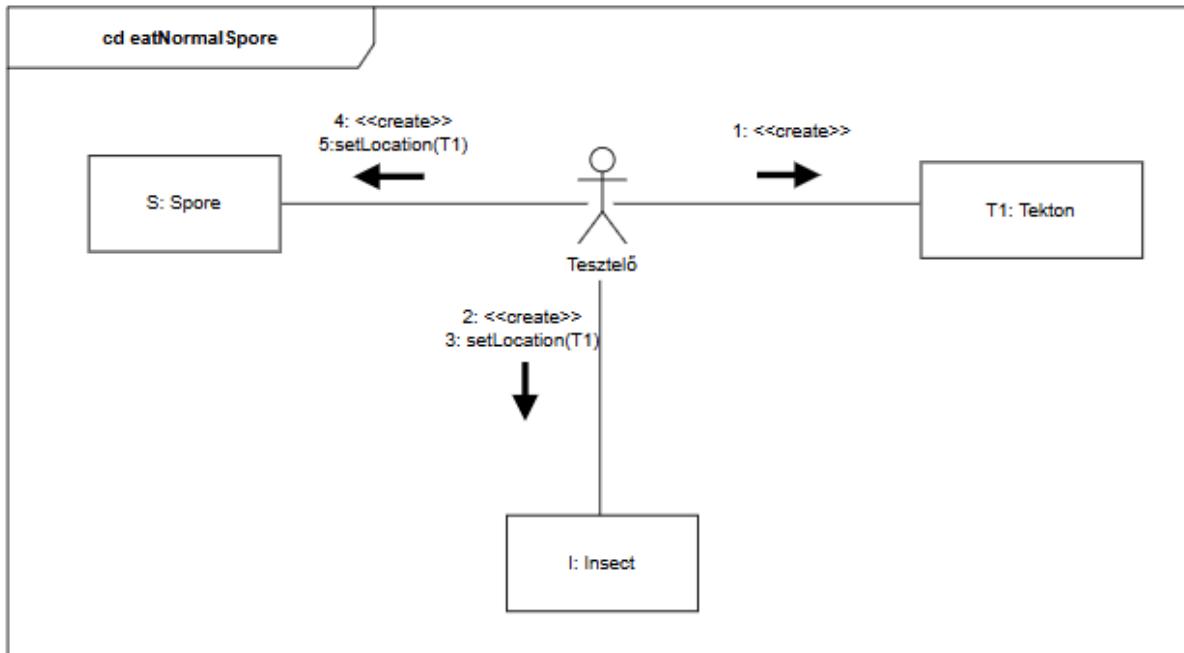


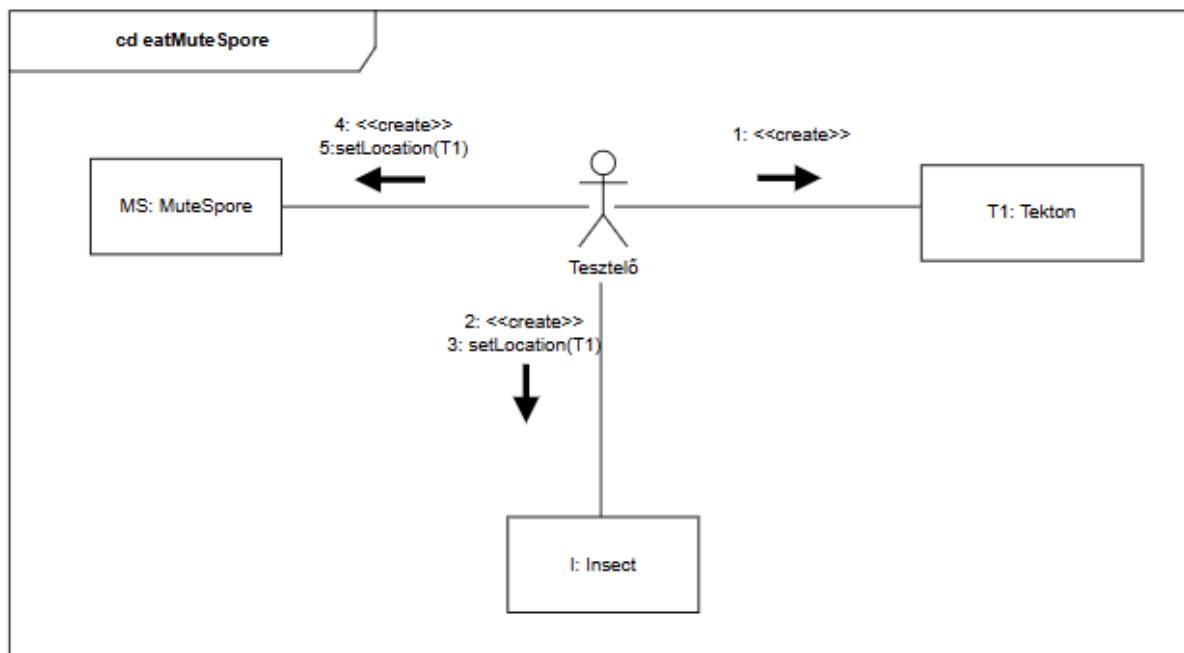
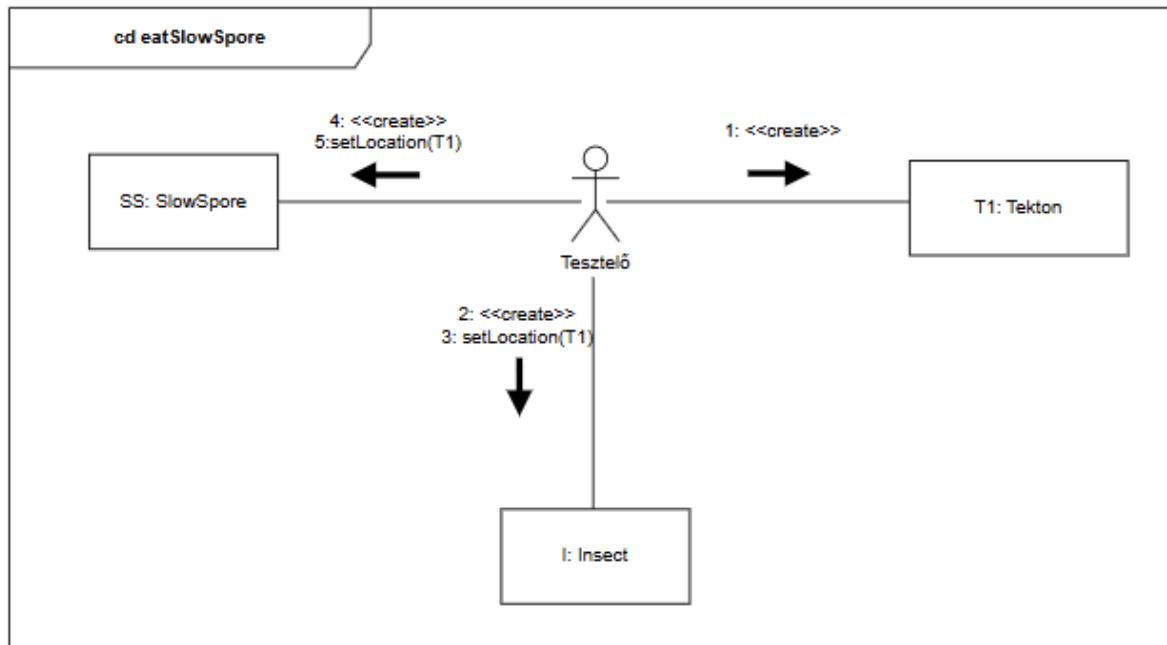


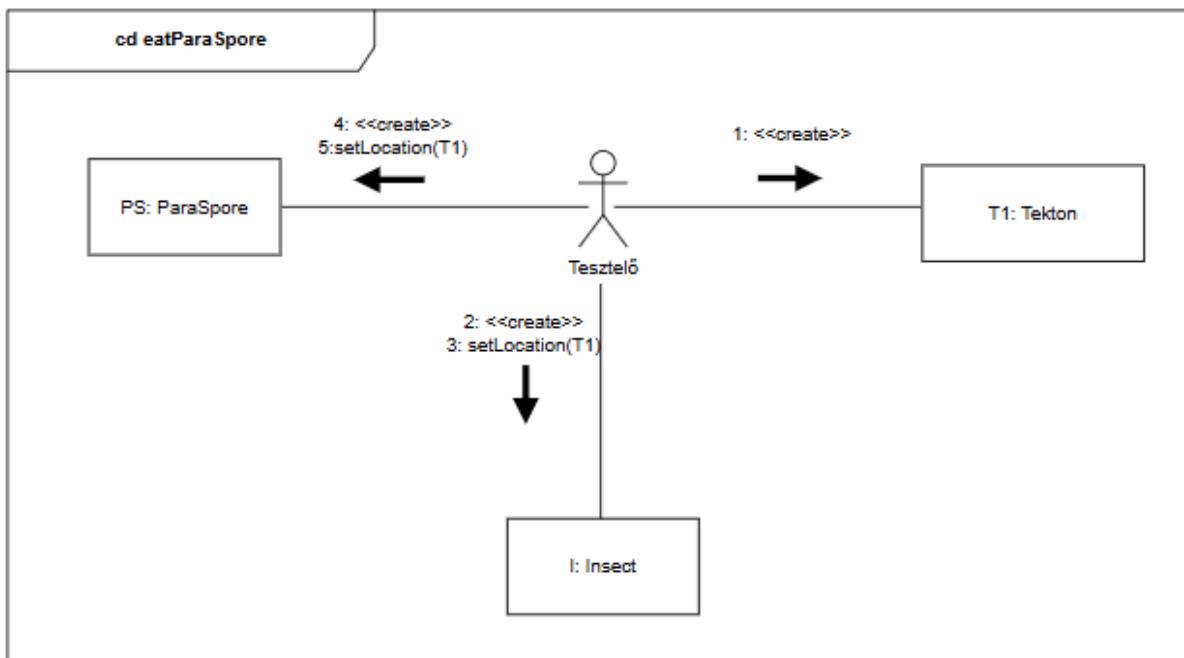












5.5 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2025.03.13. 20:00	3 óra	Bene Ji Jin Yan Ji Yuang Yin Kattan	Értekezlet. A mindenki feladatának megbeszélése, felosztása.
2025.03.14. 10:00	3 óra	Ji Jin Yan Ji Yuang Yin	Értekezlet. 5.1 átbeszélés
2025.03.14 11:00	2 óra	Ji Jin Yan	Tevékenység: 5.1 elkészítés
2025.03.15 20:00	2 óra	Ji Jin Yan	Tevékenység: Szekvencia diagram
2025.03.15 20:00	2 óra	Ji Yuang	Tevékenység: Szekvencia diagram
2025.03.15 21:00	2 óra	Kattan	Tevékenység: Kommunikációs Diagram
2025.03.15 22:00	3 óra	Yin	Tevékenység: Szekvencia diagram
2025.03.16 14:30	5 óra	Kattan	Tevékenység: Kommunikációs Diagram

6. Szkeleton elkészítése

82 – being_chilling

Konzulens:
Tóth Bence

Csapattagok

Yin Jun	TN1KZI	1171276297yin@gmail.com
Bene Máté	K4C4W1	matthew.be04@gmail.com
Kattan Ibrahim Gábor	ZU22JJ	kattanbarhum2@gmail.com
Ji Jin Yan	ZGCKAL	jijinyan0201@gmail.com
Ji Yuang	UI7V2S	yuangji688@gmail.com

2025.03.23.

6. Szkeleton beadás

6.1 Fordítási és futtatási útmutató

6.1.1 Fájllista

Fájl neve	Méret	Keletkezés ideje	Tartalom
build.bat	571 bytes	2025.03.23	a fordításhoz tartozó parancsokat tartalmazza
Fungorium-Projlab.i ml	433 bytes	2025.03.23.	Az intelliJ idea projekt fájlja.
manifest.txt	18 bytes	2025.03.23.	A program main classát jelöli meg.
run.bat	72 bytes	2025.03.23.	A programot futtatja, ha létezik a Fungorium.jar.
src/HasteSpore.java	919 bytes	2025.03.23.	A HasteSpore class-t valósítja meg ez a forrásfájl.
src/Insect.java	5.95 KB	2025.03.23.	A Insect class-t valósítja meg ez a forrásfájl.
src/Main.java	3.98 KB	2025.03.23.	A program futtatásához szükséges forrásfájl.
src/MultiThreadTekt on.java	634 bytes	2025.03.23.	A MultiThreadTekton class-t valósítja meg ez a forrásfájl.
src/MushroomBody.j ava	3.47 KB	2025.03.23.	A MushroomBody class-t valósítja meg ez a forrásfájl.
src/MushroomlessTe kton.java	693 bytes	2025.03.23.	A MushroomlessTekton class-t valósítja meg ez a forrásfájl.
src/MushroomSpecie s.java	1.22 KB	2025.03.23.	A MushroomSpecies class-t valósítja meg ez a forrásfájl.
src/MushroomThread .java	2.32 KB	2025.03.23.	A MushroomThread class-t valósítja meg ez a forrásfájl.
src/MuteSpore.java	914 bytes	2025.03.23.	A MuteSpore class-t valósítja meg ez a forrásfájl.
src/ParaSpore.java	910 bytes	2025.03.23.	A ParaSpore class-t valósítja meg ez a forrásfájl.
src/SlowSpore.java	909 bytes	2025.03.23.	A SlowSpore class-t valósítja meg ez a forrásfájl.
src/Spore.java	1.36 bytes	2025.03.23.	A Spore class-t valósítja meg ez a forrásfájl.
src/Szkeleton.java	8.51 KB	2025.03.23.	A szkeleton forgatókönyvét valósítja meg.
src/Tekton.java	9.77 KB	2025.03.23.	A Tekton class-t valósítja meg ez a forrásfájl.
src/ThreadAbsorbTek ton.java	543 bytes	2025.03.23.	A ThreadAbsorbTekton class-t valósítja meg ez a forrásfájl.

6.1.2 Fordítás

A java kód fordításához szükséges JDK telepítése és a forráskódok megléte! A kód fordításához az adott build.bat fájlt kell futtatni, ami elkészíti a .class fájlokat, a bin mappába, és a konzolban futtatható Fungorium.jar fájlt.

6.1.3 Futtatás

A futtatáshoz szükséges a JDK telepítése! Ha a Fungorium.jar fájl megvan, akkor a run.bat file-t kell futtatni, ami elindítja a programot.

6.2 Értékelés

Tag neve	Tag neptun	Munka százalékban
Yin Jun	TN1KZI	20%
Ji Jin Yan	ZGCKAL	20%
Bene Máté	K4C4W1	20%
Kattan Ibrahim Gábor	ZU22JJ	20%
Ji Juang	UI7V2S	20%

6.3 Napló

Kezdet	Időtartam	Résznevők	Leírás
2010.03.21. 18:00	2,5 óra	Horváth Németh Tóth Oláh	Értekezlet. Döntés: Horváth elkészíti az osztálydiagramot, Oláh a use-case leírásokat.
2025.03.21. 12:00	5 óra	Kattan	Tevékenység: Kommunikációs Diagramok elkészítése
2025.03.21. 20:00	4 óra	Ji Jin Yan	Tevékenység: Mushroombody, MushroomThread MushroomSpecies class készites
2025.03.22 19:00	3 óra	Ji Jin Yan	Tevékenység: Komment
2025.03.23 16:00	1 óra	Kattan	Tevékenység: Spore, MuteSpore, SlowSpore, HasteSpore és ParaSpore classok elkészítése
...

7. Prototípus koncepciója

82 – being_chilling

Konzulens:
Tóth Bence

Csapattagok

Yin Jun	TN1KZI	1171276297yin@gmail.com
Bene Máté	K4C4W1	matthew.be04@gmail.com
Kattan Ibrahim Gábor	ZU22JJ	kattanbarhum2@gmail.com
Ji Jin Yan	ZGCKAL	jijinyan0201@gmail.com
Ji Yuang	UI7V2S	yuangji688@gmail.com

2025.03.30.

7. Prototípus koncepciója

7.0 Változás hatása a modellre

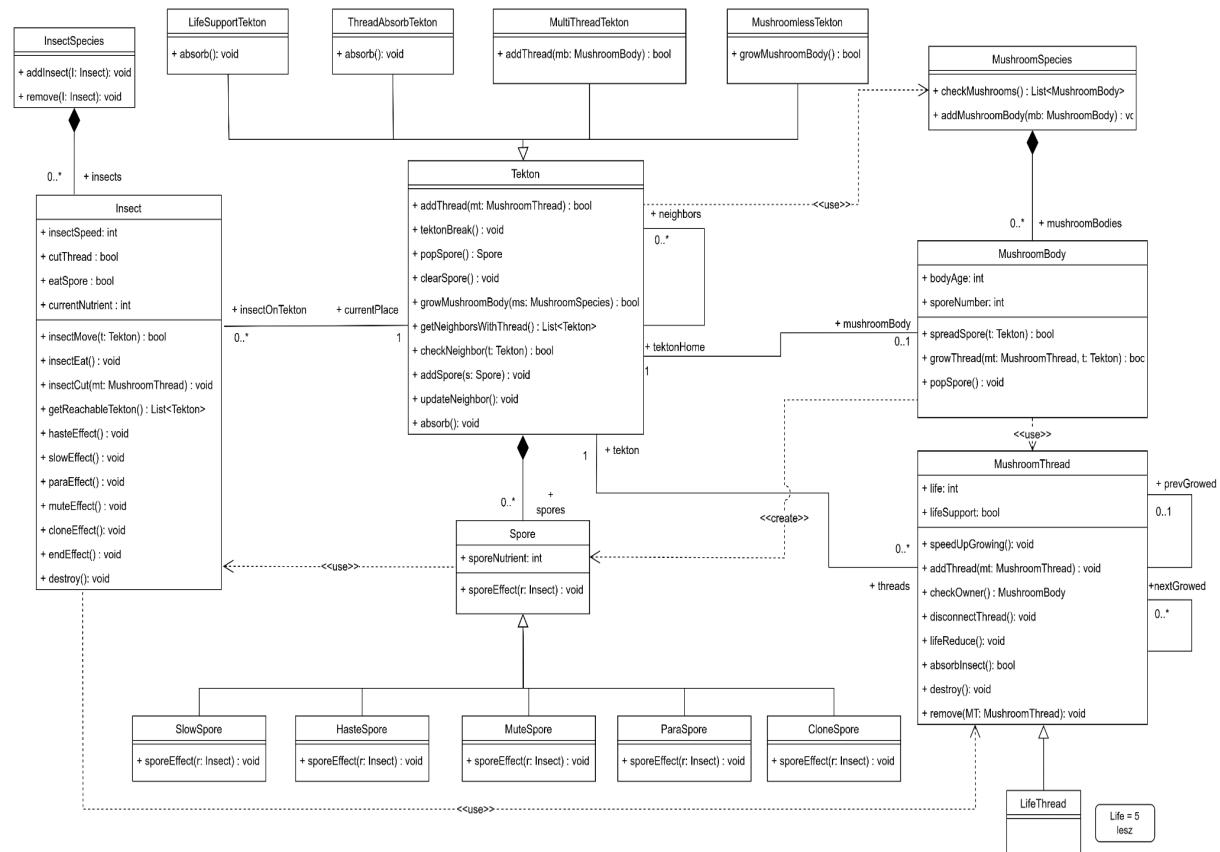
7.0.1 Módosult osztálydiagram

7.0.2 Új vagy megváltozó metódusok

Insect.cloneEffect() : void

Klónozza a rovart, a klón spóra megevése után. Ilyenkor keletkezik még egy, az előzőtől függetlenül élő rovar, akinek a rovarásza megegyezik az eredeti rovar rovarászával.

Insect.destroy(): void



A rovar destruktora, elvégzi a szükséges tisztítási lépéseket, például megszünteti a rovarra mutató referenciakat más osztályokban(null-ra állítja vagy eltávolítja egy listából).

CloneSpore.sporeEffect() : void

Klónozza az adott rovat.

InsectSpecies.addInsect(I) :void

Felvesz egy új rovat a listába.

InsectSpecies.remove(Insect) :void

Eltávolítja az adott rovat a listából.

Tekton.absorb(): void

Minden körben fogjuk hívni az összes fonalas tektonra ezt a metódust, ekkor minden, a tektonon lévő fonalnak ellenőrizzük a lifeSupport attribútumát: ha az értéke igaz (true), nem teszünk semmit, de ha hamis (false), akkor csökkentjük a fonal életét 1-gyel. Amikor egy fonal élete eléri a 0-t, meghívjuk a 'fonal' destrukturát.

ThreadAbsorbTekton esetén override-olja a függvényt, és akár van lifeSupport vagy nem, csökkentjük a fonal életét.

LifeSupportTekton esetén override-olja a függvényt, és életben tartja a rajta lévő fonalt, és ezért, nem csökkentjük a életét.

MushroomThread.disconnectThread() : void

Leválasztja az aktuális fonalat az őt megelőző fonaltól. Ezt úgy éri el, hogy az aktuális fonal prev attribútumát null-ra állítja, és eltávolítja az aktuális fonalat az előző fonal next listájából.

MushroomThread.LifeReduce() : void

1-el csökkent a fonal életét.

MushroomThread.AbsorbInsect : bool

Megeszi a tektonon található bénult rovarokat. Ilyenkor a rovar elpusztul, a fonal pedig gombatestet növeszthet.

MushroomThread.destroy() : void

A fonal destruktora, elvégzi a szükséges tisztítási lépéseket, például megszünteti a fonalra mutató referenciait más osztályokban(null-ra állítja vagy eltávolítja egy listából)

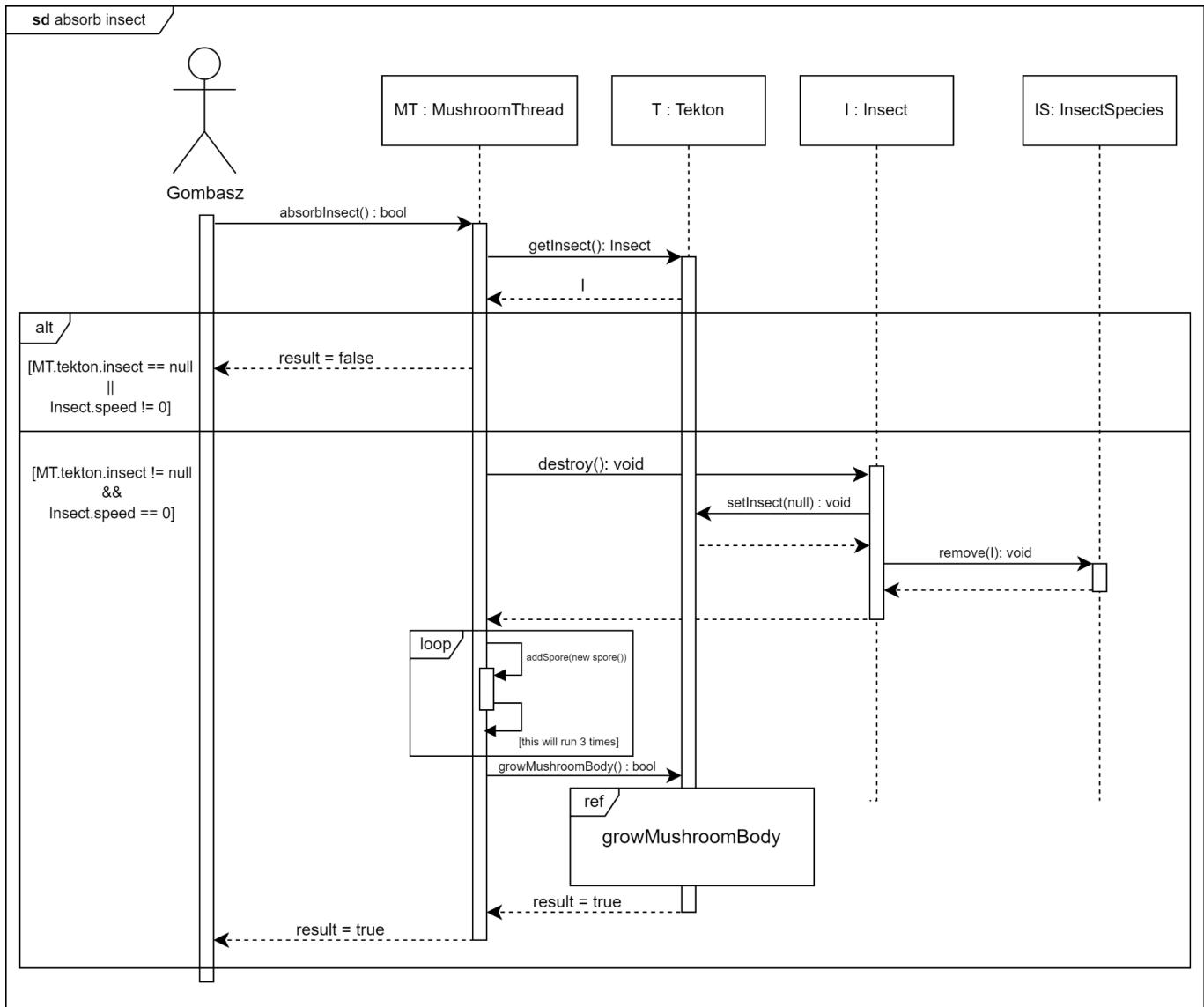
MushroomThread.remove(MushroomThread) : void

Eltávolítja az adott fonalt a listából.

Tekton.tektonEffect() : void

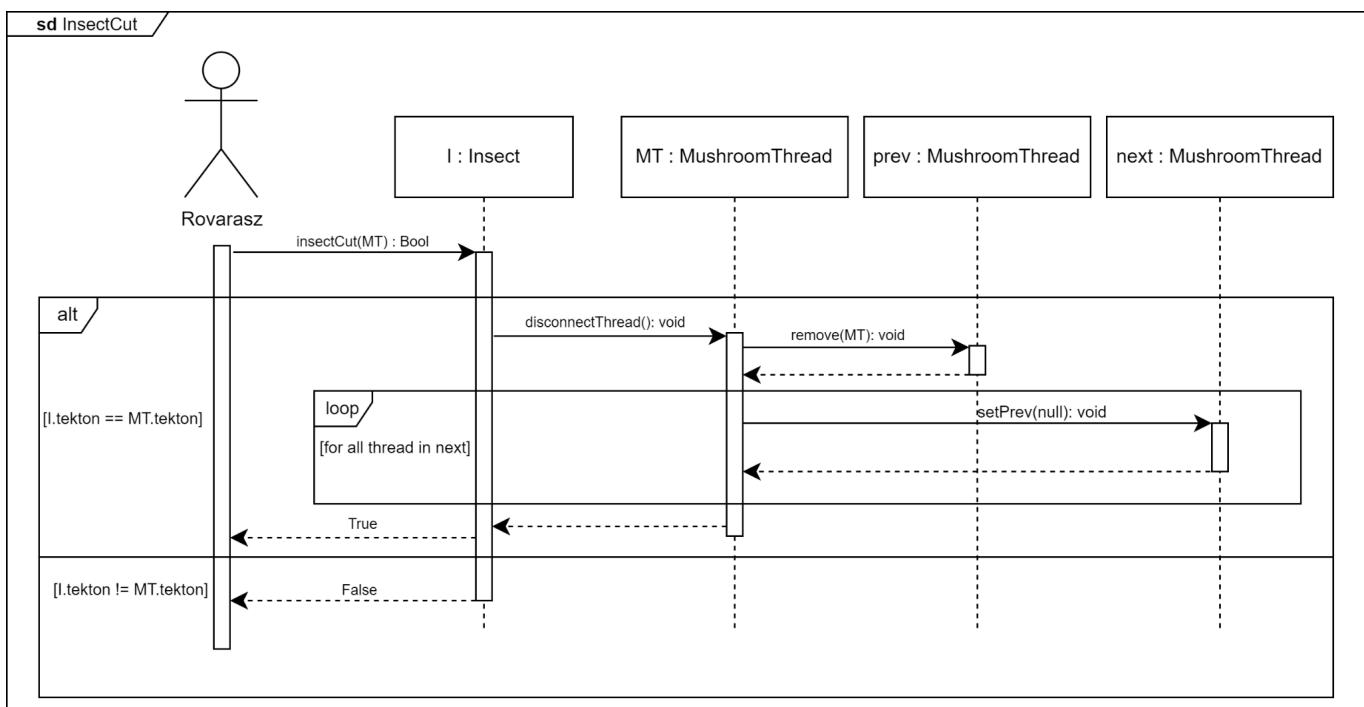
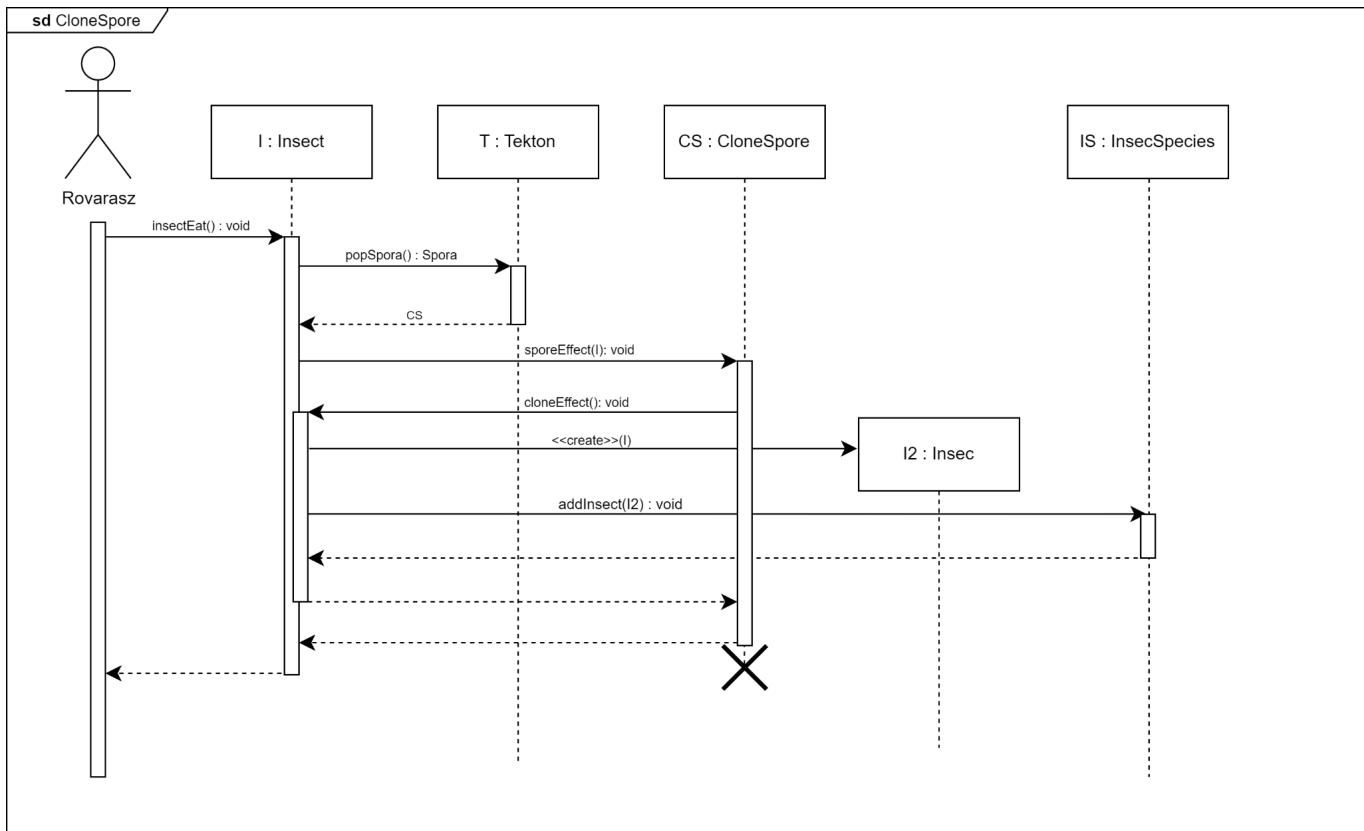
Ez a függvény törölve lett, mivel be lett vezetve az absorb függvény.

7.0.3 Szekvencia-diagramok



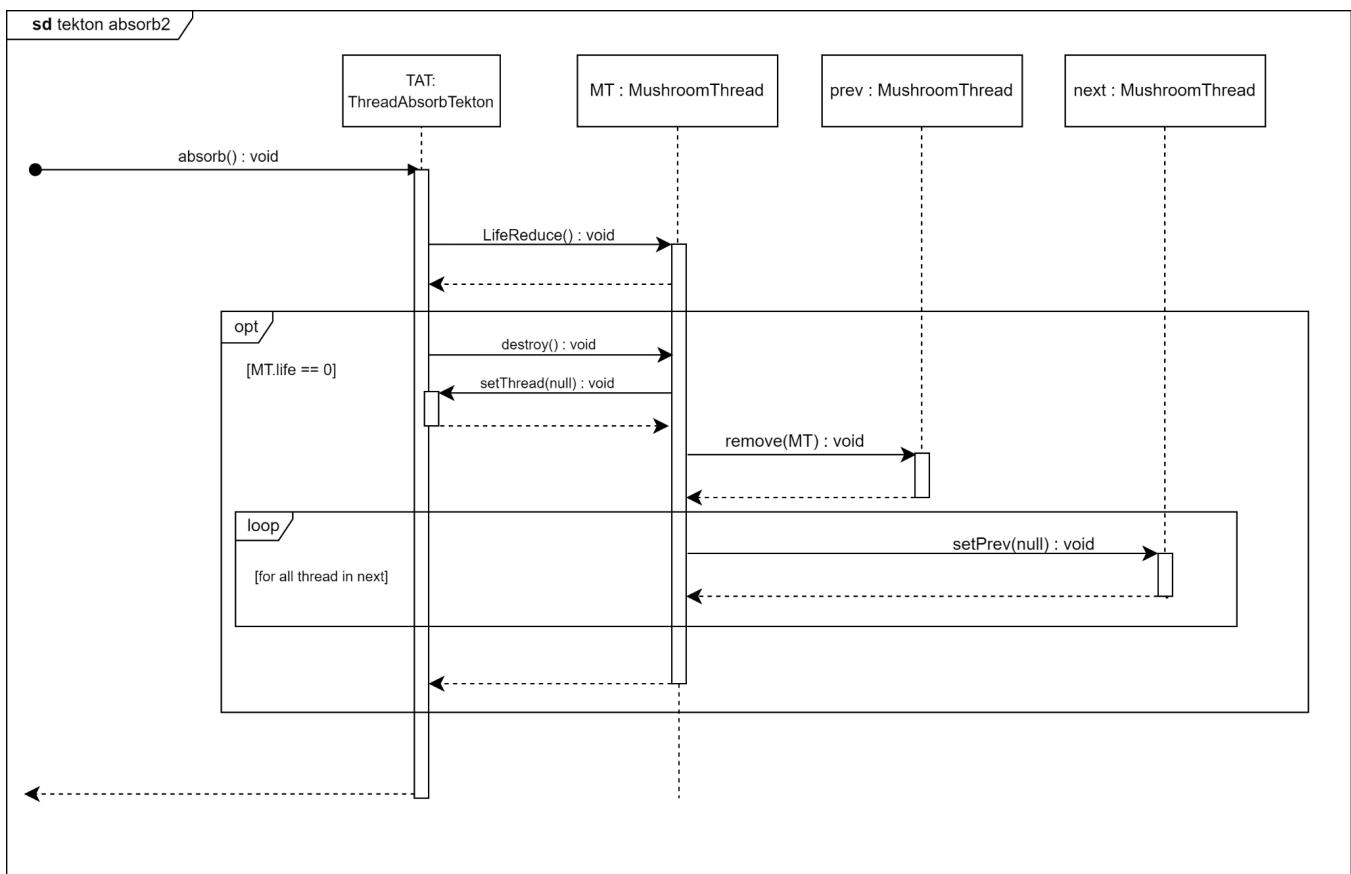
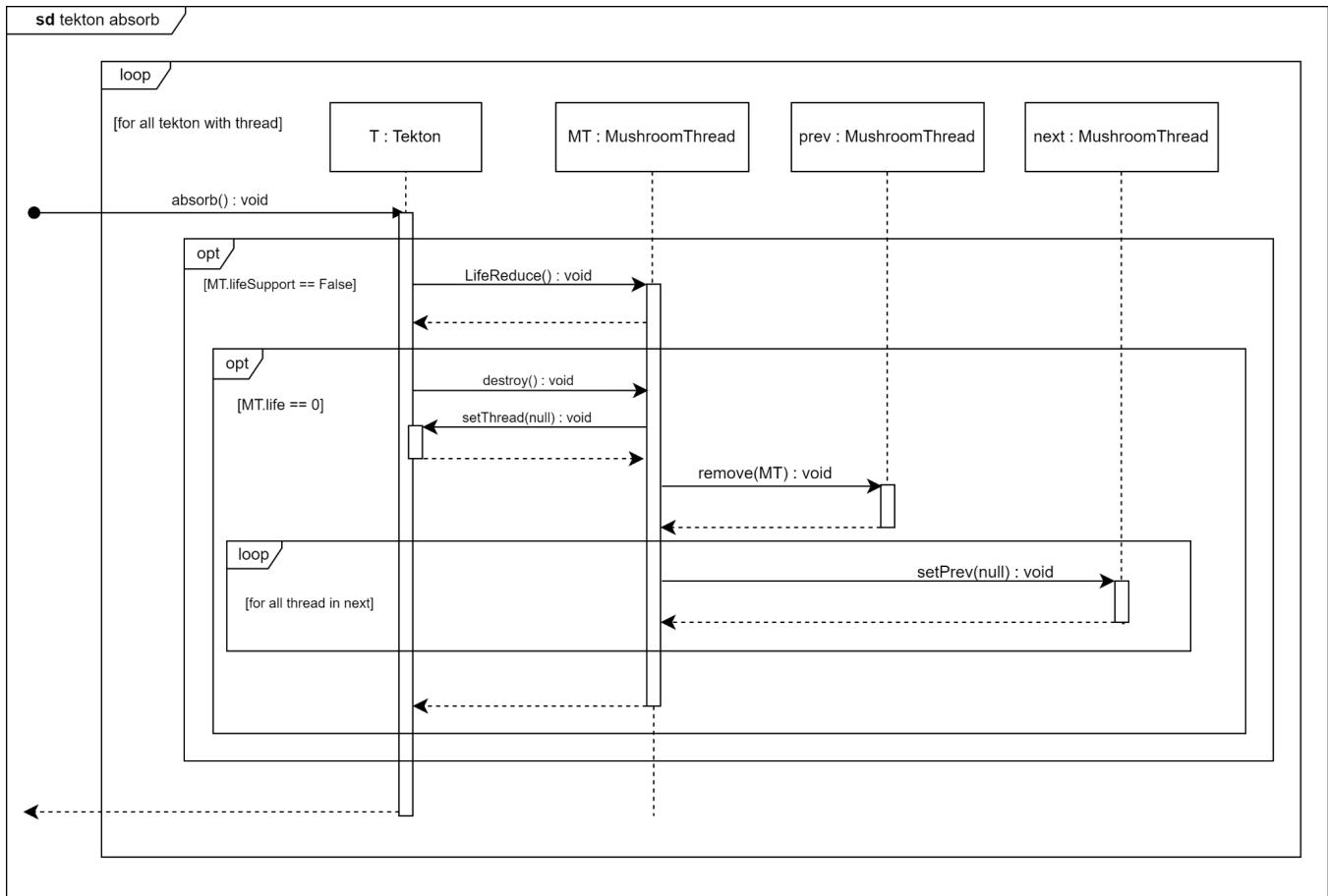
7. Prototípus koncepciója

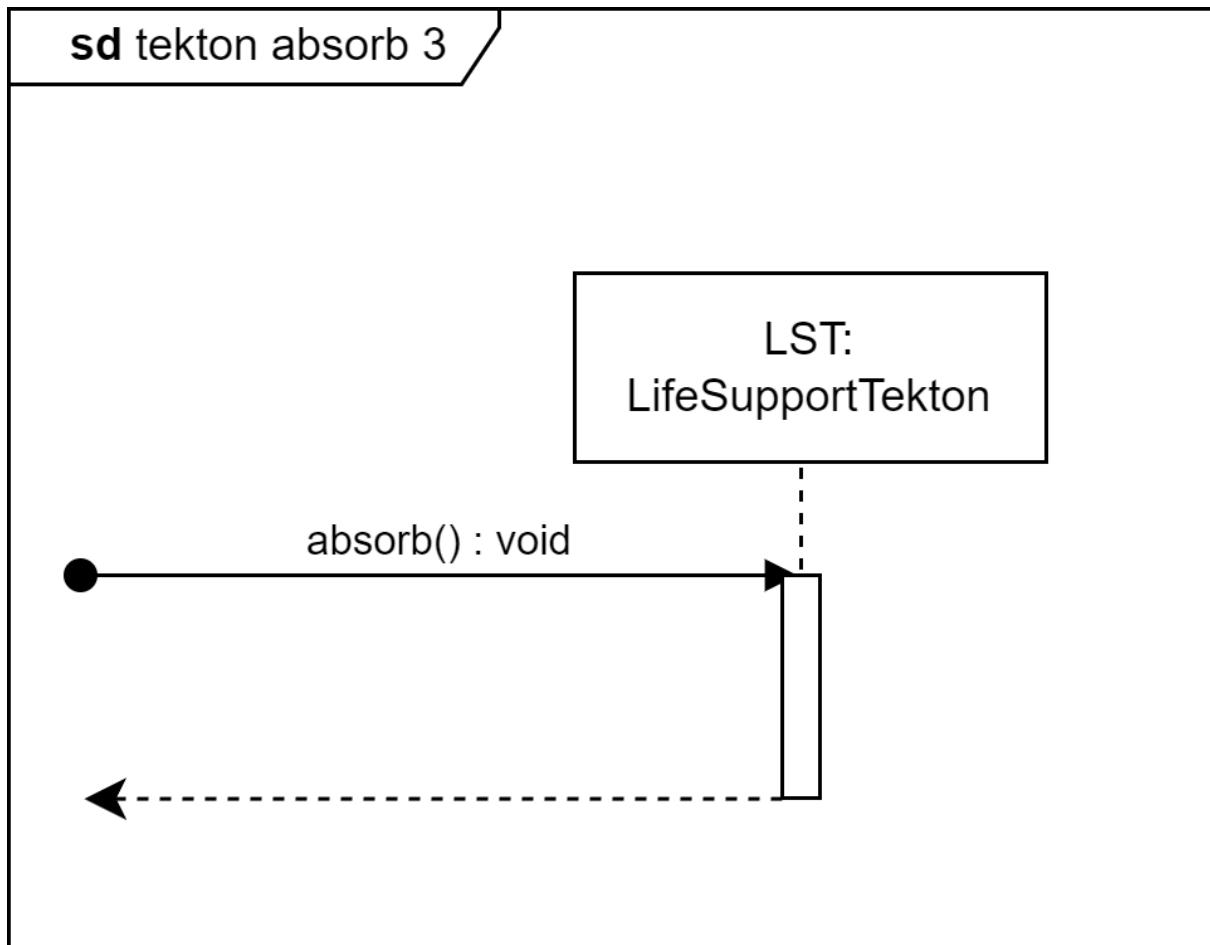
being_chilling



7. Prototípus koncepciója

being_chilling





Minden körben meg fogjuk hívni az összes fonalas tektonon az absorb() függvényt. A viselkedés a tekton típusától függ:

nem LifeSupport és nem ThreadAbsorb tekton esetén: **sd tekton absorb 1**

ThreadAbsorb tekton esetén: **sd tekton absorb 2**

LifeSupport tekton esetén: **sd tekton absorb 3**

7.1 Prototípus interface-definíciója

7.1.1 Az interfész általános leírása

A tesztelés során a bemenetet egy fájlból olvassuk be, amely a megadott parancsokat tartalmazza a megfelelő formátumban, egészen a fájl végéig, soronként max egy parancs. A teszteléshez egyetlen osztályt használunk, ebben valósítjuk meg az összes, JUnit-alapú tesztet kezelő függvényt. A bemenetet ilyen formátumban kell megadni:

COMMAND;ELVÁRT

Ahol a “COMMAND” a 7.1.2-es tesztelhető parancsok egyike, az “ELVÁRT” pedig az elvárt kimenet (a kimeneti formátumban). A bemenet feldolgozása után a kimenet a következő formátumban jelenik meg:

TESZT: <Teszt Száma>;

ELVÁRT: <Elvárt kimenet>;

TÉNYLEGES: <Tényleges kimenet>;

MEGEGYEZIK: <Igen/Nem>

Ez a forma lehetővé teszi az automatikus kiértékelést és a tesztelés eredményének egyértelmű visszajelzését.

A prototípus viszont konzolosan működik, vagyis a konzolon elindítva lehet parancsokat beírni a 7.1.2-es bekezdés szerint. Ha nem létezik az a parancs, akkor kiírja az elérhető parancsokat, ha létezik, de rossz argumentumokat adott meg, akkor a parancs szintaxisát írja ki.

7.1.2 Bemeneti nyelv

<Text inside angle brackets>	=	Kötelező elemek
[Text inside square brackets]	=	Opcionális elemek

/addplayer <player ID> <role>

Leírás: Egy játékos hozzáadása a játékhoz.

tekton type:

R = rovarász

G = gombász

Opciók:

1. Sikeresen hozzáad egy játékost.
2. Az adott érték érvénytelen

/addtekton <tekton ID> [tekton type]

Leírás: Egy új tekton hozzáadása a játékhoz szomszédok nélkül.

tekton type:

N = normal (single thread)

LS = life support

TA= thread absorb

MT = multi thread

MS = mushroomless

default: N

Opciók:

3. Sikeresen hozzáad egy tektont a megfelelő típussal.
4. Az adott érték érvénytelen

/addinsect <playerID> <insect ID> <tekton ID> [effect]

Leírás: Egy új rovar lesz hozzáadva az adott tektonon.

effect:

S = slow
H = haste
M = mute
P = paralyzed
C = clone
N = normal
default: N

Opciók:

1. Sikeresen hozzáadja a rovart a tektonhoz.
2. A rovar nem tudja hozzáadni a tektonhoz, mivel van rajta egy másik rovar. Az
3. Az adott érték érvénytelen.

**/addmush <playerID> <mushroom ID> <thread ID> <tekton ID> [mushroom age]
[mushroom spore]**

Leírás: Egy új gombatest felvétele az adott tektonra, és mellé hozzáadja a fonalat ugyanahhoz a tektonhoz

mushroom age:

1 - 100 nem negatív egész szám
default: 3

mushroom spore:

1 - 100 nem negatív egész szám
default: 3

Opciók:

1. Sikeresen létrehoz egy új gombatestet.
2. Nem sikerül, mivel a tektonon van már gombatest.
3. Az adott érték érvénytelen.

/addspore <tekton ID> <spore ID> [spore type] [nutrient number]

Leírás: Adott típusú spóra hozzáadása az adott tektonhoz.

spore type:

S = slow
H = haste
M = mute
P = paralyzed
C = clone
N = normal
default: N

nutrient number:

0-10 nem negatív egész szám
default: 5

Opciók:

1. Sikeresen hozzáadja a spórát a tektonhoz

2. Az adott érték érvénytelen.

/addthread <tekton ID> <thread ID>

Leírás: Egy új adott típusú gombafonalat ad az adott tektonra

Opciók:

1. Sikeresen létrehozza a gombafonalat
2. Nem hozza létre a gombafonalat, mivel már van gombafonal az adott tektonon (ha nem többfonalas a tekton).
3. Az adott érték érvénytelen.

/setneighbour <tekton1 ID> <tekton2 ID>

Leírás: egymásnak a szomszédjához hozzáad egymást.

Opciók:

1. Sikeresen hozzáadja a szomszédöt a tektonhoz és a "szomszéd tekton"hoz a tektont szomszédként
2. Az adott érték érvénytelen.

/connectthread <thread1 ID> <thread2 ID>

Leírás: összecsatolja a 2 kapott fonalat, a 2es fonalat hozzáadja az 1es fonal következő listájához és fordítva is (prev-hez)

Opciók:

1. Sikeres
2. Nem sikerült, mivel a fonal2-nek a prev nem null
3. Nem sikerült, mivel nem tudja elérni (nem szomszédos a tekton)
4. Az adott érték érvénytelen.

/showID

Leírás: Leírja az összes ID-t.

Opciók: Sikeres

/help

Leírás: Mutatja az összes parancsot.

Opciók: Sikeres

/break <source tekton ID> <new tekton ID> [neighbours for new tekton ...]

Leírás: Az adott tektont eltöri. Kitörli az adott tektont és mellette hozzáad egy új tektont, amelyeknek szomszédjait hozzáadja, módosítja.

Opciók:

1. Sikeresen eltöri a tektont és helyette 2 tekton lesz.
2. Nem tudja eltörni a tektont, mivel van rajta gombatest és/vagy rovar.
3. Az adott érték érvénytelen.

/absorb [tekton ID...]

Leírás: Az összes/adott tektonon a gombafonalakat elkezdi felszívni

Opciók:

1. A felszívás sikeres, mivel a gombafonal nincs semmihez sem csatlakoztatva ami életben tartaná.
2. Az adott érték érvénytelen.

/growthread <source thread ID> <new thread ID> <Tekton ID>

Leírás: Egy új adott típusú gombafonalat ad az adott tektonra ami csatlakozik a source threaddel

Opciók:

1. Sikeres.
2. Sikertelen, mivel nem szomszédos a tekton a fonalhoz képest
3. Sikertelen, már van a sima Tektonon (amin csak egyfonal lehet) fonal.
4. Az adott érték érvénytelen.

/growmush <player ID> <mushroom ID> <tekton ID>

Leírás: Gombatest növesztése a kijelölt tektonon.

Opciók:

1. Sikeres gomba növesztés, mivel van elég spóra a tektonon és van gombafonál is.
2. Sikertelen, mivel nincs elég spóra.
3. Sikertelen, mivel nincs a gombafaj gombafonala a tektonon.
4. Az adott érték érvénytelen.

/spreadspore <mushroom ID> <tekton ID> [spore type]

Leírás: Egy adott típusú spórát egy gombatest egy tektonra.

spore type:

S = slow
H = haste
M = mute
P = paralyzed
C = clone
N = normal
default: N

Opciók:

1. A szórás sikeres, minden feltétel teljesült a spóra szóráshoz.
2. A szórás sikertelen, mivel túl messze van a tekton,
3. A szórás sikertelen, elfogyott a spóra a gombatestnél.
4. Az adott érték érvénytelen.

/absorbinsect <thread ID> <new mushroom ID>

Leírás: Az adott gombafonal felszívja a rovart a tektonon, ha sikeres akkor nő egy új gombatest

Opciók:

1. A gombafonal sikeresen felszívja a rovart és gombatest nő az adott tektonon.
2. Sikertelen, mert nem paralyzed a rovar.
3. Sikertelen, mivel nincs rovar a gombafonal tektonján.
4. Az adott érték érvénytelen.

/cut <insect ID> <thread ID>

Leírás: A rovar elvágja a gombafonalat, ami a tektonból megy ki.

Opciók:

1. A fonal sikeresen elvágódik, mivel minden feltétel teljesül.
2. sikertelen mert el van némítva a rovar,
3. sikertelen mert le van bénulva a rovar,
4. sikertelen mert a gombafonal nem azon a tektonon van, amin a rovar.
5. Az adott érték érvénytelen.

/eat <insect ID>

Leírás: A rovar megeszi a tektonon a legtovább lévő spórát.

Opciók:

1. A rovar megeszi a spórát sikeresen, és megkapja az adott spóra effektjét.
2. A rovar nem tudja megenni a spórát, mivel nincs a tektonon spóra,
3. A rovar nem tudja megenni a spórát, mivel bénulás alatt van a rovar.

/move <insect ID> <tekton ID>

Leírás: A kiválasztott rovar a kijelölt tektonra lép.

Opciók:

1. A rovar rálép a tektonra, mivel minden feltétel rendben van.
2. A rovar nem tud rálépní a tektonra, mivel már van azon a tektonon rovar, vagy nem elég gyors a rovar, hogy elérje a tektont (pl 2 tektonnyit lépne), vagy nincs fonal a rovar tektonja és a ;rkiválasztott tekton között
3. A rovar nem tud mozogni, mivel le van bénulva, emiatt nem tud rálépni arra a tektonra.

/load <file>

Leírás: Betölti az előre meghatározott pályát.

Opciók:

1. Sikeres
2. Nem tudja betölteni a fájlt.

/random <on or off>

Leírás: ki vagy bekapcsol a véletlenszerűséget

/showMap

Leírás: ki írja a pályán lévő objektumokat a konzolra

7.1.3 Kimeneti nyelv

Ahova egy object kell (pl: MushroomBody vagy Tekton), oda az object ID-jét kell írni , -vel elválasztva. Emelett, a type a bemeneti nyelvvel megegyezik.

TektonID(type=<type>; mushroom=[mushroomBody]; insect=[Insect]; threads=[MushroomThread, ...]; spores=[Spore, ...]; neighbours=[Tekton, ...])

MushroomID(Age=<bodyAge: int>; Spore=<sporeNumber: int>; tekton=<tektonHome: Tekton>)

ThreadID(life=<life: int>; lifeSupport=<lifeSupport: bool>; tekton=<tekton: Tekton>; prev=[MushroomThread]; next=[MushroomThread, ...])

SporeID(Nutrient:<sporeNutrient: int>; type:<type>)

InsectID(speed=<insectSpeed: int>; cancut=<cutThread: bool>; caneat=<eatSpore: bool>; tekton=<currentPlace: Tekton>; nutrient=<int>)

PlayerID(Type=<R or G>, Own=<objects>)

7.2 Összes részletes use-case

Use-case neve	Rovar mozgás
Rövid leírás	Rovar 2 tektonnyi távolság között mozog
Aktorok	Felhasználó
Forgatókönyv	A felhasználó kiválaszt egy tektont, ahová lépni szeretne majd, a rovar a tektonra lép

Use-case neve	Rovar mozgás sikertelen
Rövid leírás	Rovar 2 tektonnyi távolság között mozogás sikertelen
Aktorok	Felhasználó
Forgatókönyv	A felhasználó kiválaszt egy tektont, ahová lépni szeretne majd, a rovar nem tud oda lép tektonra, mert nincs annyi tekton (Map széle)

Use-case neve	Spóra evés
Rövid leírás	Rovar megeszik egy spórát
Aktorok	Felhasználó
Forgatókönyv	A rovar megeszik egy spórát a tektonon, és a spóra törlődik a tektonról

Use-case neve	Spóra evés sikertelen
Rövid leírás	Rovar nem tud megenni spórát
Aktorok	Felhasználó
Forgatókönyv	A rovar megeszik egy spórát a tektonon, de sikertelen, mert nincs spóra a tektonon

Use-case neve	Rovar fonalat vágás
Rövid leírás	Rovar elvágja a kiválasztott fonalat
Aktorok	Felhasználó
Forgatókönyv	A rovar elvág egy, a felhasználó által kiválasztott fonalat.

Use-case neve	Rovar fonalat vágás sikertelen
Rövid leírás	Rovar nem tudja elvágni a kiválasztott fonalat
Aktorok	Felhasználó
Forgatókönyv	A rovar nem tud elvágni egy, a felhasználó által kiválasztott fonalat, mert rovar némítő hatás alatt van.

Use-case neve	Rovar plusz pont
Rövid leírás	Felhasználó által megadott pontot hozzáadja a rovarnak
Aktorok	Felhasználó
Forgatókönyv	A pontot hozzáadja a rovar pontszámához

Use-case neve	Rovar plusz pont sikertelen
Rövid leírás	A felhasználó által megadott pontok hozzáadása sikertelen
Aktorok	Felhasználó
Forgatókönyv	A pontot nem adja a rovar pontszámához, mert nem valid a szám (pl. negatív szám)

Use-case neve	Új rovar
Rövid leírás	Létrehoz egy új rovart
Aktorok	Felhaszló
Forgatókönyv	A felhasználó által kiválasztott tektonhoz létrehoz egy új rovart

Use-case neve	Új rovar sikertelen
Rövid leírás	Sikertelen új rovar
Aktorok	Felhasználó
Forgatókönyv	A felhasználó által kiválasztott tektonhoz nem létezik ezért nem tud új rovart létrehozni

Use-case neve	Spóra szórás
Rövid leírás	Gombatest spórát szór
Aktorok	Felhasználó
Forgatókönyv	A felhasználó kiválaszt egy gombatestet, és egy tektont ahová spórát fog szórni, majd kiszórja a spórát

Use-case neve	Spóra szórás sikertelen
Rövid leírás	Gombatest nem tudja kiszórni a spóráját
Aktorok	Felhasználó
Forgatókönyv	A felhasználó kiválaszt egy gombatestet, és egy tektont ahová spórázni fog, de nem sikerül mert nincs elég spóra a gombatestben

Use-case neve	Fonal Növesztés
Rövid leírás	Gombatest növeszt egy fonalat
Aktorok	Felhasználó
Forgatókönyv	A felhasználó kiválaszt egy gombatestet, egy ahhoz tartozó fonalat, és egy tektont ahová növeszteni szeretne

Use-case neve	Gombatest Növés
Rövid leírás	Egy tektonon egy gombatest növesztése
Aktorok	Felhasználó
Forgatókönyv	A felhasználó által kiválasztott tektonra növeszt egy gombatestet

Use-case neve	Gombatest növés sikertelen
Rövid leírás	Egy tektonon egy gombatest növesztése sikertelen
Aktorok	Felhasználó
Forgatókönyv	A felhasználó által kiválasztott tektonra növeszteni kíván egy gombatestet, de nem tud mert nincs elég spóra

Use-case neve	Gombatest gyűjtemény hozzáadás
Rövid leírás	Gombatestet hozzáadja a gombatestgyűjteményhez
Aktorok	Felhasználó
Forgatókönyv	A felhasználó kiválaszt egy gombatestet, amit hozzáad a gombatest gyűjteményhez

Use-case neve	Tekton kettétörés
Rövid leírás	1 tekton kettétörés után 2 lesz
Aktorok	Felhasználó
Forgatókönyv	A felhasználó kiválaszt egy tektont, ami kettétör

Use-case neve	Tekton kettétör sikertelen
Rövid leírás	1 tekton kettétörés sikertelen
Aktorok	Felhasználó
Forgatókönyv	A felhasználó kiválaszt egy tektont, ami ketté törne, de nem sikerül, mert tektonon van rovar

Use-case neve	Felszívódás
Rövid leírás	A Tektontól levon egy életet (ha nem LifeSupportTekton)
Aktorok	Felhasználó
Forgatókönyv	A Tekton felszívja a gombafonalat, ha a gombafonal egyik gombatesthez sincs összekötve.

Use-case neve	Tekton Hozzáadás
Rövid leírás	A felhasználó létrehoz egy új tektont
Aktorok	Felhasználó
Forgatókönyv	A felhasználó hozzáad egy tektont a játékhoz, akkor megadja, hogy milyen típusú a tekton.

Use-case neve	Szomszédok beállítás
Rövid leírás	A tektonnak a szomszéd tekonok beállítása
Aktorok	Felhasználó
Forgatókönyv	A felhasználó beállítja a tektonnak szomszédos tekonokat egy listában, közben megadja a szomszédos tekonoknak a típusát.

Use-case neve	Rovar Felszívódás
Rövid leírás	A gombaFonal a rovarat megeszi
Aktorok	Felhasználó
Forgatókönyv	A tekton lévő gombaFonal megeszi azt a rovar, ami bénító hatással alatt van.

Use-case neve	Gombafonal hozzáadás
Rövid leírás	A felhasználó hozzáad egy gombafonalat a tektonra
Aktorok	Felhasználó
Forgatókönyv	A felhasználó hozzáad egy gombafonalat a tektonhoz, akkor megadja, hogy milyen típusú a gombaFonal.

Use-case neve	Spóra hozzáadás
Rövid leírás	A felhasználó hozzáad egy spórát egy tekontra
Aktorok	Felhasználó
Forgatókönyv	A felhasználó hozzáad egy spórát a tektonhoz, akkor megadja, hogy milyen típusú a spórát.

Use-case neve	Betöltés
Rövid leírás	Játék visszatöltése mentett állapotból
Aktorok	Felhasználó
Forgatókönyv	A felhasználó, visszatölt egy mentett játékot, ekkor megadja a mentés elérési útját.

Use-case neve	Betöltés sikertelen
Rövid leírás	A játék nem tud visszatölteni
Aktorok	Felhasználó
Forgatókönyv	A felhasználó, nem tud visszatölteni egy mentett játéket, mert az elérési útja nem megfelelő vagy a mentett játék nem megfelelő, illetve üresen mentett játéket sem tud visszatölteni

Use-case neve	Új játék
Rövid leírás	Játék újrakezdése
Aktorok	Felhasználó
Forgatókönyv	A felhasználó, a new game kimenetet választva, újraindítja a játékot

Use-case neve	Térkép megjelenítés
Rövid leírás	A tektononok és rovarok, illetve a gomba és gombafonalaknak a helyzetét megjeleníti
Aktorok	Felhasználó
Forgatókönyv	Az egész térképet, illetve a térképen lévő komponenst megjeleníti, a részleges tulajdonságok kivételével..

Use-case neve	Statisztikák megjelenése
Rövid leírás	Kiírja a térképen lévő elemeket
Aktorok	Felhasználó
Forgatókönyv	Felhasználó lekéri a saját objektumainak az adatait. (gombatestek, rovarok)

7.3 Tesztelési terv

Teszt-eset neve	1. Rovar move success
Rövid leírás	A teszt ellenőrzi, hogy a rovar sikeresen át tud-e lépni egy tektonra, ha minden feltétel teljesül (van kapcsolatban álló gombafonal, van elég sebessége, és nincs bénulás alatt).
Teszt célja	Rovar mozgásának a sikerességét ellenőrzi. (Insect.insectMove())

Teszt-eset neve	2. Rovar move unsuc (no fonal)
Rövid leírás	A teszt azt ellenőrzi, hogy a rovar tud-e mozogni, ha nincs kapcsolatban álló gombafonal.
Teszt célja	Rovar mozgásának a sikertelenségét ellenőrzi. (Insect.insectMove())

Teszt-eset neve	3. Rovar move unsec (no enough speed)
Rövid leírás	A teszt azt ellenőrzi, hogy a rovar tud-e mozogni, ha nincs elegendő sebessége a kiválasztott tekton eléréséhez.
Teszt célja	Rovar mozgásának a sikertelenségét ellenőrzi. (Insect.insectMove())

Teszt-eset neve	4. Rovar move unsec (para)
Rövid leírás	A teszt ellenőrzi, hogy a rovar tud-e mozogni, ha bénulás alatt van.
Teszt célja	Rovar mozgásának a sikertelenségét ellenőrzi. (Insect.insectMove())

Teszt-eset neve	5. Rovar eat spore success
Rövid leírás	A teszt ellenőrzi, hogy a rovar sikeresen meg tudja-e enni a spórát a tektonon, és megkapja-e a tápanyagot.
Teszt célja	Spóra evésnek a sikertelenségét ellenőrzi. (Insect.insectEat())

Teszt-eset neve	6. Rovar eat spore unsec (no spore)
Rövid leírás	A teszt ellenőrzi, hogy a rovar nem tudja-e enni a spórát a tektonon, ha nincs rajta spóra.
Teszt célja	Spóra evésnek a sikertelenségét ellenőrzi. (Insect.insectEat())

Teszt-eset neve	7. Rovar eat spore unsec (para)
Rövid leírás	A teszt ellenőrzi, hogy a rovar nem tudja megenni a spórát, ha bénulás alatt van.
Teszt célja	Spóra evésnek a sikertelenségét ellenőrzi. (Insect.insectEat())

Teszt-eset neve	8. Rovar cut success
Rövid leírás	A teszt ellenőrzi, hogy a rovar sikeresen el tudja vágni a gombafonalat, ha minden feltétel teljesül.
Teszt célja	Fonal vágásnak a sikertelenségét ellenőrzi. (Insect.insectCut())

Teszt-eset neve	9. Rovar cut unsec(mute)
Rövid leírás	A teszt ellenőrzi, hogy a rovar nem tudja elvágni a gombafonalat, ha el van némitva.
Teszt célja	Fonal vágásnak a sikertelenségét ellenőrzi. (Insect.insectCut())

Teszt-eset neve	10. Rovar cut unsec(para)
Rövid leírás	A teszt ellenőrzi, hogy a rovar nem tudja elvágni a gombafonalat, ha bénulás alatt van.
Teszt célja	Fonal vágásnak sikertelenségét ellenőrzi. (Insect.insectCut())

Teszt-eset neve	11. Rovar cut unsec(no thread)
Rövid leírás	A teszt ellenőrzi, hogy a rovar nem tudja elvágni a gombafonalat, ha nincs gombafonal a tektonon.
Teszt célja	Fonal vágásnak a sikertelenségét ellenőrzi. (Insect.insectCut())

Teszt-eset neve	12. Rovar cut unsec(already cut)
Rövid leírás	A teszt ellenőrzi, hogy a rovar nem tudja elvágni a gombafonalat, ha már el lett vágva
Teszt célja	Fonal vágásnak a sikertelenségét ellenőrzi. (Insect.insectCut())

Teszt-eset neve	13. Eat haste spore
Rövid leírás	A teszt ellenőrzi, hogy a rovar megkapja a gyorsító hatást, ha a gyorsító spórát elfogyasztja.
Teszt célja	Gyorsító spóra effektus megadás sikereségét ellenőrzi. (Insect.hasteEffect())

Teszt-eset neve	14. Eat slow spore
Rövid leírás	A teszt ellenőrzi, hogy a rovar megkapja a lassító hatást ,ha lassító spórát elfogyasztja.
Teszt célja	Lassító spóra effektus megadás sikereségét ellenőrzi. (Insect.slowEffect())

Teszt-eset neve	15. Eat mute spore
Rövid leírás	A teszt ellenőrzi, hogy a rovar megkapja a némitő hatást, ha a némitő spórát elfogyasztja.
Teszt célja	Némitő spóra effektus megadás sikereségét ellenőrzi. (Insect.muteEffect())

Teszt-eset neve	16. Eat para spore
Rövid leírás	A teszt ellenőrzi, hogy a rovar megkapja a bénítő hatást, ha a bénítő spórát elfogyasztja.
Teszt célja	Bénító spóra effektus megadás sikereségét ellenőrzi. (Insect(paraEffect))

Teszt-eset neve	17. Eat clone spore
Rövid leírás	A teszt ellenőrzi, hogy a rovar sikeresen klónozza-e önmagát, miután megeszi a klónozó spórát, és létrejön-e egy új, független rovar az eredeti rovartól.
Teszt célja	Klón spóra effektus megadás sikereségét ellenőrzi. (Insect.cloneEffect())

Teszt-eset neve	18. Spread spore success
Rövid leírás	A teszt azt ellenőrzi, hogy a gombatest sikeresen szórja-e ki a spórákat a kiválasztott tektonra, amennyiben minden feltétel teljesül (van elég spóra, a tekton elérhető távolságban van).
Teszt célja	Spóra szórásnak a sikerességét ellenőrzi. (MushroomBody.spreadSpore())

Teszt-eset neve	19. Spread spore unsec(no spore)
Rövid leírás	A teszt azt ellenőrzi, hogy a gombatest nem tudja spórát szórni, ha nincs spóra a gombatestben.
Teszt célja	Spóra szórásnak a sikertelenségét ellenőrzi. (MushroomBody.spreadSpore())

Teszt-eset neve	20. Spread spore unsec(too far)
Rövid leírás	A teszt azt ellenőrzi, hogy a gombatest nem tudja spórát szórni, ha a kiválasztott tekton túl messze van, azaz nem esik a gombatest hatósugarán belül.
Teszt célja	Spóra szórásnak a sikertelenségét ellenőrzi. (MushroomBody.spreadSpore())

Teszt-eset neve	21. Grow mushroom success
Rövid leírás	A teszt azt ellenőrzi, hogy a gombatest sikeresen kinő-e a kiválasztott tektonon, ha minden feltétel teljesül (elegendő spóra, gombafonal jelenléte).
Teszt célja	Gomba növesztésnek a sikerességét ellenőrzi. (MushroomBody.growMushroomBody())

Teszt-eset neve	22. Grow mushroom unsec(no enough spore)
Rövid leírás	A teszt azt ellenőrzi, hogy a gombatest nem tudja kinöni a kiválasztott tektonon, ha nincs elegendő spóra mennyiség a gombatestben.
Teszt célja	Gomba növesztésnek a sikertelenségét ellenőrzi. (MushroomBody.growMushroomBody())

Teszt-eset neve	23. Grow mushroom unsec(no fonal)
Rövid leírás	A teszt azt ellenőrzi, hogy a gombatest nem tudja kinöni a kiválasztott tektonon, ha nincs gombafonal a tektonon.
Teszt célja	Gomba növesztésnek a sikertelenségét ellenőrzi. (MushroomBody.growMushroomBody())

Teszt-eset neve	24. Grow mushroom unsec(mushroomless)
Rövid leírás	A teszt azt ellenőrzi, hogy a gombatest nem tudja kinöni a kiválasztott tektonon, ha az a tekton "gomba nélküli" típusú, azaz nem alkalmas gomba növesztésére.
Teszt célja	Gomba növesztésnek a sikertelenségét ellenőrzi. (MushroomBody.growMushroomBody())

Teszt-eset neve	25. Grow fonal success
Rövid leírás	A teszt azt ellenőrzi, hogy a gombatest sikeresen növeszt-e új gombafonalat a kiválasztott tektonra, amennyiben minden feltétel teljesül (van elegendő erőforrás, a tekton elérhető, stb.).
Teszt célja	Fonal növesztésnek a sikerességét ellenőrzi. (MushroomBody.growThread())

Teszt-eset neve	26. Grow fonal unsec(occupied)
Rövid leírás	A teszt azt ellenőrzi, hogy a gombatest nem tudja új gombafonalat növeszteni a kiválasztott tektonra, ha az a tekton már foglalt, azaz már van rajta gombafonal.
Teszt célja	Fonal növesztésnek a sikertelenségét ellenőrzi. (MushroomBody.growThread())

Teszt-eset neve	27. Grow fonal random
Rövid leírás	A teszt azt ellenőrzi, hogy a gombafonal növekedése felgyorsul és tovább nő, ha spóra van a tektonon, ahol a fonal növekszik.
Teszt célja	Fonal növesztés gyorsítását ellenőrzi. (MushroomThread.speedUpGrowing())

Teszt-eset neve	28. Tekton Absorb
Rövid leírás	A teszt azt ellenőrzi, hogy a `Tekton.absorb()` metódus megfelelően csökkenti-e a gombafonalak életerejét a tektonon, ha azok nem kapcsolódnak életben tartó struktúrához (pl. gombatesthez vagy LifeSupportTektonhoz).
Teszt célja	Tekton absorb és élet csökkenése ellenőrzi. (Tekton.absorb(), MushroomThread.lifeReduce())

Teszt-eset neve	29. Tekton Absorb*3
Rövid leírás	A teszt azt ellenőrzi, hogy a `Tekton.absorb()` metódus háromszor egymás után meghívva, megfelelően csökkenti-e a gombafonalak életerejét a tektonon `MushroomThread.lifeReduce()` meghívva, ha sima gombafonal akkor háromszor `MushroomThread.lifeReduce()` meghívás után eltűnik.
Teszt célja	Tekton absorb és élet csökkenése ellenőrzi. (Tekton.absorb(), MushroomThread.lifeReduce())

Teszt-eset neve	30.Tekton Absorb*5
Rövid leírás	A teszt azt ellenőrzi, hogy a `Tekton.absorb()` metódus ötször egymás után meghívva, megfelelően csökkenti-e a gombafonalak életrejét a tektonon ‘MushroomThread.lifeReduce()’, ha lifeThread gombafonal akkor ötször ‘MushroomThread.lifeReduce()’ meghívás után eltűnik.
Teszt célja	Tekton absorb és élet csökkenése ellenőrzi. (Tekton.absorb(), MushroomThread.lifeReduce())

Teszt-eset neve	31. Tekton break success
Rövid leírás	A teszt ellenőrzi, hogy a tekton sikeresen kettétörök-e, és helyette két új tekton jön-e létre.
Teszt célja	Tekton ketté törlésnek a sikereségét ellenőrzi. (Tekton.tektonBreak())

Teszt-eset neve	32. Tekton break unsec(gomba)
Rövid leírás	A teszt ellenőrzi, hogy a tekton nem törhető ketté, ha gombatest van rajta.
Teszt célja	Tekton ketté törlésnek a sikertelenségét ellenőrzi. (Tekton.tektonBreak())

Teszt-eset neve	33. Tekton break unsec(rovar)
Rövid leírás	A teszt ellenőrzi, hogy a tekton nem törhető ketté, ha rovar van rajta.
Teszt célja	Tekton ketté törlésnek a sikertelenségét ellenőrzi. (Tekton.tektonBreak())

Teszt-eset neve	34. Absorb rovar success
Rövid leírás	A teszt ellenőrzi, hogy a gombafonal sikeresen megeszi-e a bénult rovart a tektonon
Teszt célja	A fonal rovar evésének sikereségét ellenőrzi. (MushroomThread.absorbInsect())

Teszt-eset neve	35. Absorb rovar unsec(not para)
Rövid leírás	A teszt ellenőrzi, hogy a gombafonal nem tudja megenni a rovart, ha az nem bénult állapotban van.
Teszt célja	A fonal rovar evésének sikertelenségét ellenőrzi. (MushroomThread.absorbInsect())

7.4 Tesztelést támogató segéd- és fordítóprogramok specifikálása

A rendszer tesztelésének támogatásához az IntelliJ IDEA fejlesztőkörnyezetet és a JUnit tesztkeretrendszer használjuk. Az IntelliJ IDEA lehetővé teszi a kód struktúrájának átlátható kezelését, a hibakeresés hatékony végrehajtását és az egységesztek kényelmes futtatását, így a fejlesztési ciklus során a hibák könnyebben azonosíthatók és javíthatók. A tesztelés során elsősorban arra törekszünk, hogy a bemeneti parancsok hatása egyértelműen ellenőrizhető legyen, az objektumok állapotának változása pontosan visszakövethető legyen, valamint a hibakezelés is megfelelő visszajelzést adjon.

A tesztelést JUnit-al valósítjuk meg. A JUnit teszt keretrendszer tartalmazza az Assertions osztályt, amely segítségével lehet a teszteket könnyedén megírni. Tartalmazza az assertEquals(expected, actual), assertFalse(actual), assertTrue(actual) függvényeket. Ezek összehasonlíják az elvárt értéket a valós értékkel, és ha nem ugyanaz, akkor Assertions.fail() függvény hívódik meg, kiíratva az értékekkel.

7.5 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2025.03.25. 8:00	3 óra	Yin	Tevékenység: SD új metódus készítése
2025.03.26. 15:00	2 óra	Bene	Tevékenység: CD elkészítése, metódusok ellenőrzése
2025.03.27 16:00	2 óra	Bene Kattan Ji Yuang Ji Jin Yan Yin	Értekezlet: Megbeszélük, ki mit csinál és hogyan fogjuk.
2025.03.27 15:00	2.5 óra	Yin	Tevékenység: Tesztesetek neve és célja megadása
2025.03.28 10:00	2 óra	Ji Jin Yan	Tevékenység: 7.2 megírása
2025.03.28 10:00	2 óra	Ji Yuang	Tevékenység: 7.2 megírása
2025.03.28 15:00	3 óra	Bene	Tevékenység: 7.1.2 megírása
2025.03.28 17:00	3 óra	Kattan	Tevékenység: 7.1.3 és 7.4 megírása
2025.03.28 20:00	1 óra	Bene	Tevékenység: 7.1.1 kiegészítése/megírása
2025.03.29 10:00	1 óra	Ji Yuang	Tevékenység: 7.3 kiegészítése/megírása
2025.03.29 10:00	1 óra	Ji Jin Yan	Tevékenység: 7.3 kiegészítése/megírása
2025.03.30 14:00	1 óra	Kattan	Tevékenység: 7.2 átnézése, kijavítása

8. Részletes tervezek

82 – being_chilling

Konzulens:
Tóth Bence

Csapattagok

Yin Jun	TN1KZI	1171276297yin@gmail.com
Bene Máté	K4C4W1	matthew.be04@gmail.com
Kattan Ibrahim Gábor	ZU22JJ	kattanbarhum2@gmail.com
Ji Jin Yan	ZGCKAL	jijinyan0201@gmail.com
Ji Yuang	UI7V2S	yuangji688@gmail.com

2025.04.13.

8. Részletes tervez

8.1 Osztályok és metódusok tervez.

Az alábbi osztályok közül ahol van interface, ott van toView és toController függvény, amelyek visszaadják a nekik megfelelő interface objectet. (pl a Tekton-nál visszaadja a toView a TektonView-t)

8.1.1 MushroomBody

- **Felelősség**

A Gombatest egy növényi entitás a játékban, amely spórákat szórhat és fonalakat növeszthet.

- **Interfészek**

MushroomBodyController, MushroomBodyView

- **Attribútumok**

- **-int bodyAge** - ez tárolja az életkorát a gombának.
- **-int sporeNumber** - A gombatest maradék spóraszáma.
- **-Tekton tektonHome** - Azt tárolja, hogy melyik tektonon van a gombatest.

- **Metódusok**

- **+bool spreadSpore(Tekton t)** - Spórát szór egy adott tektonra, megnövelve 1-el a spóraszámát a tektonnak.
- **+bool growThread(MushroomThread mt, Tekton t)** - A paraméterben kapott t tektonra növeszt fonalat, ha nem egy meglévő gombafonal végét növeszti, akkor egy új ágat hoz létre arra a tektonra.
- **+void popSpore()** - 1-el csökkenti a gombatestben hátralevő spórák számát.
- **+int getBodyAge()** - bodyAge értékét adja vissza
- **+void setBodyAge(int bodyAge)** - bodyAge értékét írja a paraméterben megadott értékre.
- **+int getSporeNumber()** - sporeNumber értékét adja vissza.
- **+Tekton getLocation()** - a tektonHome értékét adja vissza

```

method growThread(selThread: MushroomThread, selTekton: Tekton)
    if selTekton in selThread.tekton.neighbors then
        Create new MushroomThread MT2
        if selTekton.addThread(MT2) is true then
            add MT2 to selThread.nextGrown
            MT2.prev = selThread
            if size of selTekton.spores bigger than 0 then
                speedUpGrowing()
            end if
            return true
        end if
    end if
    return false
end

```

8.1.2 MushroomThread

- **Felelősség**

Az osztály a gombatestek által létrehozott gombafonalak funkcióit valósítja meg.

- **Interfészek**

MushroomThreadController, MushroomThreadView

- **Attribútumok**

- **-int life** - A fonalnak a megmaradt életéreje. Ha 0, akkor meghal a gombafonal.
- **-bool lifeSupport** - Ha igaz, akkor a fonal LifeSupportTektonon van.
- **-MushroomThread prevGrown** - A sorban előtte lévő gombafonal.
- **-MushrommThread nextGrown** - Az összes következő fonal listában.
- **-Tekton tekton** - Melyik tektonon van a fonal.

- **Metódusok**

- **+void speedUpGrowing()** - Ha a fonal olyan tektonra nő, ahol van spóra, akkor egy véletlenszerű irányba nő mégegyszer.
- **+void addThread(MushroomThread mt)** - A listához hozzáadja az adott fonalat.
- **+MushroomBody checkOwner()** - Megnézi, hogy melyik gombatesthez tartozik a jelen lévő gombafonal.
- **+void disconnectThread()** - Leválasztja az aktuális fonalat az ót megelőző fonaltól. Ezt úgy éri el, hogy az aktuális fonal prev attribútumát null-ra állítja, és eltávolítja az aktuális fonalat az előző fonal next listájából
- **+void lifeReduce()** - 1-el csökkenti a fonal életét.
- **+ int getLife()** - life attribútumot adja vissza.
- **+void setLife(int life)** - life attribútumot átírja az adott paraméter értékére.
- **+boolean isLifeSupport()** - visszaadja lifeSupport értékét.
- **+ void setLifeSupport(boolean lifeSupport)** - módosítja lifeSupport értékét a megadott paraméter értékére.
- **+MushroomThread getPreGrown()** - visszaadja a preGrown-ot.
- **+List<MushroomThread> getNextGrown()** - visszaadja a nextGrown-ot.
- **+Tekton getLocation()** - visszaadja a tekton-t.
- **+void setLocation(Tekton location)** - tekton-t átírja a paraméterben kapott értékre
- **+bool absorbInsect()** - Megeszi a tektonon található bénult rovarokat. Ilyenkor a rovar elpusztul, a fonal pedig gombatestet növeszthet.
- **+void destroy()** - A fonal destruktora, elvégzi a szükséges tisztítási lépésekét, például megszünteti a fonalra mutató referenciakat más osztályokban(null-ra állítja vagy eltávolítja egy listából)

- **+void remove(MushroomThread MT)** - Eltávolítja az adott fonalt a listából.

```

method speedUpGrowing()
    while true do
        create new MushroomThread MT2
        randomly choose a number between 0 and tekton.neighbors.size(): randIDX
        if tekton.neighbors[randIDX].addThread(MT2) is true then
            add MT2 to nextGrown
            MT2.prevGrown = this
            break
        end if
    end loop
end

method checkOwner():
    Create new MushroomThread MT2
    while prevGrown is not null do
        MT2 = prevGrown
    end loop
    return tekton.mushroomBody
end

method absorbInsect():
    If this.tekton.insectOnTekton is not null then
        this.tekton.insectOnTekton.species.remove(this.tekton.insectOnTekton)
        this.tekton.insectOnTekton = null
    return true
    Endif
    return false
End

method destroy():
    this.prevGrown = null
    for each MushroomThread mt in nextGrown
        mt.prevGrown = null
    end loop
    this.nextGrown = null
    this.tekton.threads.remove(this)
end

```

8.1.3 MushroomSpecies

- **Felelősség**

Az osztály egy gombafaj összes gombatest példányát tárolja el.

- **Interfészek**

-

- **Metódusok**

- **+List<MushroomBody> checkMushroomBody()** - Visszaadja az összes létező példányt.
- **+void addMushroomBody(MushroomBody mb)** - Felvesz egy új gombatestet.
- **+ void setMushroomBodies(List<MushroomBody> mushroomBodies)** - a mushroomBodies értékét átírja a paraméterben megadottra.

8.1.4 Tekton

- **Felelősség**

- Az osztály a játék alapmezőjét valósítja meg, a tektont. Legfőbb felelőssége a játéktérrel kapcsolatos, de fonalak és spórák működését is elősegíti.

- **Interfészek**

TektonController, TektonView

- **Attribútumok**

- **-Insect insectOnTekton:** - A rovar, amely a tektonont tartózkodik. Ha nincs a tektonon rovar, akkor null.
- **-List<Tekton> neighbors:** - A tekton szomszédainak listája.
- **-MushroomBody mushroomBody:** - Ha van gombatest a tektonon, akkor a gombatestet tárolja. Ha nincs, akkor null.
- **-List<MushroomThread> threads:** - A tektonon lévő gombafonalak listája.
- **-List<Spore> spores** - A tektonon található spórákat tárolja egy listában.

- **Metódusok**

- **+boolean addThread(mt: MushroomThread):** - Hozzáad egy gombafonalat (MushroomThread) a tektonhoz. Visszatérési értéke "boolean", ami jelzi, hogy a hozzáadás sikeres volt-e vagy sem.
- **+void tektonBreak():** - Ez a metódus a tekton félbetörését valósítja meg. Olyan tekton nem törhet ketté, amin rovar, vagy gombatest van. A tektonon lévő fonalak elszakadnak, spórák eltűnnek, és majd félbetörés után, az eredeti tekton ugyanúgy marad, emellett még keletkezik egy új tekton pluszban.
- **+Spore popSpore():** - Eltávolítja a legrégebbi spórát a tektonról. Visszatér azzal a spórával.
- **+void clearSpore():** - Eltávolítja az összes spórát a tektonról.
- **+boolean growMushroomBody(ms: MushroomSpecies) :** - Növeszt egy gombatestet a tektonon. Visszatérési értéke jelzi, hogy a növesztés sikeres volt-e vagy sem.
- **+ void addMushroom(MushroomBody mushroomBody)** - Hozzáad egy gombatestet a tektonhoz.
- **+List<Tekton> getNeighborsWithThread():** - Visszaadja a szomszédos tektonokat, amelyekhez gombafonal kapcsolódik.
- **+boolean checkNeighbor(t: Tekton):** - Ellenőrzi, hogy egy adott tekton (t) szomszédos-e az aktuális tektonnal.
- **+void addSpores(s: Spore):** - Hozzáad egy spórát (Spore) a tektonhoz.
- **+void updateNeighbor(List<Tekton>, delete: List<Tekton>):** - Frissíti a szomszédos tektonok listáját.
- **+void absorb():** - Felszívja a gombafonalakat a tektonon.
- **-void useSporeToGrow()** - Felhasznál 3 spórát a növekedéshez.
- **+void addNeighbor(Tekton t):** Hozzáadja az adott tektonot a szomszédok listájához
- **+void deleteNeighbor(Tekton t):** Törli az adott tektonot a szomszédok listájából
- **+MushroomBody getBody()** - visszaadja a mushroomBody értékét.
- **+List<MushroomThread> getThreads()** - visszaadja a threads értékét
- **+Insect getInsect()** - visszaadja az insectOnTekton értékét
- **+List<Spore> getSpores()** - visszaadja a spores értékét.
- **+void addInsect(Insect I)** - insect hozzáadás a tektonra, ami csak egy lehetséges

```

method addThread(mt: MushroomThread)
    if threads is not null then
        return false
    end if
    add mt to threads
    return true
end

method tektonBreak()

```

8. Részletes tervez

being_chilling

```
create new Tekton as T2
add this to T2.neighbors
create new List as L with Tektons
add T2 to L
updateNeighbor(L, null)
end
```

```
method popSpore()
    If spores is empty then
        return null
    end if
    Create new spore temp
    temp = spores[0]
    remove first element from spores
    return temp
end
```

```
method growMushroomBody(ms: MushroomSpecies)
    if mushroomBody is not null then
        return false
    Create new MushroomBody with MushroomSpecies ms named mb
    mushroomBody = mb
    ms.add mb
    return true
end
```

```
-----method getNeighborsWithThread()
create new List<Tekton> tektList
if threads.size is bigger than 1 then
    for every fonal in thread
        add fonal.nextGrown[0].tekton to tektList
    end loop
    return tektList
end if
add thread[0].nextGrown[0].tekton to tektList
return tektList
end
```

```
method updateNeighbor(newAdd: List<Tekton>, delete: List<Tekton>)
    if newAdd is not null then
        for each tekton in newAdd
            if tekton is not in neighbors then
                addNeighbor(tekton)
            end if
        end loop
    end if
    if delete is not null then
        for each tekton in delete
            if tekton is in neighbors then
                deleteNeighbor(tekton)
            end if
        end loop
    end if
end
```

```
method absorb()
    for each fonal in threads
        if fonal.checkOwner is null then
            fonal.lifeReduce()
        end if
    end loop
end
```

8.1.5 MushroomlessTekton

- **Felelősség**
 - Az osztály a gombanélküli tekton megvalósítása, amely egy specializált tekton. A tekton speciális tulajdonsága, hogy nem nőhet rajta gombatest.
- **Ősosztályok**
 - Tekton → MushroomlessTekton
- **Interfészek**
(TektonController, TektonView)
- **Metódusok**
 - **+override bool growMushroomBody():** - Mivel ez egy gombanélküli tekton, nem engedélyezzük a gombatest növekedését, ezért minden hamis értéket ad vissza, jelezve, hogy nem sikerült a növekedés.

8.1.6 MultiThreadTekton

- **Felelősség**
 - Az osztály a többfonalas tekton megvalósítása, amely egy specializált tekton. A többfonalas tektonon több játékos fonala is tartózkodhat egyszerre.
- **Ősosztályok**
 - Tekton → MultiThreadTekton
- **Interfészek**
(TektonController, TektonView)
- **Metódusok**
 - **+override boolean addThread(mb:MushroomBody):**

```
method addThread(mt: MushroomThread)
    if mt is in threads
        return false
    end if
    add mt to threads
    return true
end
```

8.1.7 ThreadAbsorbTekton

- **Felelősség**
 - Az osztály a fonalszívó tekton megvalósítása, amely egy specializált tekton. A fonalszívó tekton időközönként felszívia (eltünteti) a rajta lévő gombafonalakat.
- **Ősosztályok**
 - Tekton → ThreadAbsorbTekton
- **Interfészek**
(TektonController, TektonView)
- **Metódusok**
 - **+override void absorb()**: - Elszív a tektonon lévő fonal életéből 1-et mindenkorábban 0 vagy alatti életest törli.

8.1.8 LifeSupportTekton

- **Felelősség**
 - Az osztály a LifeSupport tekton megvalósítása, amelynek elsődleges feladata, hogy megakadályozza a rajta lévő gombafonalak eltűnését, ami azt jelenti, hogy a tektonon lévő gombafonalaknak az életerejét sosem vonjuk le.
- **Ősosztályok**
 - Tekton → LifeSupportTekton
- **Interfészek**
(TektonController, TektonView)
- **Metódusok**
 - **+override void absorb()**: - A LifeSupportTekton megvédi a gombafonalakat a felszívódástól, így nem tűnik el.

8.1.9 InsectSpecies

- **Felelősség**

Ez az osztály tárolja el az összes rovarászhöz tartozó összes rovart.

- **Interfészek**
 -
- **Attribútumok**
 - **-List<Insect> insects:** Egy játékos (faj) összes rovarát tárolja el itt
- **Metódusok**
 - **+void addInsect(Insect ins)** - Hozzáad egy új rovart a nyilvántartó listához
 - **+void remove(Insect ins)** - Töröl egy rovart a listából

- +List<Insect> **getInsects()** - visszaadja az insects értékét.

8.1.10 Insect

- **Felelősség**

Az osztály felelőssége a rovarokkal kapcsolatos funkciók kezelése, attribútumainak tárolása.

- **Interfészek**

InsectView, InsectController

- **Attribútumok**

- **-int insectSpeed** - A rovar jelenlegi sebessége
- **-bool cutThread** – Ha igaz, akkor a rovar vághat fonalat, ha nem akkor nem.
- **-bool eatSpore** - Ez a változó mutatja meg, hogy ehet-e spórát a rovar vagy sem.
- **-int currentNutrient** – Mennyi tápanyagot fogyasztott el a rovar.
- **-Tekton currentPlace** – Eltárolja, hogy melyik tektonon van a rovar
- **-InsectSpecies species** - Melyik fajhoz tartozik a rovar
- **-int DefaultSpeed** - A rovar alapértelmezett sebessége

- **Metódusok**

- **+bool insectMove(Tekton t)** – A rovar megpróbál lépni. Akkor sikeres, ha van fonál a két tekton között, és szomszédosak.
- **+void insectEat()** – A rovar megpróbál enni. Ha sikeres, akkor megeszi a tektonon a legrégebben lévő spórát. Akkor sikertelen, ha eatSpore az false.
- **+void insectCut(MushroomThread mt)** - A rovar elvágja a kapott fonalat, ha vághat, és ha ugyanazon a tektonon van, mint a fonal.
- **+List<Tekton> getReachableTekton(int speed)** - Visszaadja azokat a tektonokat, amelyekhez vezet fonál.
- **+void hasteEffect()** - Megnöveli a rovar sebességét 1-el.
- **+void slowEffect()** - Csökkenti a rovar sebességét 1-el.
- **+void paraEffect()** - Csökkenti a rovar sebességét 0-ra, és a rovar vágó és evő képességét is elveszi amíg az effektus él.
- **+void muteEffect()** - Lenémítja a rovart, elvéve a vágó képességét az effektus idejéig
- **+void cloneEffect()** - Egy új rovart hoz létre, amely azonos a spórát megevő rovarral. A rovar az egyik szomszédon jön létre.
- **+void destroy()** - A rovar destruktora, elvégzi a szükséges tisztítási lépésekét, például megszünteti a rovarra mutató referenciait más osztályokban(null-ra állítja vagy eltávolítja egy listából).
- **int getInsectSpeed()** - visszatér insectSpeed értékével
- **boolean canCutThread()** - visszatér cutThread értékével.
- **boolean canEatSpore()** - visszatér eatSpore értékével.
- **int getCurrentNutrient()** - visszatér currentNutrient értékével
- **Tekton getLocation()** - visszatér currentPlace értékével.
- **void setLocation(Tekton location)** - currentPlace értékét átirja location értékére.

```
method insectMove(destination: Tekton)
    if insectSpeed is 0 then
        return false
    end if
    if destination not in currentPlace.getNeighborsWithThread then
```

```

        return false
    end if
    currentPlace = destination
    return true
end

-----
method insectEat()
    if size of currentPlace.spores smaller than 1 then
        return
    end if
    if eatSpore is false then
        return
    end if
    tempSporeNutrient = sporeNutrient of last element
    remove last element from list
    currentNutrient = tempSporeNutrient + currentNutrient
end

-----
method insectCut(target: MushroomThread)
    if cutThread is false then
        return
    end if
    if currentPlace is not target.tekton then
        return
    end if
    target.disconnectThread()
end

-----
method paraEffect()
    insectSpeed = 0
    cutThread = false
    eatSpore = false
end

-----
method cloneEffect()
    idx = 0
    for each tekton in currentPlace.neighbors
        if insectOnTekton is not null then
            break
        end if
        idx++
        if idx bigger than currentPlace.neighbors.size() then
            return;
        end if
    end loop
    I2Tekton = neighbors.get(idx)
    I2 = insect with location I2Tekton and same species
    species.addInsect(I2)
    I2Tekton.insectOnTekton = I2
end

-----
method endEffect()
    cutThread = true
    eatSpore = true
    insectSpeed = DefaultSpeed
end

```

8.1.11 Spore

- **Felelősség**

Egy szórásnyi spóra hatásaiért és tulajdonságaiért felelős osztály.

- **Interfészek**
SporeView, SporeController
- **Attribútumok**
 - **-int sporeNutrient** - Eltárolja, hogy mennyi tápanyagot tartalmaz egy szórásnyi spóra.
- **Metódusok**
 - **+void sporeEffect (Insect r)** - A leszármazó osztályok a hatásaiat ezzel a metódussal valósítják meg.

8.1.12 MuteSpore

- **Felelősség**
Az osztály a némitő spórát valósítja meg.
- **Ősosztályok**
Spore → MuteSpore
- **Interfészek**
- **Metódusok**
 - **+override void sporeEffect (Insect r)** - Lenémítja az adott rovart.

8.1.13 ParaSpore

- **Felelősség**
Az osztály a bénítő spórát valósítja meg.
- **Ősosztályok**
Spore → ParaSpore
- **Interfészek**
(SporeView, SporeController)
- **Metódusok**
 - **+ override void sporeEffect (Insect r)** - Lebénítja az adott rovart.

8.1.14 HasteSpore

- **Felelősség**
Az osztály a gyorsító spórát valósítja meg.

- **Ősosztályok**
Spore → HasteSpore
- **Interfészek**
(SporeView, SporeController)
- **Metódusok**
 - **+override void sporeEffect (Insect r)** - Felgyorsítja az adott rovart.

8.1.15 SlowSpore

- **Felelősség**
Az osztály a lassító spórát valósítja meg.
- **Ősosztályok**
Spore → SlowSpore
- **Interfészek**
(SporeView, SporeController)
- **Metódusok**
 - **+override void sporeEffect (Insect r)** - Lelassítja az adott rovart.

8.1.16 CloneSpore

- **Felelősség**
Az osztály a klónozó spórát valósítja meg.
- **Ősosztályok**
Spore → CloneSpore
- **Interfészek**
(SporeView, SporeController)
- **Metódusok**
 - **+override void sporeEffect (Insect r)** - Leklónozza az adott rovart.

8.1.17 ControllerComponent

- **Felelősség**
Ez az osztály a kontrollert valósítja meg a modellben.
- **Ősosztályok**
 -

- **Interfészek**
 -
- **Attribútumok**
 - - **ViewComponent** viewComponent - A modell View részére biztosít hozzáférést.
- **Metódusok**
 - + **void gameLoop()** - A játék logikai részét valósítja meg.
 - + **void growThread(TektonController source, MushroomThreadController newThread, TektonController target)** - Ez a parancs, amit a View hív meg, a modellben egy új fonalat növeszt az adott paraméterekkel.
 - + **void growMushroom(MushroomBodyController newMushroom, TektonController target)** - Ez egy parancs, amit a view adott a controllernek, ez a modellben új gombát növeszt az adott paraméterekkel.
 - + **void spreadSpore(MushroomBodyController source, TektonController target)** - Ez egy parancs, amit a view adott a controllernek, ez a modellben spórát szór az adott paraméterekkel.
 - + **void absorbInsect(MushroomThreadController source, MushroomBodyController newMushroom)** - Ez egy parancs, amit a view adott a controllernek, ez a modellben felszív egy rovart az adott paraméterekkel.
 - - **void breakTekton(TektonController source, TektonController newTekton)** - A controller hívja meg a gameLoop függvényben, ez a modellben eltör egy tektont az adott paraméterekkel.
 - - **void absorb(List<TektonController> tektons)** - A controller hívja meg a gameLoop függvényben, ez a modellben a fonalat szívja fel (nem teljesen) az adott paramétereken.
 - + **void cut(InsectController insect, MushroomThreadController target)** - Ez egy parancs, amit a view adott a controllernek, ez a modellben elvág egy fonalat az adott paraméterekkel.
 - + **void eat(InsectController insect)** - Ez egy parancs, amit a view adott a controllernek, ez a modellben megesz egy spórát az adott paraméterekkel.
 - + **void move(InsectController insect, TektonController target)** - Ez egy parancs, amit a view adott a controllernek, ez a modellben mozgat egy rovarat az adott paraméterekkel.
 - + **void load(File file)** - Beolvas egy fájlt, ami a kezdeti helyzete lesz a pályának.

8.1.18 ViewComponent

- **Felelősség**
A modell View részét valósítja meg.
- **Interfészek**
 -
- **Attribútumok**
 - - **ControllerComponent controllerComponent** - a View hozzáférése a Controller komponenshez
- **Metódusok**

- + **void setControllerComponent(ControllerComponent controllerComponent)** - beállítja a view-hoz a controller-t (hozzáférést ad a view a controllerhez)
- + **void validate(String command)** - A beérkező commandot értelmezi és meghívja a controller megfelelő függvényét a commandtól függően.
- + **void help()** - A playernek szükséges parancsokat írja ki.
- + **void showId()** - Az összes id-t kiírja a konzolra
- + **void showMap()** - A mappot kiírja a konzolra.
- + **void unsuccessfulCommand()** - Ha nem sikeres a parancs, akkor kiírja a kimenetet.

8.1.19 GameModel

- Felelősség

Ez az osztály tárol minden objektumot a játékban

- Interfészek
-

- Attribútumok

+**Static Map map:** Az összes tekton ami a térképet felépíti.

+**Static List<Gombasz> gombasz:** Az összes gombász ebben van.

+**Static List<Rovarasz> rovarasz:** Az összes rovarász ebben van.

+**Static List<Object> gameObjects:** Az összes többi elem, pl. fonalak, gombatestek itt vannak.

8.1.20 Rovarasz

- Felelősség

Ez a rovarász szerepét megvalósító osztály.

- Ősosztályok

Player → Rovarasz

- Attribútumok

+**InsectSpecies insectSpecies:** minden játékosnak saját rovarfaja van, itt tárolja az összes rovarát.

8.1.21 Gombasz

- Felelősség

Ez a gombász szerepét megvalósító osztály.

- Ősosztályok

Player → Gombasz

- Attribútumok

+**MushroomSpecies mushroomSpecies:** minden játékosnak saját gombafaja van, itt tárolja az összes gombatestét.

8.1.22 Player

- Felelősség

Ez egy absztrakt osztály, közös őse a rovarásznak és a gombásznak, minden két szerep ugyanis egy játékos.

- Attribútumok

+**int score:** A játékos összes pontszámát tárolja el.

8.1.23 Map

- Felelősség

Ez a játékot felépítő tektonokat tárolja el, emiatt is térkép a neve.

- Attribútumok

+List<Tekton> **tektonList**: A lista, amely az összes tektont tárolja.

8.2 A tesztek részletes tervezeti leírásuk a teszt nyelvén

8.2.1 Rovar move success

- Leírás

A teszt ellenőrzi, hogy a rovar sikeresen át tud-e lépni egy tektonra, ha minden feltétel teljesül (van kapcsolatban álló gombafonal, van elég sebessége, és nincs bénulás alatt).

- Ellenőrzött funkcionalitás, várható hibahelyek

Rovar mozgásának a sikerességét ellenőrzi. (Insect.insectMove()).

- Bemenet

```
/addplayer test1 R
/addtekton t1
/addtekton t2
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addthread t1 f1
/addthread t2 f2
/addthread t3 f3
/connectthread f1 f2
/connectthread f2 f3
/addinsect test1 i1 t1
/move i1 t3
/showMap
```

- Elvárt kimenet

```
t1(type=N; threads=f1; neighbours=t2)
t2(type=N; threads=f2; neighbours=t1, t3)
t3(type=N; threads=f3; neighbours=t2; insect=i1)
f1(life=3; lifesupport=0; tekton=t1; next=f2)
f2(life=3; lifesupport=0; tekton=t2; prev=f1; next=f3)
f3(life=3; lifesupport=0; tekton=t3; prev=f2)
i1(speed=2; cancut=true; caneat=true; tekton=t3; nutrient=0)
test1(type=R; own=i1)
```

8.2.2 Rovar move unsuc (no fonal)

- Leírás

A teszt azt ellenőrzi, hogy a rovar tud-e mozogni, ha nincs kapcsolatban álló gombafonal.

- Ellenőrzött funkcionalitás, várható hibahelyek

Rovar mozgásának a sikertelenségét ellenőrzi. (Insect.insectMove()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addtekton t2
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addinsect test1 i1 t1
/move i1 t3
/showMap
```

- **Elvárt kimenet**

```
move failed
t1(type=N; neighbours=t2; insect=i1)
t2(type=N; neighbours=t1, t3)
t3(type=N; neighbours=t2;)
i1(speed=2; cancut=true; caneat=true; tekton=t1; nutrient=0)
test1(type=R; own=i1)
```

8.2.3 Rovar move unsec (no enough speed)

- **Leírás**

A teszt azt ellenőrzi, hogy a rovar tud-e mozogni, ha nincs elegendő sebessége a kiválasztott tekton eléréséhez.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Rovar mozgásának a sikertelenségét ellenőrzi. (Insect.insectMove()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addtekton t2
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addthread t1 f1
/addthread t2 f2
/addthread t3 f3
/connectthread f1 f2
/connectthread f2 f3
/addinsect test1 i1 t1 S
/move i1 t3
/showMap
```

- **Elvárt kimenet**

```
move failed
t1(type=N; threads=f1; neighbours=t2; insect=i1)
t2(type=N; threads=f2; neighbours=t1, t3)
t3(type=N; threads=f3; neighbours=t2;)
f1(life=3; lifesupport=0; tekton=t1; next=f2)
f2(life=3; lifesupport=0; tekton=t2; prev=f1; next=f3)
f3(life=3; lifesupport=0; tekton=t3; prev=f2)
```

```
i1(speed=1; cancut=true; caneat=true; tekton=t1; nutrient=0)
test1(type=R; own=i1)
```

8.2.4 Rovar move unsec (para)

- **Leírás**

A teszt ellenőrzi, hogy a rovar tud-e mozogni, ha bénulás alatt van.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Rovar mozgásának a sikertelenségét ellenőrzi. (Insect.insectMove()).

- **Bemenet**

```
/addplayer test1 R
```

```
/addtekton t1
```

```
/addtekton t2
```

```
/addtekton t3
```

```
/setneighbour t1 t2
```

```
/setneighbour t2 t3
```

```
/addthread t1 f1
```

```
/addthread t2 f2
```

```
/addthread t3 f3
```

```
/connectthread f1 f2
```

```
/connectthread f2 f3
```

```
/addinsect test1 i1 t1 P
```

```
/move i1 t3
```

```
/showMap
```

- **Elvárt kimenet**

move failed

```
t1(type=N; threads=f1; neighbours=t2; insect=i1)
```

```
t2(type=N; threads=f2; neighbours=t1, t3)
```

```
t3(type=N; threads=f3; neighbours=t2;)
```

```
f1(life=3; lifesupport=0; tekton=t1; next=f2)
```

```
f2(life=3; lifesupport=0; tekton=t2; prev=f1; next=f3)
```

```
f3(life=3; lifesupport=0; tekton=t3; prev=f2)
```

```
i1(speed=0; cancut=false; caneat=false; tekton=t1; nutrient=0)
```

```
test1(type=R; own=i1)
```

8.2.5 Rovar eat spore success

- **Leírás**

A teszt ellenőrzi, hogy a rovar sikeresen meg tudja-e enni a spórát a tektonon, és megkapja-e a tápanyagot.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Spóra evésnek a sikerességét ellenőrzi. (Insect.insectEat()).

- **Bemenet**

```
/addplayer test1 R
```

```
/addtekton t1
```

```
/addinsect test1 i1 t1
```

```
/addspore t1 s1
```

```
/eat i1
```

```
/showMap
```

- **Elvárt kimenet**

```
t1(type=N; insect=i1)
i1(speed=2; cancut=true; caneat=true; tekton=t1; nutrient=5)
test1(type=R; own=i1)
```

8.2.6 Rovar eat spore unsec (no spore)

- **Leírás**

A teszt ellenőrzi, hogy a rovar nem tudja-e enni a spórát a tektonon, ha nincs rajta spóra.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Spóra evésnek a sikertelenségét ellenőrzi. (Insect.insectEat()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addinsect test1 i1 t1
/eat i1
/showMap
```

- **Elvárt kimenet**

eat failed

```
t1(type=N; insect=i1)
i1(speed=2; cancut=true; caneat=true; tekton=t1; nutrient=0)
test1(type=R; own=i1)
```

8.2.7 Rovar eat spore unsec (para)

- **Leírás**

A teszt ellenőrzi, hogy a rovar nem tudja megenni a spórát, ha bénulás alatt van.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Spóra evésnek a sikertelenségét ellenőrzi. (Insect.insectEat()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addinsect test1 i1 t1 P
/addspore t1 s1
/eat i1
/showMap
```

- **Elvárt kimenet**

eat failed

```
t1(type=N; insect=i1; spore=s1)
i1(speed=0; cancut=false; caneat=false; tekton=t1; nutrient=0)
s1(nutrient=0; type=N)
test1(type=R; own=i1)
```

8.2.8 Rovar cut success

- **Leírás**

A teszt ellenőrzi, hogy a rovar sikeresen el tudja vágni a gombafonalat, ha minden feltétel teljesül.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Fonal vágásnak a sikereségét ellenőrzi. (Insect.insectCut()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addtekton t2
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addthread t1 f1
/addthread t2 f2
/addthread t3 f3
/connectthread f1 f2
/connectthread f2 f3
/addinsect test1 i1 t2
/cut i1 f2
/showMap
```

- **Elvárt kimenet**

```
t1(type=N; threads=f1; neighbours=t2; insect=i1)
t2(type=N; threads=f2; neighbours=t1, t3)
t3(type=N; threads=f3; neighbours=t2;)
f1(life=3; lifesupport=0; tekton=t1;)
f2(life=3; lifesupport=0; tekton=t2; next=f3)
f3(life=3; lifesupport=0; tekton=t3; prev=f2)
i1(speed=2; cancut=true; caneat=true; tekton=t2; nutrient=0)
test1(type=R; own=i1)
```

8.2.9 Rovar cut unsec(mute)

- **Leírás**

A teszt ellenőrzi, hogy a rovar nem tudja elvágni a gombafonalat, ha el van némítva.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Fonal vágásnak a sikertelenségét ellenőrzi. (Insect.insectCut()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addtekton t2
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addthread t1 f1
/addthread t2 f2
/addthread t3 f3
/connectthread f1 f2
/connectthread f2 f3
/addinsect test1 i1 t2 M
/cut i1 f2
/showMap
```

- **Elvárt kimenet**

```
cut failed
t1(type=N; threads=f1; neighbours=t2; insect=i1)
t2(type=N; threads=f2; neighbours=t1, t3)
```

```
t3(type=N; threads=f3; neighbours=t2;)
f1(life=3; lifesupport=0; tekton=t1; next=f2)
f2(life=3; lifesupport=0; tekton=t2; prev=f1; next=f3)
f3(life=3; lifesupport=0; tekton=t3; prev=f2)
i1(speed=2; cancut=false; caneat=true; tekton=t2; nutrient=0)
test1(type=R; own=i1)
```

8.2.10 Rovar cut unsec(para)

- **Leírás**

A teszt ellenőrzi, hogy a rovar nem tudja elvágni a gombafonalat, ha bénulás alatt van.

- **Ellenőrzött funkcionális, várható hibahelyek**

Fonal vágásnak sikertelenségét ellenőrzi. (Insect.insectCut()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addtekton t2
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addthread t1 f1
/addthread t2 f2
/addthread t3 f3
/connectthread f1 f2
/connectthread f2 f3
/addinsect test1 i1 t2 P
/cut i1 f2
/showMap
```

- **Elvárt kimenet**

```
cut failed
t1(type=N; threads=f1; neighbours=t2; insect=i1)
t2(type=N; threads=f2; neighbours=t1, t3)
t3(type=N; threads=f3; neighbours=t2;)
f1(life=3; lifesupport=0; tekton=t1; next=f2)
f2(life=3; lifesupport=0; tekton=t2; prev=f1; next=f3)
f3(life=3; lifesupport=0; tekton=t3; prev=f2)
i1(speed=0; cancut=false; caneat=false; tekton=t2; nutrient=0)
test1(type=R; own=i1)
```

8.2.11 Rovar cut unsec(no thread)

- **Leírás**

A teszt ellenőrzi, hogy a rovar nem tudja elvágni a gombafonalat, ha nincs gombafonal a tektonon.

- **Ellenőrzött funkcionális, várható hibahelyek**

Fonal vágásnak a sikertelenségét ellenőrzi. (Insect.insectCut()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addtekton t2
```

```
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addinsect test1 i1 t2
/cut i1 f2
/showMap
```

- **Elvárt kimenet**

cut failed
t1(type=N; neighbours=t2; insect=i1)
t2(type=N; neighbours=t1, t3)
t3(type=N; neighbours=t2;)
i1(speed=2; cancut=true; caneat=true; tekton=t2; nutrient=0)
test1(type=R; own=i1)

8.2.12 Rovar cut unsec(already cut)

- **Leírás**

A teszt ellenőrzi, hogy a rovar nem tudja elvágni a gombafonalat, ha már el lett vágva

- **Ellenőrzött funkcionális, várható hibahelyek**

Fonal vágásnak a sikertelenségét ellenőrzi. (Insect.insectCut()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addtekton t2
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addthread t1 f1
/addthread t2 f2
/addthread t3 f3
/connectthread f2 f3
/addinsect test1 i1 t2
/cut i1 f2
/showMap
```

- **Elvárt kimenet**

cut failed
t1(type=N; threads=f1; neighbours=t2; insect=i1)
t2(type=N; threads=f2; neighbours=t1, t3)
t3(type=N; threads=f3; neighbours=t2;)
f1(life=3; lifesupport=0; tekton=t1;)
f2(life=3; lifesupport=0; tekton=t2; next=f3)
f3(life=3; lifesupport=0; tekton=t3; prev=f2)
i1(speed=2; cancut=true; caneat=true; tekton=t2; nutrient=0)
test1(type=R; own=i1)

8.2.13 Eat haste spore

- **Leírás**

A teszt ellenőrzi, hogy a rovar megkapja a gyorsító hatást, ha a gyorsító spórát elfogyasztja.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Gyorsító spóra effektus megadás sikereségét ellenőrzi. (Insect.hasteEffect()).

- **Bemenet**

/addplayer test1 R

/addtekton t1

/addinsect test1 i1 t1

/addspore t1 s1 H

/eat i1

/showMap

- **Elvárt kimenet**

t1(type=N; insect=i1)

i1(speed=3; cancut=true; caneat=true; tekton=t1; nutrient=5)

test1(type=R; own=i1)

8.2.14 Eat slow spore

- **Leírás**

A teszt ellenőrzi, hogy a rovar megkapja a lassító hatást ,ha lassító spórát elfogyasztja.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Lassító spóra effektus megadás sikereségét ellenőrzi. (Insect.slowEffect()).

- **Bemenet**

/addplayer test1 R

/addtekton t1

/addinsect test1 i1 t1

/addspore t1 s1 S

/eat i1

/showMap

- **Elvárt kimenet**

t1(type=N; insect=i1)

i1(speed=1; cancut=true; caneat=true; tekton=t1; nutrient=5)

test1(type=R; own=i1)

8.2.15 Eat mute spore

- **Leírás**

A teszt ellenőrzi, hogy a rovar megkapja a némitó hatást, ha a némitó spórát elfogyasztja.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Némító spóra effektus megadás sikereségét ellenőrzi. (Insect.muteEffect()).

- **Bemenet**

/addplayer test1 R

/addtekton t1

/addinsect test1 i1 t1

/addspore t1 s1 M

/eat i1

/showMap

- **Elvárt kimenet**

```
t1(type=N; insect=i1)
i1(speed=2; cancut=false; caneat=true; tekton=t1; nutrient=5)
test1(type=R; own=i1)
```

8.2.16 Eat para spore

- **Leírás**

A teszt ellenőrzi, hogy a rovar megkapja a bénító hatást, ha a bénító spórát elfogyasztja.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Bénító spóra effektus megadás sikereségét ellenőrzi. (Insect.paraEffect()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addinsect test1 i1 t1
/addspore t1 s1 P
/eat i1
/showMap
```

- **Elvárt kimenet**

```
t1(type=N; insect=i1)
i1(speed=0; cancut=false; caneat=false; tekton=t1; nutrient=5)
test1(type=R; own=i1)
```

8.2.17 Eat clone spore

- **Leírás**

A teszt ellenőrzi, hogy a rovar sikeresen klónozza-e önmagát, miután megeszi a klónozó spórát, és létrejön-e egy új, független rovar az eredeti rovartól.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Klón spóra effektus megadás sikereségét ellenőrzi. (Insect.cloneEffect()).

- **Bemenet**

```
/addplayer test1 R
/addtekton t1
/addtekton t2
/setneighbour t1 t2
/addinsect test1 i1 t1
/addspore t1 s1 C
/eat i1
/showMap
```

- **Elvárt kimenet**

```
t1(type=N; threads=f1; neighbours=t2; insect=i1)
t2(type=N; threads=f2; neighbours=t1, t3)
i1(speed=2; cancut=true; caneat=true; tekton=t1; nutrient=5)
i1_clone(speed=2; cancut=true; caneat=true; tekton=t2; nutrient=0)
test1(type=R; own=i1, i1_clone)
```

8.2.18 Spread spore success

- **Leírás**

A teszt azt ellenőrzi, hogy a gombatest sikeresen szórja-e ki a spórákat a kiválasztott tektonra, amennyiben minden feltétel teljesül (van elég spóra, a tekton elérhető távolságban van).

- **Ellenőrzött funkcionális, várható hibahelyek**

Spóra szórásnak a sikereségét ellenőrzi. (`MushroomBody.spreadSpore()`).

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addtekton t2
/setneighbour t1 t2
/addmush test1 m1 f0 t1 3 3
/spreadspore m1 t2
/showMap
```

- **Elvárt kimenet**

```
t1(type=N; mushroom=m1; neighbours=t2)
t2(type=N; neighbours=t1; spores=m1_spore3)
f0(life=3; lifesupport=true; t1)
m1(Age=3; spore=2; t1)
m1_spore3(nutrient=5; N)
test1(type=G; own=m1)
```

8.2.19 Spread spore unsec(no spore)

- **Leírás**

A teszt azt ellenőrzi, hogy a gombatest nem tudja spórát szórni, ha nincs spóra a gombatestben.

- **Ellenőrzött funkcionális, várható hibahelyek**

Spóra szórásnak a sikertelenségét ellenőrzi.

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addtekton t2
/setneighbour t1 t2
/addmush test1 m1 f0 t1 3 0
/spreadspore m1 t2
/showMap
```

- **Elvárt kimenet**

```
spread spore failed
t1(type=N; neighbours=t2)
t2(type=N; neighbours=t1)
f0(life=3; lifesupport=true; t1)
m1(Age=3; spore=0; t1)
test1(type=G; own=m1)
```

8.2.20 Spread spore unsec(too far)

- **Leírás**

A teszt azt ellenőrzi, hogy a gombatest nem tudja spórát szórni, ha a kiválasztott tekton túl messze van, azaz nem esik a gombatest hatósugarán belül.

- **Ellenőrzött funkcionális, várható hibahelyek**

Spóra szórásnak a sikertelenségét ellenőrzi.

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addtekton t2
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addmush test1 m1 f0 t1 3 3
/spreadspore m1 t3
/showMap
```

- **Elvárt kimenet**

```
spread spore failed
t1(type=N; mushroom=m1; neighbours=t2)
t2(type=N; neighbours=t1, t3)
t3(type=N; neighbours=t2;)
f0(life=3; lifesupport=true; t1)
m1(Age=3; spore=3; t1)
test1(type=G; own=m1)
```

8.2.21 Grow mushroom success

- **Leírás**

A teszt azt ellenőrzi, hogy a gombatest sikeresen kinő-e a kiválasztott tektonon, ha minden feltétel teljesül (elegendő spóra, gombafonal jelenléte).

- **Ellenőrzött funkcionális, várható hibahelyek**

Gomba növezsétsnek a sikerességét ellenőrzi.

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addtekton t2
/setneighbour t1 t2
/addmush test1 m1 f0 t1 3 3
/addthred t2 f1
/connectthread f0 f1
/addspore t2 s1
/addspore t2 s2
/addspore t2 s3
/growmush test1 m2 t2
/showMap
```

- **Elvárt kimenet**

```
t1(type=N; mushroom=m1; threads=f0; neighbours=t2)
t2(type=N; mushroom=m2; threads=f1; neighbours=t1)
m1(Age=3; Spore=3; tekton=t1)
m2(Age=3; Spore=3; tekton=t2)
f0(life=3; lifeSupport=true; tekton=t1; next=f1)
f1(life=3; lifeSupport=true; tekton=t2; prev=f0)
test1(Type=G, Own=m1, m2)
```

8.2.22 Grow mushroom unsec(no enough spore)

- **Leírás**

A teszt azt ellenőrzi, hogy a gombatest nem tudja kinöni a kiválasztott tektonon, ha nincs elegendő spóra mennyisége a gombatestben

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Gomba növesztésnek a sikertelenségét ellenőrzi.

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addtekton t2
/setneighbour t1 t2
/addmush test1 m1 f0 t1 3 3
/addthred t2 f1
/connectthread f0 f1
/growmush test1 m2 t2
/showMap
```

- **Elvárt kimenet**

grow mushroom failed
t1(type=N; mushroom=m1; threads=f0; neighbours=t2)
t2(type=N; threads=f1; neighbours=t1)
m1(Age=3; Spore=3; tekton=t1)
f0(life=3; lifeSupport=true; tekton=t1; next=f1)
f1(life=3; lifeSupport=true; tekton=t2; prev=f0)
test1(Type=G, Own=m1)

8.2.23 Grow mushroom unsec(no fonal)

- **Leírás**

A teszt azt ellenőrzi, hogy a gombatest nem tudja kinöni a kiválasztott tektonon, ha nincs gombafonal a tektonon.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Gomba növesztésnek a sikertelenségét ellenőrzi.

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addtekton t2
/setneighbour t1 t2
/addmush test1 m1 f0 t1
/addspore t2 s1
/addspore t2 s2
/addspore t2 s3
/growmush test1 m2 t2
/showMap
```

- **Elvárt kimenet**

grow mushroom failed
t1(type=N; mushroom=m1; neighbours=t2)
t2(type=N; spores=s1,s2,s3; neighbours=t1)
s1(nutrient=5; type=N)
s2(nutrient=5; type=N)

```
s3(nutrient=5; type=N)
m1(Age=3; Spore=3; tekton=t1)
test1(Type=G, Own=m1)
```

8.2.24 Grow mushroom unsec(mushroomless)

- **Leírás**

A teszt azt ellenőrzi, hogy a gombatest nem tudja kinöni a kiválasztott tektonon, ha az a tekton "gomba nélküli" típusú, azaz nem alkalmas gomba növesztésére.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Gomba növesztésnek a sikertelenségét ellenőrzi.

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addtekton t2 MS
/setneighbour t1 t2
/addmush test1 m1 f0 t1
/addthred t2 f1
/connectthread f0 f1
/addspore t2 s1
/addspore t2 s2
/addspore t2 s3
/growmush test1 m2 t2
/showMap
```

- **Elvárt kimenet**

```
grow mushroom failed
t1(type=N; mushroom=m1; neighbours=t2; threads=f0)
t2(type=MS; spores=s1, s2, s3; neighbours=t1; threads=f1)
m1(Age=3; Spore=3; tekton=t1)
f0(life=3; lifeSupport=true; tekton=t1; next=f1)
f1(life=3; lifeSupport=true; tekton=t2; prev=f0) s1(Nutrient=5; type=N)
s2(Nutrient=5; type=N)
s3(Nutrient=5; type=N)
test1(Type=G, Own=m1)
```

8.2.25 Grow fonal success

- **Leírás**

A teszt azt ellenőrzi, hogy a gombatest sikeresen növeszt-e új gombafonalat a kiválasztott tektonra, amennyiben minden feltétel teljesül (van elegendő erőforrás, a tekton elérhető, stb.)

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Fonal növesztésnek a sikerességét ellenőrzi.

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addtekton t2
/setneighbour t1 t2
/addmush test1 m1 f0 t1
/growthread f0 f1 t2
/showMap
```

- **Elvárt kimenet**

```
t1(type=N; mushroom=m1; neighbours=t2; threads=f0)
t2(type=N; neighbours=t1; threads=f1)
m1(Age=3; Spore=3; tekton=t1)
f0(life=3; lifeSupport=true; tekton=t1; next=f1)
f1(life=3; lifeSupport=true; tekton=t2; prev=f0)
test1(Type=G, Own=m1)
```

8.2.26 Grow fonal unsec(occupied)

- **Leírás**

A teszt azt ellenőrzi, hogy a gombatest nem tudja új gombafonalat növeszteni a kiválasztott tektonra, ha az a tekton már foglalt, azaz már van rajta gombafonal

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Fonal növesztésnek a sikertelenségét ellenőrzi.

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addtekton t2
/setneighbour t1 t2
/addmush test1 m1 f0 t1
/addthread t2 f1
/growthread f0 f1 t2
/showMap
```

- **Elvárt kimenet**

grow thread failed

```
t1(type=N; mushroom=m1; neighbours=t2; threads=f0)
t2(type=N; neighbours=t1; threads=ffff1)
m1(Age=3; Spore=3; tekton=t1)
f0(life=3; lifeSupport=true; tekton=t1)
ffff1(life=3; lifeSupport=true; tekton=t2)
test1(Type=G; Own=m1)
```

8.2.27 Grow fonal random

- **Leírás**

A teszt azt ellenőrzi, hogy a gombafonal növekedése felgyorsul és tovább nő, ha spóra van a tektonon, ahová a fonal növekszik.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Fonal növesztés gyorsítását ellenőrzi.

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addtekton t2
/addtekton t3
/addtekton t4
/setneighbour t1 t2
/setneighbour t2 t3
/setneighbour t2 t4
/setneighbour t3 t4
/addmush test1 m1 f0 t1
```

```
/addspore t2 s1
/growththread f0 f1 t2
/showMap
• Elvárt kimenet
t1(type=N; mushroom=m1; neighbours=t2; threads=f0)
t2(type=N; spores=s1; neighbours=t1, t3; threads=f1)
t3(type=N; neighbours=t2, t4)
t4(type=N; neighbours=t3)
m1(Age=3; Spore=3; tekton=t1)
f0(life=3; lifeSupport=true; tekton=t1; next=f1)
f1(life=3; lifeSupport=true; tekton=t2; prev=f0; next=f2)
f2(life=3; lifeSupport=true; tekton=t3; prev=f0)
s1(Nutrient=5; type=N)
test1(Type=G; Own=m1)
```

8.2.28 Tekton Absorb

- Leírás

A teszt azt ellenőrzi, hogy a `Tekton.absorb()` metódus megfelelően csökkenti-e a gombafonalak életterejét a tektonon, ha azok nem kapcsolódnak életben tartó struktúrához (pl. gombatesthez vagy LifeSupportTektonhoz).

- Ellenőrzött funkcionalitás, várható hibahelyek

Tekton absorb és élet csökkenése ellenőrzi.

- Bemenet

```
/addtekton t1
/addthread t1 f1
/addtekton t2 T
/addthread t2 f2
/addtekton t3 L
/addthread t3 f3
/absorb
/showMap
```

- Elvárt kimenet

```
t1(type=N; threads=f1)
t2(type=T; threads=f2)
t3(type=L; threads=f3)
f1(life=2; lifeSupport=false; tekton=t1)
f2(life=2; lifeSupport=false; tekton=t2)
f3(life=3; lifeSupport=false; tekton=t3)
```

8.2.29 Tekton Absorb*3

- Leírás

A teszt azt ellenőrzi, hogy a `Tekton.absorb()` metódus háromszor egymás után meghívva, megfelelően csökkenti-e a gombafonalak életterejét a tektonon
`MushroomThread.lifeReduce()` meghívva, ha sima gombafonal akkor háromszor
`MushroomThread.lifeReduce()` meghívás után eltűnik.

- Ellenőrzött funkcionalitás, várható hibahelyek

Tekton absorb és élet csökkenése ellenőrzi

- Bemenet

```
/addtekton t1
```

```
/addthread t1 f1
/addtekton t2 T
/addthread t2 f2
/addtekton t3 L
/addthread t3 f3
/absorb
/absorb
/absorb
/showMap
```

- **Elvárt kimenet**

t1(type=N)
t2(type=T)
t3(type=L; threads=f3)
f3(life=3; lifeSupport=false; tekton=t3)

8.2.30 Tekton break success

- **Leírás**

A teszt ellenőrzi, hogy a tekton sikeresen kettétörök-e, és helyette két új tekton jön-e létre.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Tekton ketté törlésnek a sikerességét ellenőrzi.

- **Bemenet**

```
/addtekton t1
/addthread t1 f1
/break t1 tb2
/showMap
```

- **Elvárt kimenet**

t1(type=N; neighbours=tb2)
tb2(type=N; neighbours=t1)

8.2.31 Tekton break unsec(gomba)

- **Leírás**

A teszt ellenőrzi, hogy a tekton nem törhető ketté, ha gombatest van rajta.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Tekton ketté törlésnek a sikertelenségét ellenőrzi.

- **Bemenet**

```
/addplayer test1 G
/addtekton t1
/addmush test1 m1 f0 t1
/break t1 tb2
/showMap
```

- **Elvárt kimenet**

break failed
t1(type=N; mushroom=m1; threads=f0)
m1(Age=3; Spore=3; tekton=t1)
f0(life=3; lifeSupport=true; tekton=t1)
test1(Type=G; Own=m1)

8.2.32 Tekton break unsec(rovar)

- Leírás

A teszt ellenőrzi, hogy a tekton nem törhető ketté, ha rovar van rajta.

- Ellenőrzött funkcionalitás, várható hibahelyek

Tekton ketté törlésnek a sikertelenségét ellenőrzi.

- Bemenet

```
/addplayer test1 R
/addtekton t1
/addinsect test1 i1 t1
/break t1 tb2
/showMap
```

- Elvárt kimenet

```
break failed
t1(type=N; insect=i1)
i1(speed=3; cancut=true; caneat=true; tekton=t1; nutrient=0)
test1(Type=R; Own=i1)
```

8.2.33 Absorb rovar success

- Leírás

A teszt ellenőrzi, hogy a gombafonal sikeresen megeszi-e a bénult rovart a tektonon

- Ellenőrzött funkcionalitás, várható hibahelyek

A fonal rovar evésének sikerességét ellenőrzi.

- Bemenet

```
/addplayer P1 R
/addplayer P2 G
/addtekton t1
/addtekton t2
/setneighbour t1 t2
/addmush P2 m1 f0 t1
/addthread t2 f1
/connectthread f0 f1
/addinsect P1 i1 t2 P
/absorbinsect f2 m2
/showMap
```

- Elvárt kimenet

```
t1(type=N; mushroom=m1; neighbours=t2; threads=f0)
t2(type=N; mushroom=m2; neighbours=t1; threads=f1)
m1(Age=3; Spore=3; tekton=t1)
m2(Age=1; Spore=3; tekton=t2)
f0(life=3; lifeSupport=true; tekton=t1; next=f1)
f1(life=3; lifeSupport=true; tekton=t2; prev=f0)
P1(Type=R)
P2(Type=G; Own=m1,m2)
```

8.2.34 Absorb rovar unsec(not para)

- Leírás

A teszt ellenőrzi, hogy a gombafonal nem tudja megenni a rovart, ha az nem bénult állapotban van.

- **Ellenőrzött funkcionális, várható hibahelyek**

A fonal rovar evésének sikertelenségét ellenőrzi.

- **Bemenet**

```
/addplayer P1 R
/addplayer P2 G
/addtekton t1
/addtekton t2
/setneighbour t1 t2
/addmush P2 m1 f0 t1
/addthread t2 f1
/connectthread f0 f1
/addinsect P1 i1 t2
/absorbinsect f2 m2
/showMap
```

- **Elvárt kimenet**

```
absorb failed
t1(type=N; mushroom=m1; neighbours=t2; threads=f0)
t2(type=N; insect=i1; neighbours=t1; threads=f1)
m1(Age=3; Spore=3; tekton=t1)
f0(life=3; lifeSupport=true; tekton=t1; next=f1)
f1(life=3; lifeSupport=true; tekton=t2; prev=f0)
i1(speed=2; cancut=true; caneat=true; tekton=t2; nutrient=0)
P1(Type=R; Own=i1)
P2(Type=G; Own=m1)
```

8.2.35 Rovar összetett teszt

- **Leírás**

A teszt ellenőrzi, a rovar evés, mozgás, vágást tevékenységét.

- **Ellenőrzött funkcionális, várható hibahelyek**

A rovarnak a tevékenységét ellenőrzi

- **Bemenet**

```
/addplayer P1 R
/addtekton t1
/addtekton t2
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addthread t1 f1
/addthread t2 f2
/addthread t3 f3
/connectthread f1 f2
/connectthread f2 f3
/addspore t2 s1
/addinsect P1 i1 t1
/move i1 t2
/eat i1
/cut i1 f2
```

/showMap

- **Elvárt kimenet**

```
t1(type=N; neighbours=t2; threads=f1)
t2(type=N; insect=i1; neighbours=t1, t3; threads=f2)
t3(type=N; neighbours=t2; threads=f3)
f1(life=3; lifeSupport=false; tekton=t1)
f2(life=3; lifeSupport=false; tekton=t2; next=f3)
f3(life=3; lifeSupport=false; tekton=t3; prev=f2)
i1(speed=2; cancut=true; caneat=true; tekton=t2; nutrient=5)
P1(Type=R; Own=i1)
```

8.2.36 Gomba összetett teszt

- **Leírás**

A teszt ellenőrzi, a gomba növesztést, a gombának a spóra szórását és a gombafonal növesztését.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A gomba összetett funkcióit ellenőrzi.

- **Bemenet**

```
/addplayer P1 G
/addtekton t1
/addtekton t2
/addtekton t3
/setneighbour t1 t2
/setneighbour t2 t3
/addmush P1 m1 f0 t1 3 3
/addthread t2 f1
/connectthread f0 f1
/addspore t2 s1
/addspore t2 s2
/spreadspore m1 t2
/growmush test1 m2 t2
/growthread f1 f2 t3
/showMap
```

- **Elvárt kimenet**

```
t1(type=N; mushroom=m1; neighbours=t2; threads=f0)
t2(type=N; mushroom=m2; neighbours=t1, t3; threads=f1)
t3(type=N; neighbours=t2; threads=f2)
m1(Age=3; Spore=2; tekton=t1)
m2(Age=1; Spore=3; tekton=t2)
f0(life=3; lifeSupport=true; tekton=t1; next=f1)
f1(life=3; lifeSupport=true; tekton=t2; prev=f0; next=f2)
f2(life=3; lifeSupport=true; tekton=t3; prev=f1)
P1(Type=G; Own=m1, m2)
```

8.3 A tesztelést támogató programok tervei

A részletes tesztelési terv egy összetett, JUnit alapú tesztkeretrendszer, amely a projekt tesztkönyvtárában definiált, egyedi forgatókönyveket tartalmazó alkönyvtárak feldolgozására épül. minden ilyen teszteset-könyvtárban egy parancsfájl (commands.txt) tartalmazza a játék logikáját meghívó, a specifikált bemeneti nyelvnek megfelelő parancsokat, amíg az elvárt kimeneti fájl (expected.txt) első sora definiálja az összehasonlítás elvárt eredményét, a többi sora pedig a kimeneti nyelv szerint részletezi a parancsok végrehajtása után várt játék állapotot.

A központi JUnit tesztosztály a **TesztMain**.

- **Attribútumok:**
 - private final String commandFile
 - private final String expectedFile
 - private final GameModel expectedResult
 - private final GameModel actualResult
- **Függvények**
 - private boolean parseModel(GameModel actual, GameModel expected)

A kettő modellt hasonlíta össze és visszatér a összehasonlítás értékével (hamis, ha nem hasonlít, igaz, ha hasonlít)
 - private void interpretCommands(String fileName, GameModel model)

A fileName nevezetű fájlból kiolvassa a parancsokat, amiket lejátszik és a model-be menti a módosításokat. (bemeneti nyelvet írja át modellbe)
 - private boolean translateExpectedTo(String fileName, GameModel model)

A filename nevezetű fájlból kiolvassa az állapotot, amelyet majd visszatérési értékként visszaküld, de előtte a model-be beolvassa a fileban lévő modell állapotát. (a kimeneti nyelvet átírja modellbe)
 - private void runTests()

Minden tesztet lefuttat az adott alkönyvtárban, külön-külön függetlenül egymástól.
 - private void beforeTests()

Minden teszt előtt lefut, és minden visszatérít a tesztek előtti állapotba.

Ha helyes a teszt, akkor kiírja egy új fájlba (results.txt) az adott alkönyvtárba, hogy helyes az adott teszt, vagy ha nem helyes, akkor viszont kiírja a teszt sikereségét, és mellé az aktuális bemenetet és az elvárt bemenetet is.

8.4 Napló

Kezdet	Időtartam	Résznevők	Leírás
2025.04.04. 13:00	2,5 óra	Bene Kattan Yin Ji Yuang Ji Jin Yan	Értekezlet. A feladatok felosztása személyenként

2025.04.05. 15:00	5 óra	Kattan	Tevékenység: 8.1 írása
2025.04.05. 18:00	2,5 óra	Ji Yuang	Tevékenység: 8.1 írása
2025.04.05. 17:00	2 óra	Yin	Tevékenység: 8.2 írása
2025.04.05. 22:00	2 óra	Ji Jin Yan	Tevékenység: 8.1 írása
2025.04.06. 13:00	1 óra	Yin	Tevékenység: Use case javítás megírása
2025.04.07. 12:00	4 óra	Yin	Tevékenység: 8.2 írása
2025.04.07. 15:00	5 óra	Bene	Tevékenység: Interfacek, view, controller tervezet
2025.04.08. 14:00	1.5 óra	Ji Jin Yan	Tevékenység: 8.1 írása
2025.04.08. 16:00	1 óra	Ji Yuang	Tevékenység: 8.1 írása
2025.04.09. 15:00	3 óra	Yin	Tevékenység: Kimeneti, bemeneti nyelv megírása, helyességtörzés
2025.04.09. 18:00	2 óra	Bene	Tevékenység: Játékot segítő classok bevezetése
2025.04.10. 11:00	5 óra	Kattan	Tevékenység: 8.1 lektorásáa, javítása
2025.04.12. 15:00	1,5 óra	Ji Yuang	Tevékenység: 8.1 bővítése
2025.04.13. 15:00	3 óra	Bene	Tevékenység: 8.3 írása
2025.04.13. 12:00	3 óra	Ji Jin Yan	Tevékenység: 8.1 bővítése
2025.04.13. 16:00	1 óra	Ji Yuang	Tevékenység: 8.1 bővítése

10. Prototípus elkészítése

82 – being_chilling

Konzulens:
Tóth Bence

Csapattagok

Yin Jun	TN1KZI	1171276297yin@gmail.com
Bene Máté	K4C4W1	matthew.be04@gmail.com
Kattan Ibrahim Gábor	ZU22JJ	kattanbarhum2@gmail.com
Ji Jin Yan	ZGCKAL	jijinyan0201@gmail.com
Ji Yuang	UI7V2S	yuangji688@gmail.com

2025.04.27.

10. Prototípus beadása

10.1 Fordítási és futtatási útmutató

10.1.1 Fájllista

Fájl neve	Méret	Keletkezés ideje	Tartalom
build.gradle.kts	1.0kb	2025-04-27	build beállítások gradlewhez
build.gradle.ktsbuild.gradle.kts	60.0kb	2025-03-28	gradlew futtatásához szükséges
gradle\wrapper\gradle-wrapper.properties	1.0kb	2025-03-28	gradlew beállításai
gradlew	9.0kb	2025-03-28	gradlew használatához
gradlew.bat	3.0kb	2025-03-28	gradlew használatához
settings.gradle.kts	1.0kb	2025-03-28	gradlew beállításai
src\main\java\com\beingchilling\controller\ControllerComponent.java	30.0kb	2025-04-27	MVC controller component osztály megvalósítása
src\main\java\com\beingchilling\controller\InsectController.java	1.0kb	2025-04-27	Insect controller interface megvalósítása
src\main\java\com\beingchilling\controller\MushroomBodyController.java	1.0kb	2025-04-25	MushroomBody controller interface megvalósítása
src\main\java\com\beingchilling\controller\MushroomThreadController.java	1.0kb	2025-04-25	MushroomThread controller interface megvalósítása
src\main\java\com\beingchilling\controller\SporeController.java	1.0kb	2025-04-25	Spore controller interface megvalósítása
src\main\java\com\beingchilling\controller\TektonController.java	1.0kb	2025-04-25	Tekton controller interface megvalósítása
src\main\java\com\beingchilling\game\BiMap.java	3.0kb	2025-04-25	BiMap egy util class kétirányú HashMap
src\main\java\com\beingchilling\game\GameModel.java	1.0kb	2025-04-25	GameModel a játék összes karaktere ebben szerepel
src\main\java\com\beingchilling\game\Map.java	1.0kb	2025-04-25	Map class megvalósítása
src\main\java\com\beingchilling\Main.java	2.0kb	2025-04-27	Main a program futtatásához szükséges
src\main\java\com\beingchilling\model\CloneSpore.java	1.0kb	2025-04-27	CloneSpore class megvalósítása

src\main\java\com\beingch illing\model\HasteSpore.java	1.0kb	2025-04-27	HasteSpore class megvalósítása
src\main\java\com\beingch illing\model\Insect.java	8.0kb	2025-04-27	Insect class megvalósítása
src\main\java\com\beingch illing\model\InsectSpecies.java	2.0kb	2025-04-27	InsectSpecies class megvalósítása
src\main\java\com\beingch illing\model\LifeSupportTekton.java	2.0kb	2025-04-25	LifeSupportTekton class megvalósítása
src\main\java\com\beingch illing\model\MultiThreadTekton.java	2.0kb	2025-04-25	MultiThreadTekton class megvalósítása
src\main\java\com\beingch illing\model\MushroomBody.java	6.0kb	2025-04-27	MushroomBody class megvalósítása
src\main\java\com\beingch illing\model\MushroomlessTekton.java	2.0kb	2025-04-25	MushroomlessTekton class megvalósítása
src\main\java\com\beingch illing\model\MushroomSpecies.java	2.0kb	2025-04-27	MushroomSpecies class megvalósítása
src\main\java\com\beingch illing\model\MushroomThread.java	7.0kb	2025-04-27	MushroomThread class megvalósítása
src\main\java\com\beingch illing\model\MuteSpore.java	1.0kb	2025-04-27	MuteSpore class megvalósítása
src\main\java\com\beingch illing\model\ParaSpore.java	1.0kb	2025-04-27	ParaSpore class megvalósítása
src\main\java\com\beingch illing\model\SlowSpore.java	1.0kb	2025-04-27	SlowSpore class megvalósítása
src\main\java\com\beingch illing\model\Spore.java	2.0kb	2025-04-27	Spore class megvalósítása
src\main\java\com\beingch illing\model\Tekton.java	10.0kb	2025-04-27	Tekton class megvalósítása
src\main\java\com\beingch illing\model\ThreadAbsorbTekton.java	2.0kb	2025-04-25	ThreadAbsorbTekton class megvalósítása
src\main\java\com\beingch illing\view\InsectView.java	1.0kb	2025-04-25	Insect view interface megvalósítása
src\main\java\com\beingch illing\view\MushroomBodyBod	1.0kb	2025-04-25	MushroomBody view interface megvalósítása

10. Prototípus beadása

being_chilling

yView.java			
src\main\java\com\beingch illing\view\MushroomThre adView.java	1.0kb	2025-04-25	MushroomThread View interface megvalósítása
src\main\java\com\beingch illing\view\SporeView.java	1.0kb	2025-04-25	Spore View interface megvalósítása
src\main\java\com\beingch illing\view\TektonView.jav a	1.0kb	2025-04-25	Tekton View interface megvalósítása
src\main\java\com\beingch illing\view\ViewCompon ent.java	9.0kb	2025-04-27	MVC modell View controller osztályának megvalósítása
src\main\resources\start.txt	3.0kb	2025-04-27	A kezdő pálya felépítése
src\test\java\SzkeletonTest. java	18.0kb	2025-04-26	A teszt class megvalósítása, ez kezeli a txt fájlokat.
src\test\java\TesztMain.jav a	7.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 1\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 1\output.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 10\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 10\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 11\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 11\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 12\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 12\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 13\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 13\output.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).

10. Prototípus beadása

being_chilling

src\test\resources\tests\teszt 14\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 14\output.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 15\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 15\output.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 16\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 16\output.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 17\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 17\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 18\input.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 18\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 19\input.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 19\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 2\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 2\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 20\input.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 20\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).

10. Prototípus beadása

being_chilling

src\test\resources\tests\teszt 21\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 21\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 22\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 22\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 23\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 23\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 24\input.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 24\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 25\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 25\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 26\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 26\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 27\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 27\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 28\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 28\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).

10. Prototípus beadása

being_chilling

src\test\resources\tests\teszt 29\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 29\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 3\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 3\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 30\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 30\output.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 31\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 31\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 32\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 32\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 33\input.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 33\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 34\input.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 34\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 35\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 35\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).

src\test\resources\tests\teszt 36\input.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 36\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 4\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 4\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 5\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 5\output.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 6\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 6\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 7\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 7\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 8\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 8\output.txt	1.0kb	2025-04-27	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).
src\test\resources\tests\teszt 9\input.txt	1.0kb	2025-04-26	Adott teszthez szükséges fájlok (input=bemenet, output=kimenet).

10.1.2 Fordítás

A program lefordításához a gradlew.bat jar parancs futtatása (cmd-vel) szükséges a project mappában. Ez létrehozza a build/libs mappába a jar file-t.

A tesztek futtatásához gradlew.bat test parancs futtatása (cmd-vel) szükséges. Az eredményt a build/resources/test/tests mappában található összegezve és egyesével is.

10.1.3 Futtatás

A futtatás a gradlew.bat run parancs futtatásával lehetséges, vagy az előző résznél kapott .jar fájl futtatásával (vagyis java -jar *filename.jar* parancs segítségével lehet a fentebb említett mappában (build/libs)).

10.2 Tesztek jegyzőkönyvei

10.2.1 Rovar move success

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz output file
Változtatások	output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.2 Rovar move unsuc (no fonal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) move failed eltávolítása

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.3 Rovar move unsec (no enough speed)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) move failed eltávolítása lifesupport=0 helyett false

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.4 Rovar move unsec (para)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) move failed eltávolítása lifesupport=0 helyett false

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.5 Rovar eat spore success

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.

10.2.6 Rovar eat spore unsec (no spore)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	eat failed eltávolítása

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.7 Rovar eat spore unsec (para)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) s1 nutrient 5-re átírása spore átírása spores-ra t1-nél

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.8 Rovar cut success

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) insect rossz tektonon -> t2-re átirva lifesupport=0 átirása false-ra

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.9 Rovar cut unsec(mute)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.10 Rovar cut unsec(para)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.11 Rovar cut unsec(no thread)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) insect t2-re átirása

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.12 Rovar cut unsec(already cut)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) insect t2-re átírása lifesupport=0 átírása false-ra

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.13 Eat haste spore

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.

10.2.14 Eat slow spore

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.

10.2.15 Eat mute spore

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.

10.2.16 Eat para spore

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.

10.2.17 Eat clone spore

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail

Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.18 Spread spore success

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	Parancs átírása, /spreadspore +1 paraméter (N) output fornázása (nem volt egységes az elvárt formátummal) t2 spores=m1_spore3 lett threads=f0 hiányzott tekton= hiányzott

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.19 Spread spore unsec(no spore)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancsok
Változtatások	Parancs átírása, /addmush-nál 0-át 1-re és még 1 spreadspore hozzáadás /spreadspore-nál +1 paraméter N hozzáadása output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.20 Spread spore unsec(too far)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	Parancs átírása, /spreadspore-nál +1 paraméter N hozzáadása output fornázása (nem volt egységes az elvárt formátummal) tekton= beillesztése t1 elő

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.21 Grow mushroom success

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) addthred -> addthread

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.22 Grow mushroom unsec(no enough spore)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) addthred -> addthread

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.23 Grow mushroom unsec(no fonal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.24 Grow mushroom unsec(mushroomless)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail

Lehetséges hibaok	Rossz parancs
Változtatások	Parancs átírása, /addthred helyett /addthread

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.25 Grow fonal success

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.26 Grow fonal unsec(occupied)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.27 Grow fonal random

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal), t4 neighbours t2 hiányzott, f2 átnevezése f1.ran

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.28 Tekton Absorb

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.

Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) L, T helyett LS és TA f3 lifesupport = true

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.29 Tekton Absorb*3

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) L, T helyett LS és TA f3 lifesupport = true

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.30 Tekton break success

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) 0 helyett false

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.31 Tekton break unsec(gomba)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.32 Tekton break unsec(rovar)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal) speed 3 helyett 2

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.33 Absorb rovar success

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	Parancs átírása, /absorbinsect-nél f2-öt f1-re output fornázása (nem volt egységes az elvárt formátummal) age=0 m2-nél és spore=5

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.34 Absorb rovar unsec(not para)

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	Parancs átírása, /absorbinsect-nél f2-öt f1-re output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.35 Rovar összetett teszt

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	output fornázása (nem volt egységes az elvárt formátummal)

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.2.36 Gomba összetett teszt

Tesztelő neve	Bene
Teszt időpontja	2025.04.25.
Teszt eredménye	fail
Lehetséges hibaok	Rossz parancs
Változtatások	Parancs átírása, /growmush test1-et P1-re és /spreadspore + paraméter N output fornázása (nem volt egységes az elvárt formátummal) age=0 m2-nél és spore=5

Tesztelő neve	Bene
Teszt időpontja	2025.04.26.

10.3 Értékelés

Tag neve	Tag neptun	Munka százalékban
Ji Jin Yan	ZGCKAL	20%
Yin Jun	TN1KZI	20%
Kattan Ibrahim Gábor	ZU22JJ	20%
Ji Yuang	UI7V2S	20%
Bene Máté	K4C4W1	20%

10.4 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2025.04.24. 14:00	2 óra	Bene	Tevékenység: Feladatok kiosztása
2025.04.24. 10:00	4 óra	Yin	Tevékenység: Teszteket kezelő class megírása
2025.04.25. 15:00	5 óra	Kattan	Tevékenység: ControlComponent osztály és ViewComponent osztály megvalósítása
2025.04.25. 09:00	3 óra	Ji Jin Yan	Tevékenység: Modell-beli osztályok megvalósítása/befej ezése
2025.04.26. 12:00	2 óra	Ji Yuang	Tevékenység: Tesztek megírása
2025.04.26. 13:00	2 óra	Yin	Tevékenység: Modell javítása
2025.04.26. 15:00	3 óra	Ji Jin Yan	Tevékenység: ControllComponent javítása
2025.04.26. 15:00	4 óra	Bene	Tevékenység: ControllComponent és ViewComponentjaví tása
2025.04.27. 15:00	3 óra	Bene	Tevékenység: Tesztek javítása
2025.04.27. 08:00	4 óra	Yin	Tevékenység: Tesztek javítása
2025.04.27. 15:00	3 óra	Bene	Tevékenység: Dokumentum megírása

11. Grafikus változat tervezés

82 – being_chilling

Konzulens:
Tóth Bence

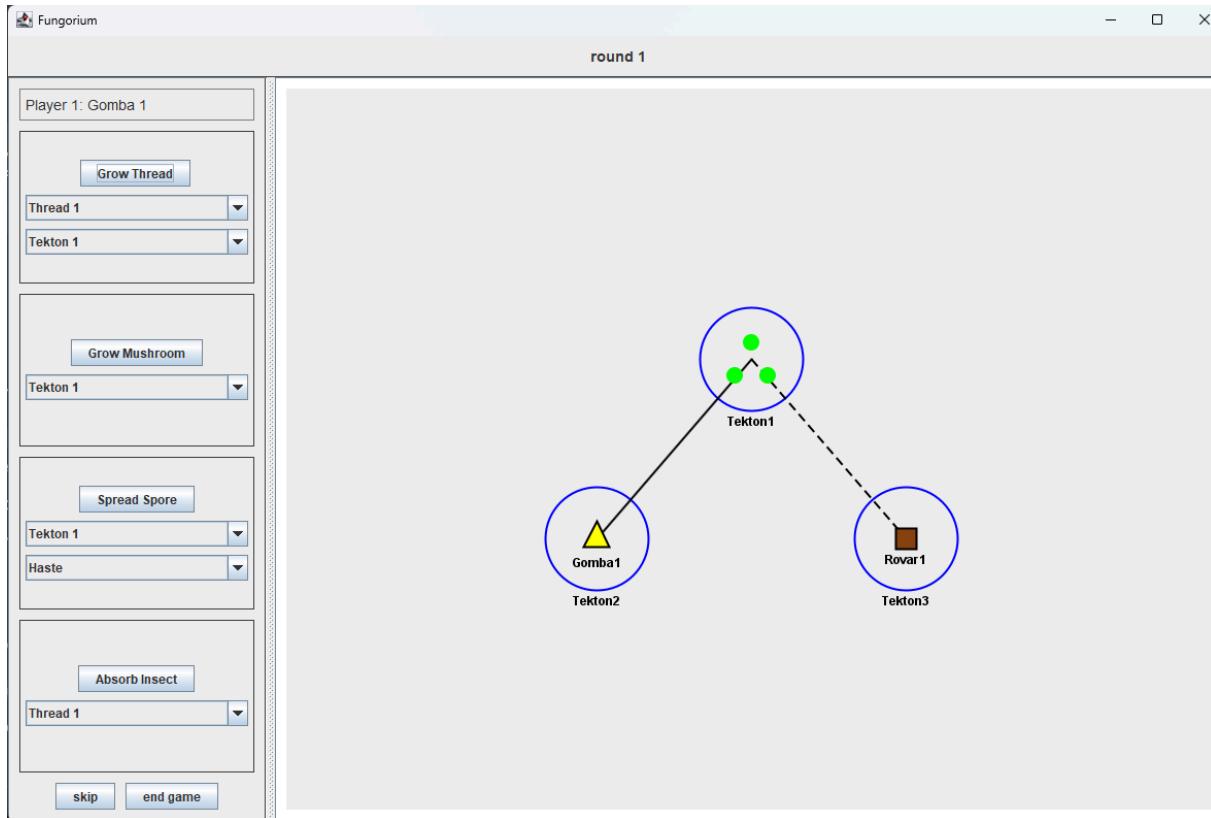
Csapattagok

Yin Jun	TN1KZI	1171276297yin@gmail.com
Bene Máté	K4C4W1	matthew.be04@gmail.com
Kattan Ibrahim Gábor	ZU22JJ	kattanbarhum2@gmail.com
Ji Jin Yan	ZGCKAL	jijinyan0201@gmail.com
Ji Yuang	UI7V2S	yuangji688@gmail.com

2025.05.04.

11. Grafikus felület specifikációja

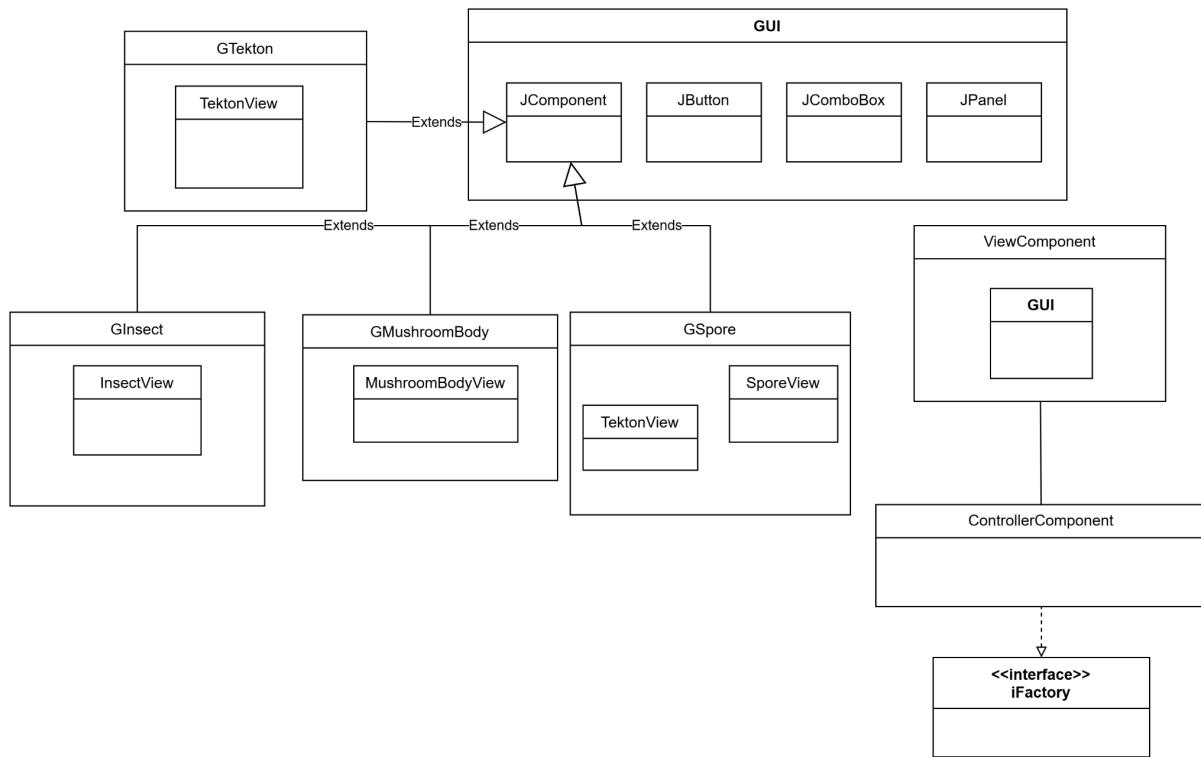
11.1 A grafikus interfész



Felépítés:

Bal oldalon látható a vezérlőpanel, jobb oldalon pedig a játéktér grafikusan ábrázolva. A kék töltetlen körök tektonokat, a zöld kitöltött körök spórákat, a sárga háromszögek gombákat, a barna négyzetek pedig rovarokat ábrázolnak. A játék kör alapú, azt, hogy hányadik körön vagyunk azt a játéktér fölött látjuk, bal oldalon alul pedig át is lehet ugrani egy kört, ha nem szeretnénk semmit csinálni. A vezérlőpult tetején látható az, hogy melyik játékos melyik objektuma van kiválasztva. Alatta pedig a többi, csapattól függő funkciók elvégzése lehetséges. Ki lehet választani, hogy melyik tektonra akarunk növeszteni melyik fonalból, melyik tektonon akarunk gombát növeszteni, melyik tektonra akarunk minden spórát rakni, vagy pedig azt, hogy ha fel akarunk szívni egy rovart.

11.2 A grafikus rendszer architektúrája



11.2.1 A felület működési elve

A grafikai modell egy kevert modellt valósít meg, ahol a megjelenést a GUI, és a ViewComponent valósít meg.

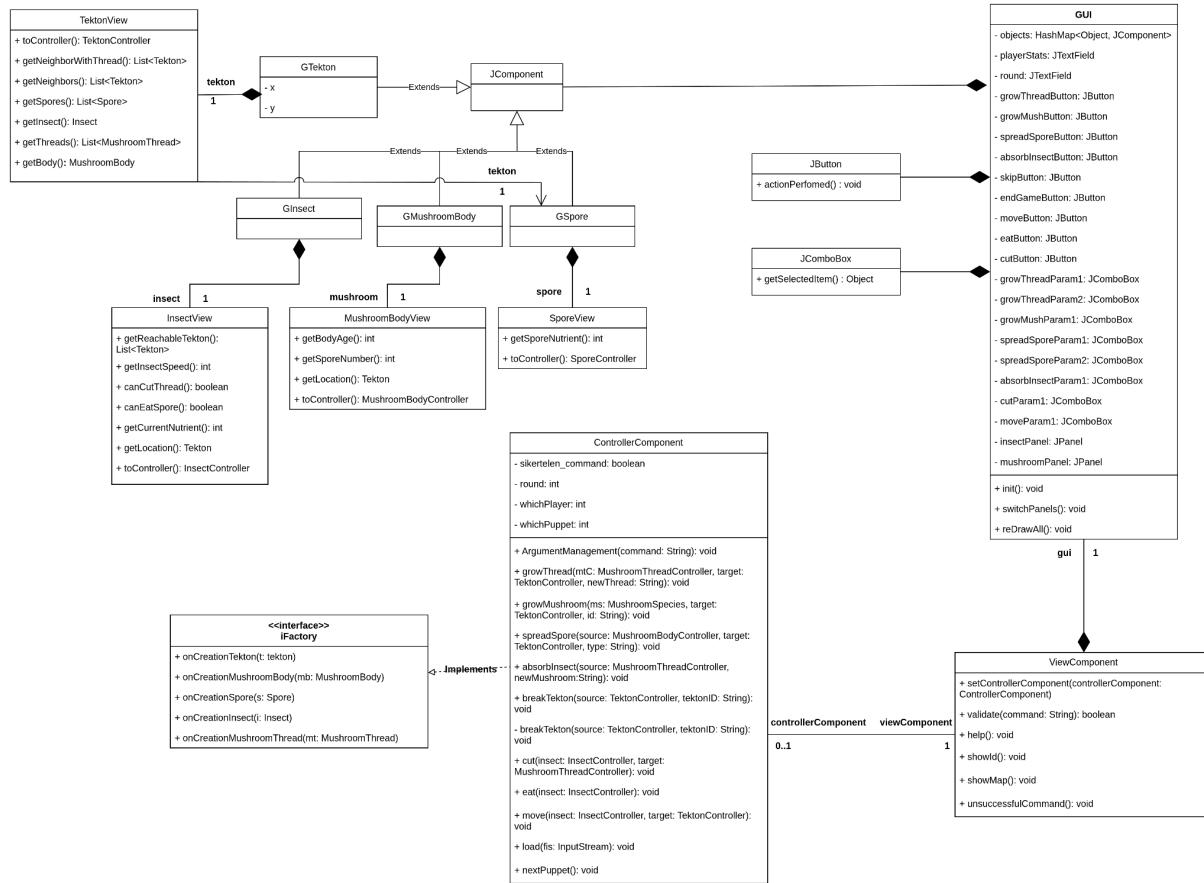
A nézet központi eleme a GUI osztály, amely a fő ablakot és a benne lévő vezérlőket (gombok, panelek) tartalmazza. A ViewComponent aggregálja és megjeleníti az egyes modellelemekhez tartozó konkrét nézeteket (MushroomView, InsectView, SporeView). Ezek a specifikus nézetek kapcsolják össze a modellbeli objektumokat a képernyőn látható grafikus elemekkel.

A ControllerComponent a központi vezérlő, ez az osztály felelős a játék logikájának futtatásáért, a felhasználói parancsok értelmezéséért és a modell állapotának frissítéséért.

A modellben történt változás után a vezérlő értesíti a nézetet a frissítés szükségességéről, valószínűleg a reDrawAll metódusának meghívásán keresztül, ami push elven működik.

Az reDrawAll hatására elindul a felület újraraajzolási folyamata. Ebben a fázisban a ViewComponent és az általa tartalmazott specifikus nézetek (MushroomView stb.) lekérdezik a hozzájuk tartozó modell objektumok aktuális állapotát (pl. pozíció), hogy a komponensek a legfrissebb adatok alapján tudják kirajzolni magukat. Ez a rajzolás közbeni adatlekérdezés pull művelet.

11.2.2 A felület osztály-struktúrája



11.3 A grafikus objektumok felsorolása

11.3.1 iFactory

- **Felelősség**

objektumok létrehozásának folyamatát absztrahálja

- **Ősosztályok**

-

- **Interfészek**

-

- **Attribútumok**

-

- **Metódusok**

- + **onCreation Tekton(t: tekton):**
A tekton-t berak egy tektonList hashmapba.
- + **onCreationMushroomBody(mb: MushroomBody) :**
A Mushroombody-t berak egy gombasz hashmapba.

- + **onCreationSpore(s: Spore)** :
A Spore-t hozzá ad egy tektonhoz
- + **onCreationInsect(i: Insect)** :
Az Insect-et berak egy ravarasz hashmapba.
- + **onCreationMushroomThread(mt: MushroomThread)**:
Egy MushroomThread-et hozzá ad tektonhoz

11.3.2 InsectView

- **Felelősség**

egy konkrét Insect objektum vizuális megjelenítéséért felelős a felhasználói felületen

- **Ősosztályok**

-

- **Interfészek**

-

- **Attribútumok**

-

- **Metódusok**

- +List<Tekton> getReachable Tekton():
Az általa elérhető tekton adja vissza.
- +int getInsectSpeed():
rovar sebességét adja vissza.
- +boolean canCutThread():
tud-e vágni.
- +boolean canEatSpore():
tud-e spórát enni.
- +int getCurrentNutrient():
visszaadaj hogya rovarnak mennyi tápanyaga van.
- +Tekton getLocation():
A tekton kapja meg ahol a rovar van.
- +InsectController toController():
átváltja magát InsectController-é.

11.3.3 MushroomBodyView

- **Felelősség**

egy konkrét gombatest objektum vizuális megjelenítéséért felelős a felhasználói felületen

- **Ősosztályok**

-

- **Interfész**

-

- **Attribútumok**

-

- **Metódusok**

- + int **getBodyAge()**:
gombatest életkorát adja vissza
- + int **getSpore Number()**:
gombatestben lévő spórák számát ad vissza
- +Tekton **getLocation()**:
tektonat ad vissza, amin a gombatest van
- + **MushroomBody Controller toController()**:
átkonvertálja magát MushroomBody Controller-é

11.3.4 SporeView

- **Felelősség**

egy konkrét spóra objektum vizuális megjelenítéséért felelős a felhasználói felületen

- **Ősosztályok**

-

- **Interfészek**

-

- **Attribútumok**

-

- **Metódusok**

- + int **getSporeNutrient()**:
a spórának a tápanyag tartalmát adja vissza
- + **SporeController toController()**:
átkonvertálja magát SporeController-é

11.3.5 ControllerComponent (változtatások)

- **Attribútumok**

- -boolean **sikertelen_command**: Jelzi, hogy a megadott parancs sikeresen lefuttatható, vagy sem.
- -int **round**: Leírja, hogy hányadik körnél tartunk
- -int **whichPlayer**: Leírja, hányadik játékos köre van éppen.
- -int **whichPuppet**: Leírja, hogy a játékosnak éppen melyik bábúja jön.

- **Metódusok**

- **void nextPuppet()**: Igazgatja a játékosokat és a bábukat, úgy, hogy sorra kerüljenek
- **void ArgumentManagement(String command)**: ellenőrzi, hogy szintaktikailag helyes parancsot kaptunk.

11. Grafikus felület specifikációja

being_chilling

```

method ArgumentManagement(command: String)
    if viewComponent.validate(command) is false then
        viewComponent.unsuccessfulCommand
        return
    end if
    create new String[] words
    split command by " " into words
    switch(words[0])
        case /addplayer
            if words[2] is R then
                add new InsectSpecies to GameModel.gameObjects with words[1] ID
            end if
            else
                add new MushroomSpecies to GameModel.gameObjects with words[1] ID
        case /addtekton
            if words.length is 2 then
                create new Tekton t
                add t to GameModel.gameObjects with words[1] ID
                add t to GameModel.map.tektonList with words[1] ID
            end if
            else
                switch(words[2])
                    case N
                        create new Tekton t
                        add t to GameModel.gameObjects with words[1] ID
                        add t to GameModel.map.tektonList with words[1] ID
                    case LS
                        create new LifeSupportTekton t
                        add t to GameModel.gameObjects with words[1] ID
                        add t to GameModel.map.tektonList with words[1] ID
                    case TA
                        create new ThreadAbsorbTekton t
                        add t to GameModel.gameObjects with words[1] ID
                        add t to GameModel.map.tektonList with words[1] ID
                    case MT
                        create new MultiThreadTekton t
                        add t to GameModel.gameObjects with words[1] ID
                        add t to GameModel.map.tektonList with words[1] ID
                    case MS
                        create new MushroomlessTekton t
                        add t to GameModel.gameObjects with words[1] ID
                        add t to GameModel.map.tektonList with words[1] ID
                    case /addinsect
                        create new Insect newInsect
                        if words.length is 5 then
                            switch(words[4])
                                case S
                                    newInsect.slowEffect()
                                case H
                                    newInsect.hasteEffect()
                                case M
                                    newInsect.muteEffect()
                                case P
                                    newInsect.paraEffect()
                                case C
                                    newInsect.cloneEffect()
                            end if
                            add newInsect to GameModel.gameObjects with words[2] ID
                            add newInsect to InsectSpecies with words[1] ID
                            add newInsect to Tekton with words[3] ID
                            set newInsect location to Tekton with words[3] ID
                        case /addmush
                            create new MushroomBody with location of Tekton with words[4] ID
                            if words.length is greater than 5 then
                                newMushroom.setBodyAge(words[5])
                                if words.length is greater than 6 then
                                    newMushroom.setBodyAge(words[6])
                                end if
                            end if
                            else
                                newMushroom.setBodyAge(3)
                                newMushroom.setSporeNumber(3)
                            add newMushroom to GameModel.gameObjects with words[2] ID
                        end if
                    end if
                end if
            end if
        end if
    end if
end if

```

```

set newMushroom.owner to MushroomSpecies with words[1] ID
add newMushroom to MushroomSpecies with words[1] ID
add new MushroomThread to tekton with words[4] ID
case /addspore
    create new int nutr
    if words.length is greater than 4 then
        nutr = words[4]
    end if
    else
        nutr = 5
    if words.length is greater than 3 then
        spore = switch(words[3])
        case S
            create new SlowSpore with nutrient nutr
        case H
            create new HasteSpore with nutrient nutr
        case M
            create new MuteSpore with nutrient nutr
        case P
            create new ParaSpore with nutrient nutr
        case C
            create new CloneSpore with nutrient nutr
    end if
    add spore to GameModel.gameObjects with ID words[2]
    add spore to tekton with ID words[1]
case /addthread
    create new MushroomThread newThread2
    if adding newThread2 to tekton with words[1] ID returns true then
        add newThread2 to GameModel.gameObjects
    end if
case /setneighbour
    add Tekton with words[1] ID to Tekton with words[2] ID neighbours
    add Tekton with words[2] ID to Tekton with words[1] ID neighbours
case /connectthread
    call addThread() for MushroomThread with words[1] ID with MushroomThread with words[2] ID as parameter
case /showid
    viewComponent.showId()
case /help
    viewComponent.help()
case /absorb
    if words.length is greater than 1 then
        for each item in words after the first index
            call absorb for tekton with ID item
        end loop
    end if
    else
        call absorb for every tekton in GameModel.map.tektonList
case /showmap
    viewComponent.showMap()
case /break
    call breakTekton on tekton with ID words[1], with new ID words[2]
    if words.length is greater than 3 then
        split words[3] by ", " into neighbours[]
        create new list<Tekton> newAdd
        for each str in neighbours[]
            add tekton with ID str to newAdd
        end loop
        call updateNeighbor() on tekton with ID words[1] using newAdd and empty list
    end if
case /growththread
    call growThread(source, target, words[2]) where
        source = MushroomThread with ID words[1]
        target = Tekton with ID words[3],
case /growmush
    call growMushroom(ms, target, words[2]) where
        ms = MushroomSpecies with words[1] ID
        target = Tekton with words[3] ID
case /spreadspore
    if words.length is 4 then
        type = words[3]
    else
        type = ""
    end if
    call spreadSpore(source, target, sporeType) where
        source = MushroomBody with ID words[1]
        target = Tekton with ID words[2]

```

```

sporeType = type
case /absorbinsect
    call absorbInsect(MushroomThread with words[1] ID, words[2])
case /cut
    if value of words[2] ID in GameModel.gameObjects is null then
        break
    end if
    call cut(Insect with words[1] ID, MushroomThread with words[2] ID)
case /eat
    call eat(Insect with words[1] ID)
case /move
    call move(Insect with words[1] ID, Tekton with words[2] ID)
case /random
    GameModel.randomSwitch = !GameModel.randomSwitch
case /load
    call load with stream from resource file "start.txt"
end

```

11.3.6 GUI

- **Felelősség**

az osztály felelős a teljes grafikus felhasználói felület létrehozásáért és kezeléséért, és a feladata a játéktér grafikus megjelenítése és a felhasználói parancsok fogadása.

- **Ősosztályok**

-

- **Interfészek**

-

- **Attribútumok**

- - **HashMap<Object, JComponent> objects:** Tárolja a JComponenteket egy HashMapben és objektumokhoz rendelve a könnyebb kezelhetőség érdekében.
- - **JTextField playerStats:** Szövegmező, amely a játékosok aktuális statisztikáit (pl. melyik player + melyik bábuja) jeleníti meg.
- - **JTextField round:** Szövegmező, amely az aktuális játék körének sorszámát mutatja.
- - **JButton growThreadButton:** Gomb a fonál növesztése akció kiváltására.
- - **JButton growMushButton:** Gomb a gomba növesztése akció kiváltására.
- - **JButton spreadSporeButton:** Gomb a spóra szórása akció kiváltására.
- - **JButton absorbInsectButton:** Gomb a rovar elnyelése akció kiváltására.
- - **JButton skipButton:** Gomb az aktuális kör kihagyására.
- - **JButton endGameButton:** Gomb a játék idő előtti befejezésére.
- - **JButton moveButton:** Gomb a rovar mozgatás akció kiváltására.
- - **JButton eatButton:** Gomb a rovar evés akció kiváltására.
- - **JButton cutButton:** Gomb a rovar vágás akció kiváltására.
- - **JComboBox growThreadParam1:** Legördülő lista a fonál növesztése parancs első paraméterének kiválasztására.
- - **JComboBox growThreadParam2:** Legördülő lista a fonál növesztése parancs második paraméterének kiválasztására.
- - **JComboBox growMushParam1:** Legördülő lista a gomba növesztése parancs első paraméterének kiválasztására.
- - **JComboBox spreadSporeParam1:** Legördülő lista a spóra szórása parancs első paraméterének kiválasztására.

- - **JComboBox spreadSporeParam2:** Legördülő lista a spóra szórása parancs második paraméterének kiválasztására.
 - - **JComboBox absorbInsectParam1:** Legördülő lista a rovar elnyelése parancs első paraméterének kiválasztására.
 - - **JComboBox cutParam1:** Legördülő lista a rovar vágás parancs első paraméterének kiválasztására.
 - - **JComboBox moveParam1:** Legördülő lista a rovar mozgatás parancs első paraméterének kiválasztására.
 - - **JPanel insectPanel:** Panel, amely a rovarokkal kapcsolatos vezérlőket és információkat tartalmazza.
 - - **JPanel mushroomPanel:** Panel, amely a gombákkal kapcsolatos vezérlőket és információkat tartalmazza.
- **Metódusok**
 - + **void init():** Inicializálja a teljes grafikus felhasználói felületet, létrehozza és konfigurálja az ablakokat, paneleket, gombokat és egyéb vezérlőket.
 - + **void switchPanels():** Lehetővé teszi a panelek (pl. rovar panel, gomba panel) közötti váltást a GUI-ban.
 - + **void reDrawAll():** Frissíti és újrarájzolja a teljes grafikus felületet

```

Method init()
create new JFrame mainWindow
set mainWindow title to "Fungorium"
set mainWindow default close operation to EXIT_ON_CLOSE

create new JPanel controlPanel
create new JPanel gameFieldPanel

set mainWindow layout to BorderLayout
add controlPanel to mainWindow at BorderLayout.WEST
add gameFieldPanel to mainWindow at BorderLayout.CENTER

create new JTextField playerStats with initial text "Player: -, Puppet: -"
create new JTextField round with initial text "Round: 0"
create new JButton growThreadButton with text "Grow Thread"
create new JButton growMushButton with text "Grow Mushroom"
create new JButton spreadSporeButton with text "Spread Spore"
create new JButton absorbInsectButton with text "Absorb Insect"
create new JButton skipButton with text "Skip Turn"
create new JButton endGameButton with text "End Game"
create new JButton moveButton with text "Move Insect"
create new JButton eatButton with text "Eat Spore"
create new JButton cutButton with text "Cut Thread"
create new JComboBox growThreadParam1 with options from GameModel
create new JComboBox growThreadParam2 with options from GameModel
create new JComboBox growMushParam1 with options from GameModel
create new JComboBox spreadSporeParam1 with options from GameModel
create new JComboBox spreadSporeParam2 with options from GameModel
create new JComboBox absorbInsectParam1 with options from GameModel
create new JComboBox cutParam1 with options from GameModel
create new JComboBox moveParam1 with options from GameModel
create new JPanel insectPanel
create new JPanel mushroomPanel

add playerStats to controlPanel
add round to controlPanel
add growThreadButton to controlPanel
add growMushButton to controlPanel
add spreadSporeButton to controlPanel
add absorbInsectButton to controlPanel
add skipButton to controlPanel
add endGameButton to controlPanel
add moveButton to controlPanel
add eatButton to controlPanel
add cutButton to controlPanel
add growThreadParam1 to controlPanel
add growThreadParam2 to controlPanel
add growMushParam1 to controlPanel
add spreadSporeParam1 to controlPanel
add spreadSporeParam2 to controlPanel
add absorbInsectParam1 to controlPanel
add cutParam1 to controlPanel
add moveParam1 to controlPanel
add insectPanel to controlPanel
add mushroomPanel to controlPanel

create new HashMap objects (Object -> JComponent)
set mainWindow visibility to true
End Method

```

```

Method reDrawAll()
clear gameFieldPanel
viewComponent : ViewComponent = getViewComponent()
for each entry in objects HashMap
object : Object = entry.key
jComponent : JComponent = entry.value
if jComponent is GIInsect
insectView : InsectView = jComponent.getInsectView()
set jComponent position to object.getPosition()
add jComponent to gameFieldPanel
else if jComponent is GMushroomBody
mushroomBodyView : MushroomBodyView = jComponent.getMushroomBodyView()
set jComponent position to object.getPosition()
add jComponent to gameFieldPanel
else if jComponent is GSporer
sporeView : SporeView = jComponent.getSporeView()
set jComponent position to object.getPosition()
add jComponent to gameFieldPanel
else if jComponent is GTekton
tektonView : TektonView = jComponent.getTektonView()
set jComponent position to object.getPosition()
add jComponent to gameFieldPanel
end for each
controllerComponent : ControllerComponent = viewComponent.getControllerComponent()
roundData : String = "Round: " + controllerComponent.round
playerStatsData : String = "Player: " + controllerComponent.whichPlayer + ", Puppet: " + controllerComponent.whichPuppet
update playerStats text with playerStatsData
update round text with roundData
repaint gameFieldPanel
End Method

```

11.3.7 ViewComponent

- **Felelősség**

Az osztály felelős a játék állapotával kapcsolatos információk megjelenítéséért a felhasználó felé, valamint a felhasználói parancsok alapvető ellenőrzéséért

- **Ősosztályok**

-

- **Interfészek**

-

- **Attribútumok**

- - GUI gui: a GUI osztályhoz való hozzáférést biztosítja

- **Metódusok**

- **+ void setControllerComponent(controllerComponent: ControllerComponent):**
beállítja a view-hoz a controller-t (hozzáférést ad a view a controllerhez).
- **+ boolean validate(command: String):** A beérkező commandot értelmezi és meghívja a controller megfelelő függvényét a commandtól függően.
- **void help():** A playernek szükséges parancsokat írja ki.
- **+ void showId():** Az összes id-t kiírja a konzolra
- **+ void showMap():** A mappot kiírja a konzolra.
- **+ void unsuccessfulCommand():** Ha nem sikeres a parancs, akkor kiírja a kimenetet.

11.3.8 GInsect

- **Felelősség**

vizuálisan megjelenítsen egy Insect (rovar) objektumot a felhasználói felületen

- **Ősosztályok**

JComponent

- **Interfészek**

-

- **Attribútumok**

- -**InsectView Insect:** egy konkrét Insect objektum vizuális megjelenítéséért felelős a felhasználói felületen

- **Metódusok**

11.3.9 GMushroomBody

- **Felelősség**

vizuálisan megjelenítsen egy Mushroom(Gomba) objektumot a felhasználói felületen

- **Ősosztályok**

JComponent

- **Interfészek**

-

- **Attribútumok**

- -**MushroomBodyView Mushroombody:** egy konkrét Mushroom objektum vizuális megjelenítéséért felelős a felhasználói felületen

- **Metódusok**

11.3.10 GSpore

- **Felelősség**

vizuálisan megjelenítsen egy Spore objektumot a felhasználói felületen

- **Ősosztályok**

JComponent

- **Interfészek**

-

- **Attribútumok**

-**SporeView Spore:** egy konkrét Spore objektum vizuális megjelenítéséért felelős a felhasználói felületen

- **Metódusok**

11.3.11 GTekton

- **Felelősség**

vizuálisan megjelenítsen egy Tekton objektumot a felhasználói felületen

- **Ősosztályok**

JComponent

- **Interfészek**

-

- **Attribútumok**

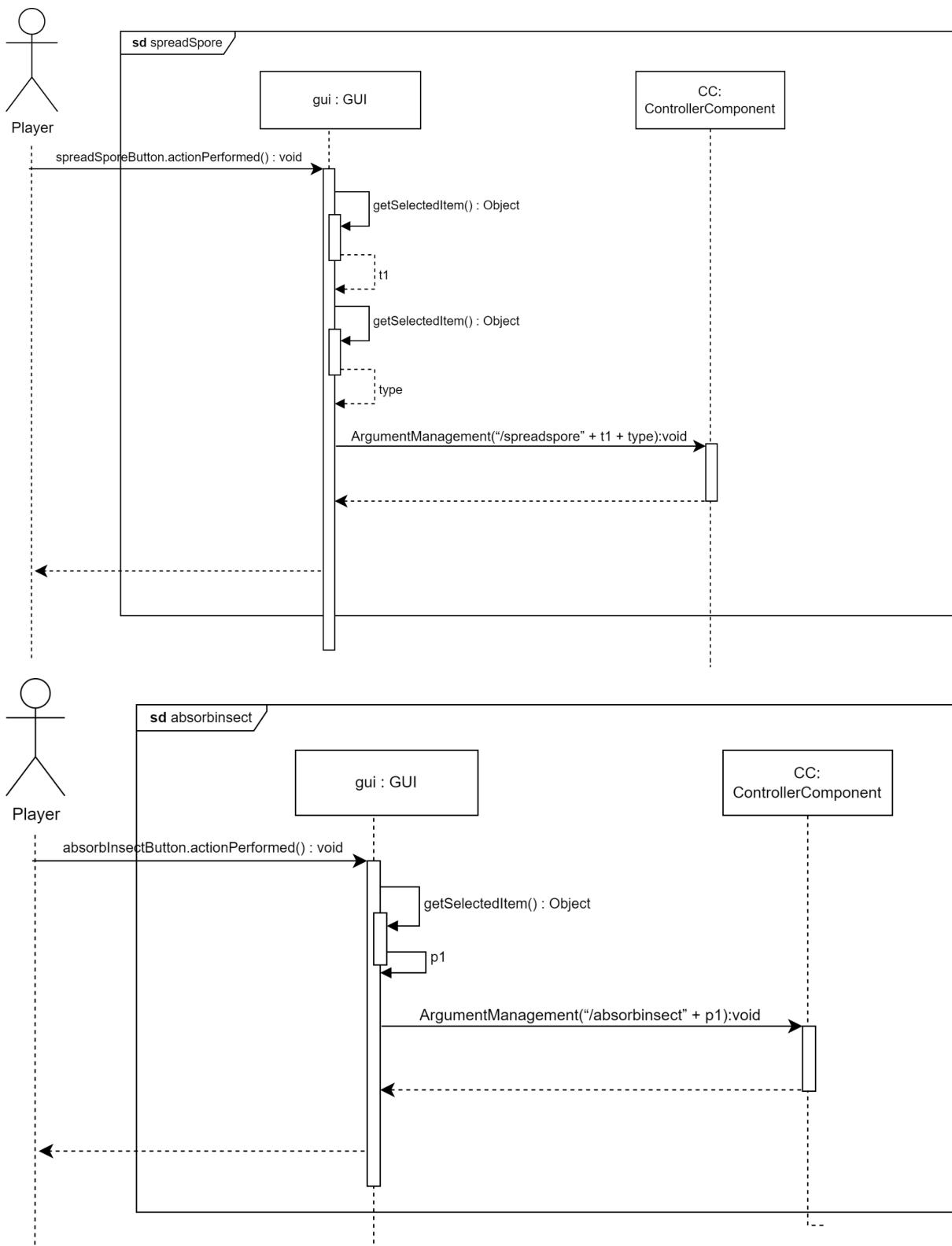
-**TektonView Tekton:** egy konkrét Tekton objektum vizuális megjelenítéséért felelős a felhasználói felületen

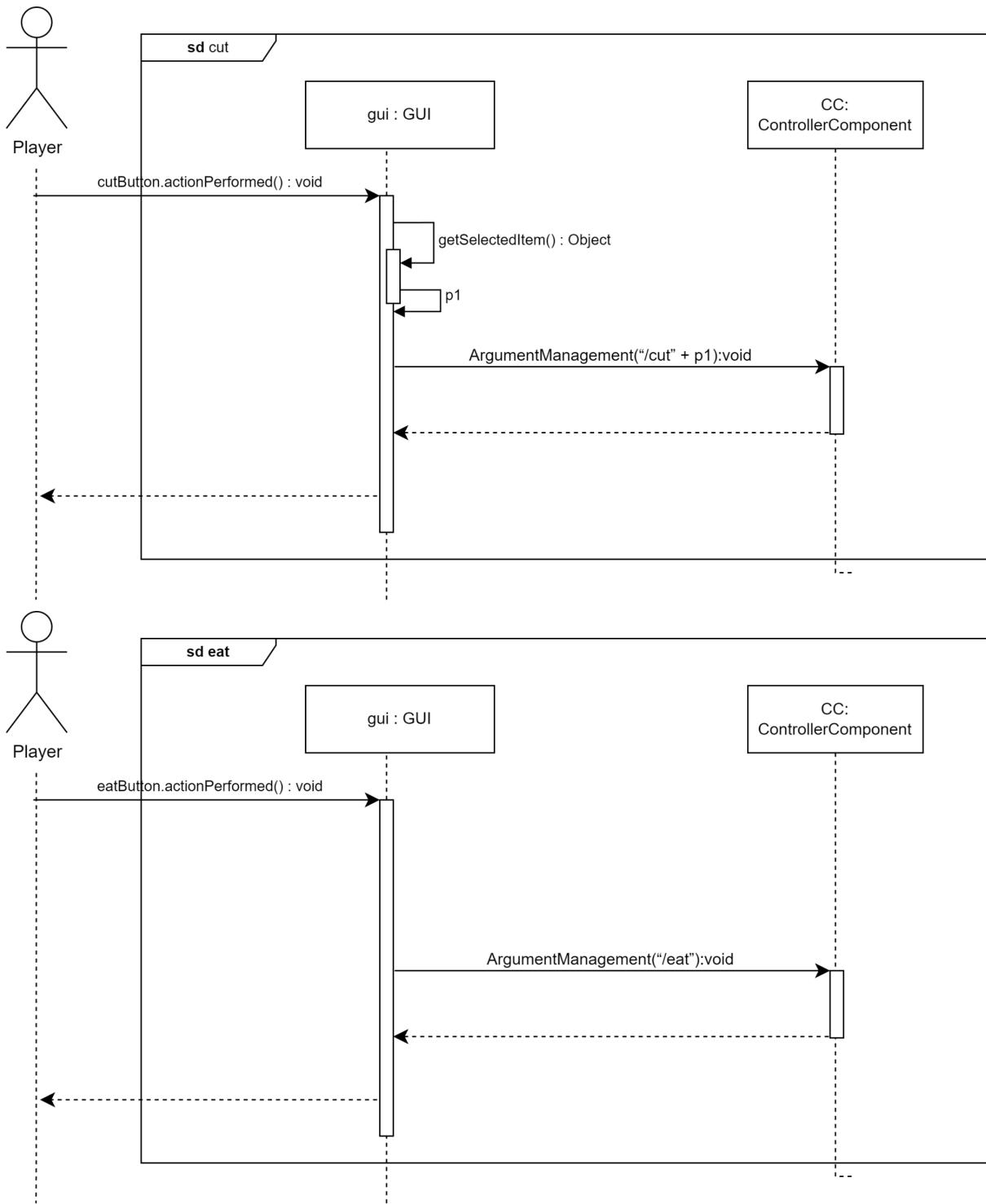
-int x: Tekton x koordinátáját írja le.

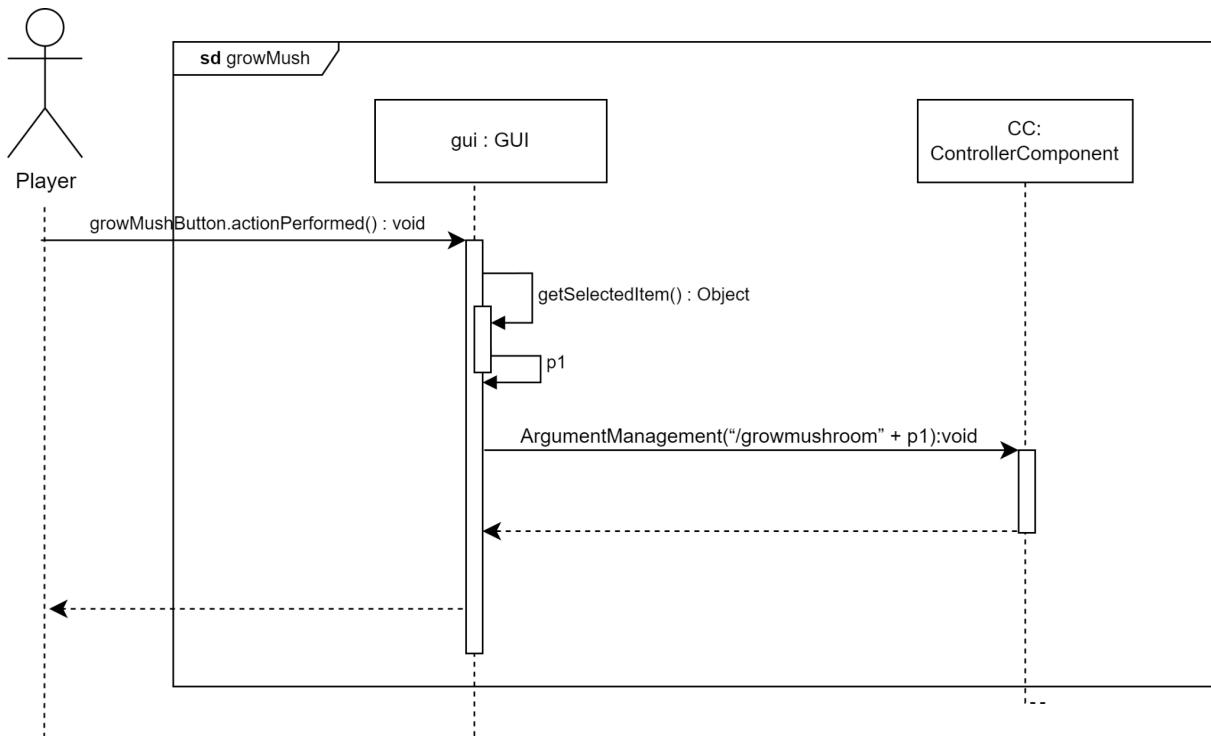
-int Y: Tekton y koordinátáját írja le

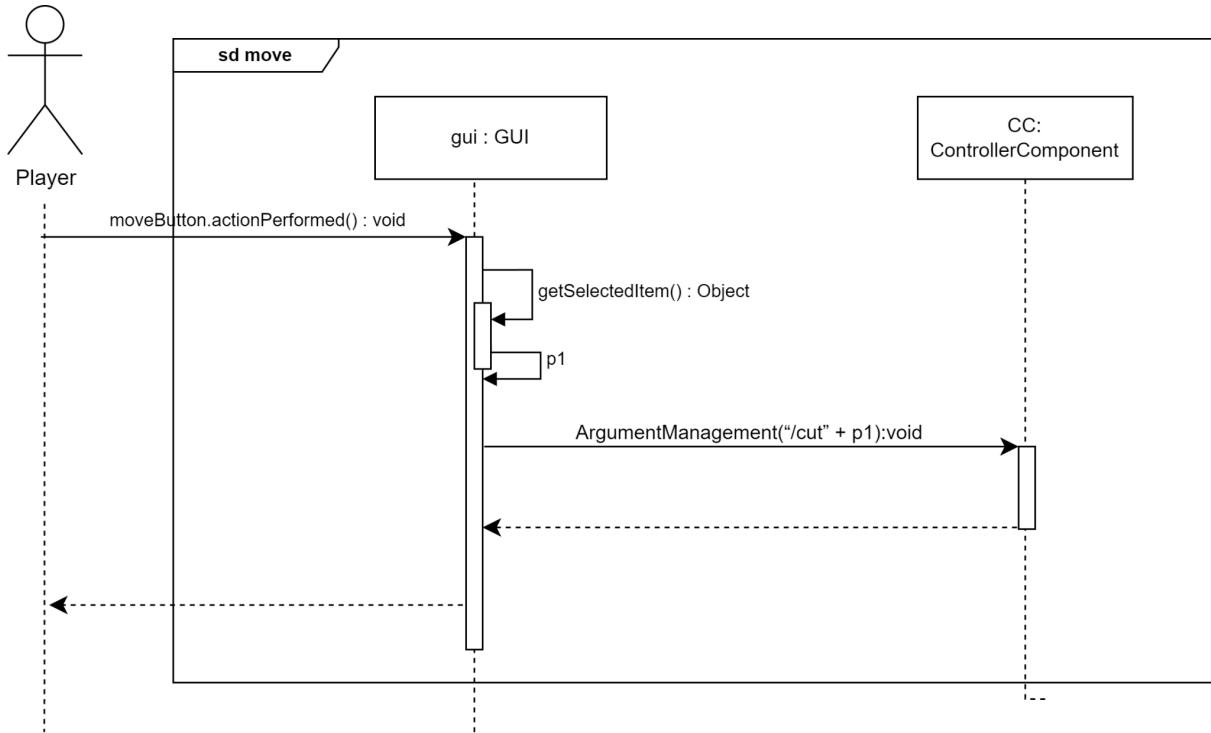
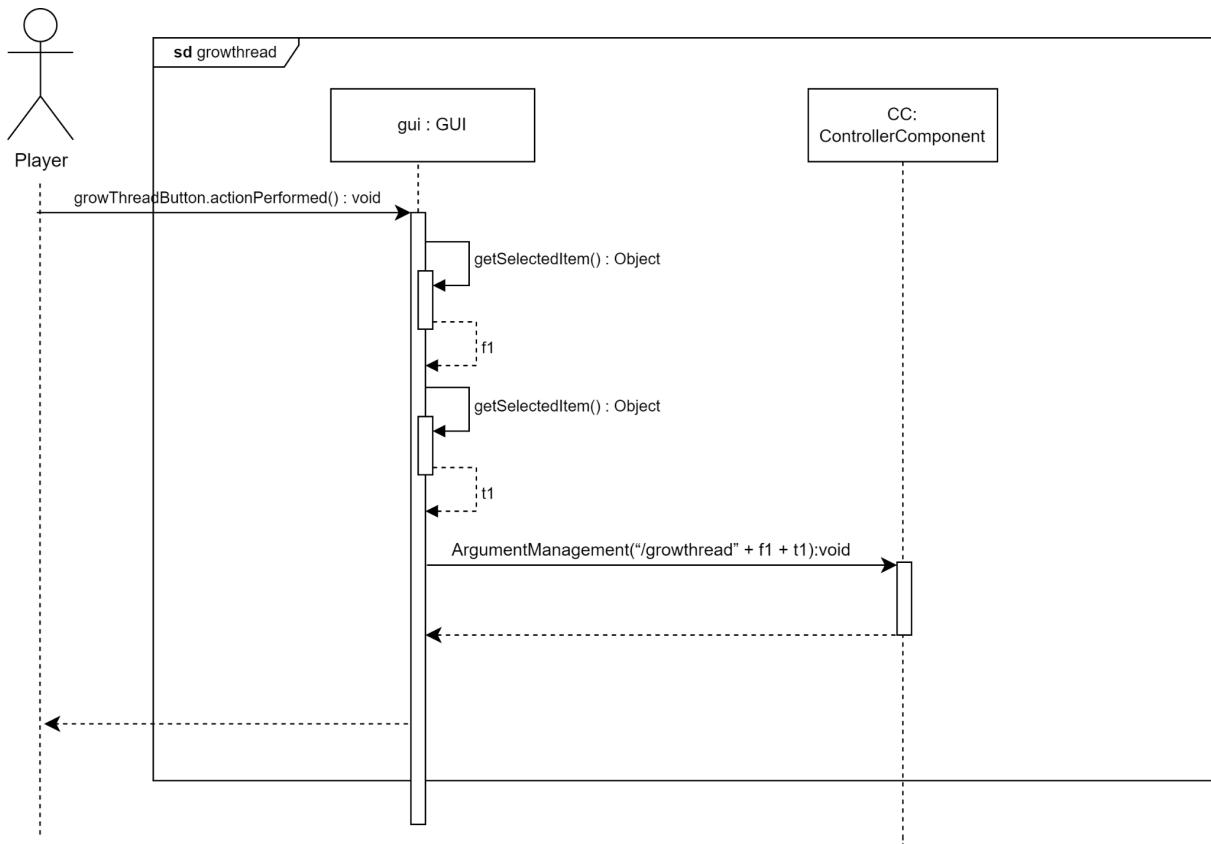
- **Metódusok**

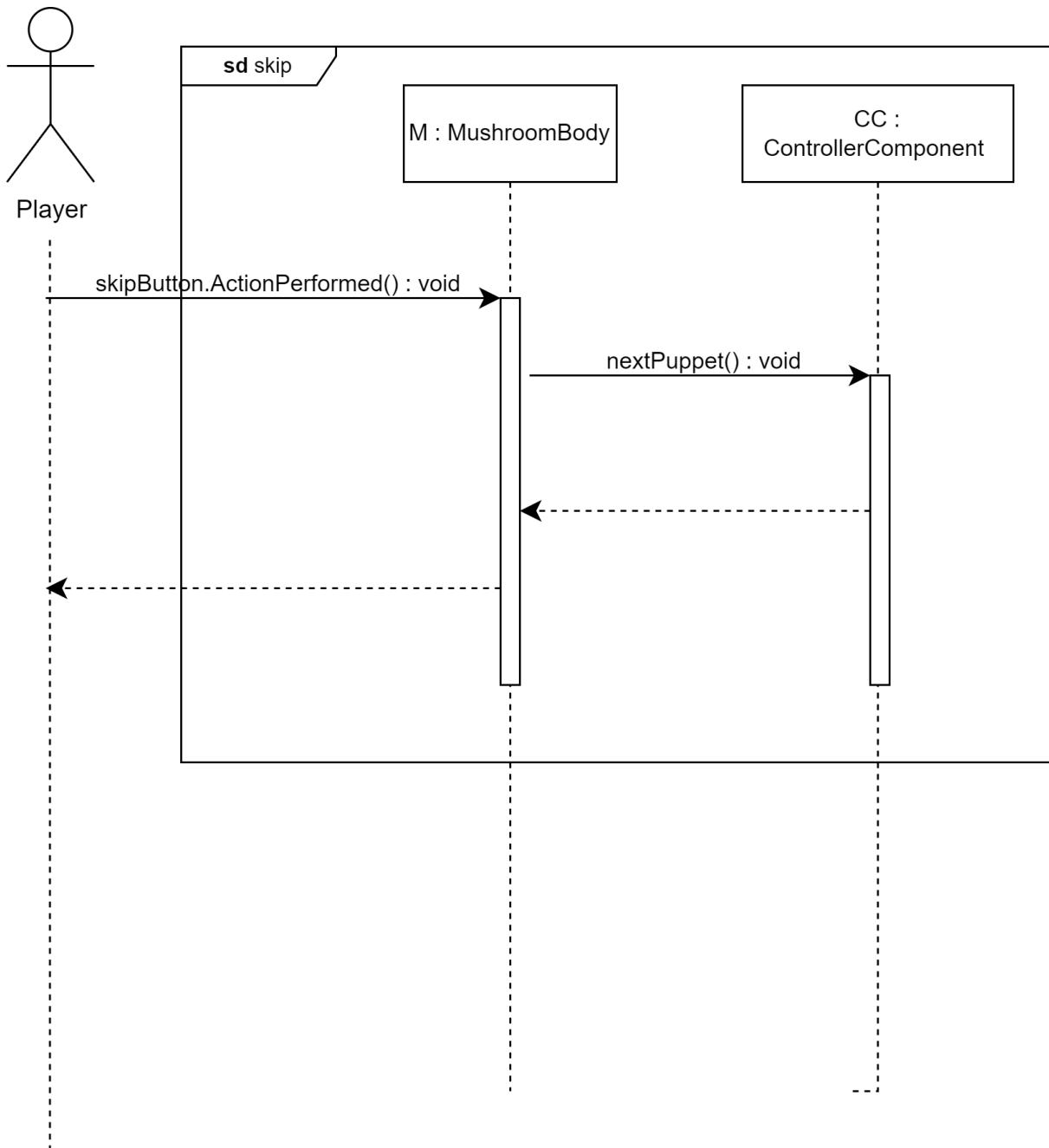
11.4 Kapcsolat az alkalmazói rendszerrel











11.5 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2025.05.02 20:00	1.5 óra	Yin	11.1 rajz készítése
2025.05.02. 20:00	2 óra	Bene	Tevékenység: 11.2 class diagram készítése
2025.05.03 14:00	0.5 óra	Kattan	Tevékenység: 11.1 leírás elkészítése
2025.05.03. 19:00	1 óra	Bene	Tevékenység: 11.2 class diagram befejezése
2025.05.03. 20:00	2 óra	Ji Jin Yan	Tevékenység: 11.2
2025.05.03. 21:00	2 óra	Yin	SD készítése
2025.05.04. 12:00	2 óra	Kattan	Tevékenység: 11.3 készítése
2025.05.04 16:00	3 óra	Ji Yuang	Tevékenység: 11.3 készítése
2025.05.04 18:00	2 óra	Ji Jin Yan	Tevékenység: 11.3
2025.05.04 16:00	1 óra	Bene	Tevékenység: 11.2 structure diagram megírása

13. Grafikus változat beadása

82 – being_chilling

Konzulens:
Tóth Bence

Csapattagok

Yin Jun	TN1KZI	1171276297yin@gmail.com
Bene Máté	K4C4W1	matthew.be04@gmail.com
Kattan Ibrahim Gábor	ZU22JJ	kattanbarhum2@gmail.com
Ji Jin Yan	ZGCKAL	jijinyan0201@gmail.com
Ji Yuang	UI7V2S	yuangji688@gmail.com

2025.05.19.

13. Grafikus változat beadása

13.1 Fordítási és futtatási útmutató

13.1.1 Fájllista

Fájl neve	Méret	Keletkezés ideje	Tartalom
build.gradle.kts	1KB	2025-05-19	build beállítások gradlewhez
build.gradle.ktsbuild.gradle.kts	60KB	2025-05-19	gradlew futtatásához szükséges
gradle\wrapper\gradle-wrapper.properties	1KB	2025-05-19	gradlew beállításai
gradlew	9KB	2025-05-19	gradlew beállításai
gradlew.bat	3KB	2025-05-19	gradlew használatához
settings.gradle.kts	1KB	2025-05-19	gradlew használatához
ControllerComponent.java	21KB	2025-05-19	A játék kontroller osztálya
InsectController.java	1KB	2025-05-19	Insect controller interface megvalósítása
MushroomBodyController.java	1KB	2025-05-19	MushroomBody controller interface
MushroomThreadController.java	1KB	2025-05-19	MushroomThread controller interface
SporeController.java	1KB	2025-05-19	Spore controller interface
TektonController.java	1KB	2025-05-19	Tekton controller interface
BiMap.java	2KB	2025-05-19	kétirányú HashMap osztálya
GameModel.java	1KB	2025-05-19	GameModel osztálya, mint egy játék adattároló.
Map.java	1KB	2025-05-19	Térkép osztálya
GInsect.java	1KB	2025-05-19	Grafikus rovar osztálya
GMushroomBody.java	1KB	2025-05-19	Grafikus gombatest osztálya
GSpore.java	1KB	2025-05-19	Grafikus spóra osztálya
GTekton.java	1KB	2025-05-19	Grafikus tekton osztálya
GUI.java	40KB	2025-05-19	Grafikus felhasználói felületet kezelő osztálya
CloneSpore.java	1KB	2025-05-19	A klón spóra osztálya
HasteSpore.java	1KB	2025-05-19	A gyorsító spóra osztálya
Insect.java	7KB	2025-05-19	A rovar osztálya
InsectSpecies.java	1KB	2025-05-19	Egy játékos által irányított rovarcsoport osztálya
LifeSupportTekton.java	2KB	2025-05-19	Life Support tekton osztálya
MultiThreadTekton.java	1KB	2025-05-19	Több gombafonalas tekton osztálya

MushroomBody.java	5KB	2025-05-19	Gombatest osztálya
MushroomlessTekton.java	2KB	2025-05-19	Gomba nélküli tekton osztálya
MushroomSpecies.java	1KB	2025-05-19	Egy játékos által irányított gombacsoport osztálya
MushroomThread.java	5KB	2025-05-19	Gombafonal osztálya
MuteSpore.java	1KB	2025-05-19	Némító spóra osztálya
ParaSpore.java	1KB	2025-05-19	bénító spóra osztálya
SlowSpore.java	1KB	2025-05-19	lassító spóra osztálya
Spore.java	1KB	2025-05-19	Spóra osztálya
Tekton.java	10KB	2025-05-19	Tekton osztálya
ThreadAbsorbTekton.java	2KB	2025-05-19	Gombafonal elnyelés tekton osztálya
MushroomBodyView.java	1KB	2025-05-19	MushroomBody view interface
InsectView.java	1KB	2025-05-19	Insect view interface
MushroomThreadView.java	1KB	2025-05-19	MushroomThread View interface
SporeView.java	1KB	2025-05-19	Spore View interface
TektonView.java	1KB	2025-05-19	Tekton View interface
Main.java	1KB	2025-05-19	Main osztálya
start.txt	4KB	2025-05-19	kezdő pályához szükséges

13.1.2 Fordítás és telepítés

A program lefordításához a gradlew.bat jar parancs futtatása (cmd-vel) szükséges a project mappában. Ez létrehozza a build/libs mappába a jar file-t. A telepítésnél az előbbi lefordításához létre hozott build/libs jar file maga a "telepített", futtatható alkalmazás. Ezt a fájlt lehet másolni és futtatni bármilyen gépen, ahol van Java futtatókörnyezet (JRE) telepítve.

13.1.3 Futtatás

A futtatás a gradlew.bat run parancs futtatásával lehetséges, vagy az előző résznél kapott .jar fájl futtatásával (vagyis java -jar *filename*.jar parancs segítségével lehet a fentebb említett mappában (build/libs)).

13.2 Értékelés

Tag neve	Tag neptun	Munka százalékban
Bene Máté	K4C4W1	20%
Yin Jun	TN1KZI	20%
Ji Jin Yan	ZGCKAL	20%
Kattan Ibrahim Gábor	ZU22JJ	20%
Ji Yuang	UI7V2S	20%

13.3 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2025.05.18. 15:00	2,5 óra	Ji Yuang	Tevékenység: Dokumentáció megírása
2025.05.17. 15:00	3 óra	Bene	Tevékenység: UI megírása
2025.05.17. 21:00	3 óra	Ji Jin Yan	Tevékenység: Rajzoló osztályok elkészítése
2025.05.17. 21:00	3 óra	Kattan	Tevékenység: Rajzoló osztályok elkészítése
2025.05.18. 14:00	2.5 óra	Kattan	Tevékenység: Rajzoló osztályok finomítása
2025.05.18. 15:00	6 óra	Bene	Tevékenység: Debug, fix, tekton rajzolás
2025.05.18. 17:00	3 óra	Ji Jin Yan	Tevékenység: Rajzoló osztályok finomítása
2025.05.18. 19:00	1 óra	Kattan	Tevékenység: Rajzoló osztályok finomítása
2025.05.18. 20:00	4 óra	Yin Jun	Tevékenység: button listenner, debugolas

14. Összefoglalás

14.1 A projektre fordított összes munkaidő

Tag neve	Munkaidő (óra)
Bene Máté	85
Kattan Ibrahim Gábor	67
Yin Jun	77
Ji Jin Yan	60
Ji Yuang	47.5
Összesen	336.5

- A feltöltött programok forrássorainak száma**

Fázis	KódSOROK száma
Szkeleton	2263
Prototípus	5137
Grafikus változat	1725
Összesen	9125

14.2 • Projekt összegzés

14.2.1 Mit tanultak a projektből konkrétan és általában?

Egy projekt hogyan zajlik, mik a teendők, hogyan érdemese elkezdeni, nekilátni.

14.2.2 Mi volt a legnehezebb és a legkönnyebb?

A kezdete volt a legnehezebb, vége egyszerűbb volt, mivel már meg volt a model, csak arra kellett építeni, ha jól lett meghaladni.

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

Igen

14.2.4 Ha nem, akkor hol okozott ez nehézséget?

-

14.2.5 Milyen változtatási javaslatuk van?

Kevesebb feladat, vagy több idő.

14.2.6 Milyen feladatot ajánlanának a projektre?

RPG játékot

14.2.7 Egyéb kritika és javaslat

-