

## 4. Analízis modell (II. változat)

23 – totoro

Konzulens:

Haragos Gergő Viktor

### Csapattagok

**Bohus János**

Dúcz Ákos

Galambos Csaba

Horváth Avarka

Végvári Péter

**XYOVFZ**

GC1RTE

BVBQKN

OODMXV

AW3JH1

**janos.bohus1@gmail.com**

akos.ducz@gmail.com

galambos.csaba05@gmail.com

avarkahorvath@gmail.com

vegvaripeter21@gmail.com

2025.03.05

## 4. Analízis modell kidolgozása

### 4.1 *Objektum katalógus*

#### 4.1.1 Pálya

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

#### 4.1.2 Tekton

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

#### 4.1.3 Fonalelnyelő tekton

Olyan tektonon, ahonnan a fonalak pár kör után felszívódnak.

#### 4.1.4 Gombamentes tekton

Olyan tekton, amin nem tud gombatest nőni.

#### 4.1.5 Egyfonalas tekton

Olyan tekton, amin csak egy fonál lehet.

#### 4.1.6 Gombatest

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

#### 4.1.7 Fonál

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

#### 4.1.8 Gombakolónia

Egy gombafaj összefüggő gombatestjei és fonalai alkotják.

#### 4.1.9 Gyorsító spóra

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

#### 4.1.10 Lassító spóra

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

#### 4.1.11 Bénító spóra

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

#### **4.1.12 Csáprágó bénító spóra**

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

#### **4.1.13 Rovar**

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

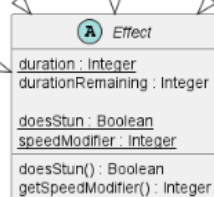
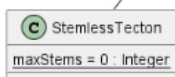
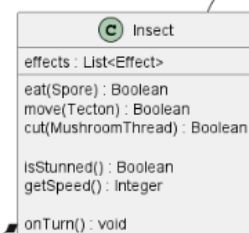
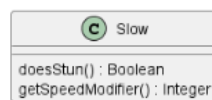
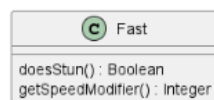
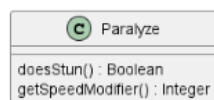
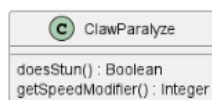
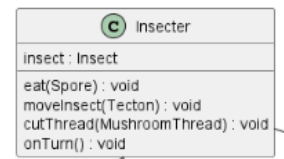
#### **4.1.14 Rovarász**

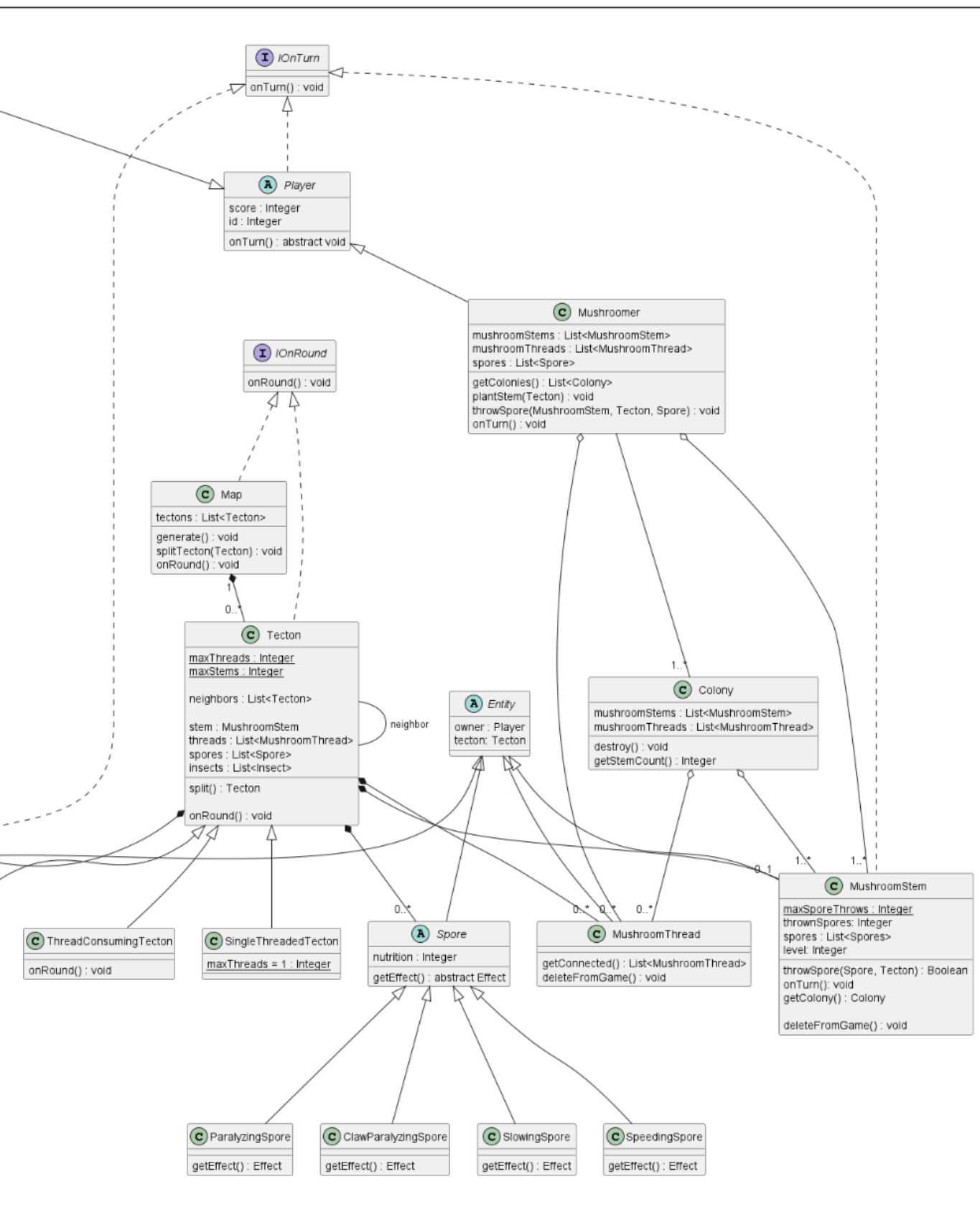
A rovarászok egy rovar irányítanak.

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



fungorium





## 4.3 Osztályok leírása

### 3.1.1 ClawParalyze

- **Felelősség**

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

- **Ősosztályok**

Effect → ClawParalyze

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

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

### 3.1.2 ClawParalyzingSpore

- **Felelősség**

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

- **Ősosztályok**

Entity → Spore → ClawParalyzingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

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

### 4.3.1 Colony

- **Felelősség**

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

- **Ősosztályok**

Az osztálynak nincs őse.

- **Interfészek**

A Colony osztály nem valósít meg interfészt.

- **Asszociációk**

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

- **Attribútumok**

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

- **Metódusok**

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

### 3.1.3 Effect

- **Felelősség**

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

- **Ősosztályok**

Nincs ősosztálya.

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

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



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

### 3.1.4 Entity

- **Felelősség**

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

Nem származik le egy osztályból sem.
- **Interfészek**

Nem valósít meg interfészt.
- **Asszociációk**
- **Attribútumok**
  - **Player owner**: Az a játékos, aki birtokolja az objektumot
  - **Tecton tecton**: Az entitást tartalmazó tekton.
- **Metódusok**

### 3.1.5 Fast

- **Felelősség**

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

- **Ősosztályok**

Effect → Fast

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

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

### 3.1.6 Insect

- **Felelősség**

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

- **Ősosztályok**

Entity → Insect

- **Interfészek**

Az Insect osztály az IOnTurn interfészt valósítja meg.

- **Asszociációk**

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

- **Attribútumok**

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

- **Metódusok**

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

### 3.1.7 Insecter

- **Felelősség**

A rovarász típusú játékosokat kezelő osztály.

- **Ősosztályok**

Player → Insecter

- **Interfészek**

Az Insecter osztály nem valósít meg interfészt.

- **Asszociációk**

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

- **Attribútumok**

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

- **Metódusok**

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

### 3.1.8 IOnRound

- **Felelősség**

Interfész, azok implementálják, akik valamit csinálnak minden körben.

- **Ősosztályok**

Nincs.

- **Asszociációk**

- **Metódusok**

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

### 3.1.9 IOnTurn

- **Felelősség**

Interfész, azok implementálják, akik valamit csinálnak minden lépésben.

- **Ősosztályok**

Nincs.

- **Asszociációk**

- **Metódusok**

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

### 3.1.10 Map

- **Felelősség**

A tektonokért felel, lényegében a pálya.

- **Össztályok**

Az osztálynak nincs össztálya.

- **Interfészek**

Az IOnRound interfészt valósítja meg.

- **Asszociációk**

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

- **Attribútumok**

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

- **Metódusok**

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

### 3.1.11 Mushroomer

- **Felelősség**

A gombász típusú játékost kezelő osztály.

- **Ősosztályok**

- Player → Mushroomer

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

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

- **Attribútumok**

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

- **Metódusok**

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

### 3.1.12 MushroomStem

- **Felelősség**

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

- **Ősosztályok**

Entity → MushroomStem

- **Interfészek**

Az IOnTurn interfészt valósítja meg.

- **Asszociációk**

- **Attribútumok**

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

- **Metódusok**

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

### 3.1.13 MushroomThread

- **Felelősség**

A tektonokon elhelyezkedő, és azokat összekötő gombafonalakat megvalósító osztály.

- **Ősosztályok**

Entity → MushroomThread

- **Interfészek**

Nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

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

### 3.1.14 Paralyze

- **Felelősség**

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

- **Ősosztályok**

Effect → Paralyze

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

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

### 3.1.15 ParalyzingSpore

- **Felelősség**

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

- **Ősosztályok**

Entity → Spore → ParalyzingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

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

### 3.1.16 Player

- **Felelősség**

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

- **Ősosztályok**

Nincs őosztálya.

- **Interfészek**

Az IOnTurn interfészt valósítja meg.

- **Asszociációk**

- **Attribútumok**

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

- **Metódusok**

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

### 3.1.17 SingleThreadedTecton

- **Felelősség**

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

- **Ősosztályok**

Tecton → SingleThreadedTecton

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

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

- **Metódusok**



### 3.1.18 StemlessTecton

- **Felelősség**

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

- **Ősosztályok**

Tecton → StemlessTecton

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

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

- **Metódusok**

### 3.1.19 Slow

- **Felelősség**

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

- **Ősosztályok**

Effect → Slow

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

- **Metódusok**

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

### 3.1.20 SlowingSpore

- **Felelősség**

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

- **Ősosztályok**

Entity → Spore → SlowingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

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

### 3.1.21 SpeedingSpore

- **Felelősség**

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

- **Ősosztályok**

Entity → Spore → SpeedingSpore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Attribútumok**

- **Metódusok**

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

### 3.1.22 Spore

- **Felelősség**

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

- **Ősosztályok**

Entity → Spore

- **Interfészek**

Az osztály nem valósít meg interfészt.

- **Asszociációk**

- **Attribútumok**

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

- **Metódusok**

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

### 3.1.23 Tecton

- **Felelősség**

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

- **Interfészek**

Megvalósítja az IOnRound interfészt.

- **Asszociációk**

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

- **Attribútumok**

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

- **Metódusok**

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

### 3.1.24 ThreadConsumingTecton

- **Felelősség**

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

- **Ősosztályok**

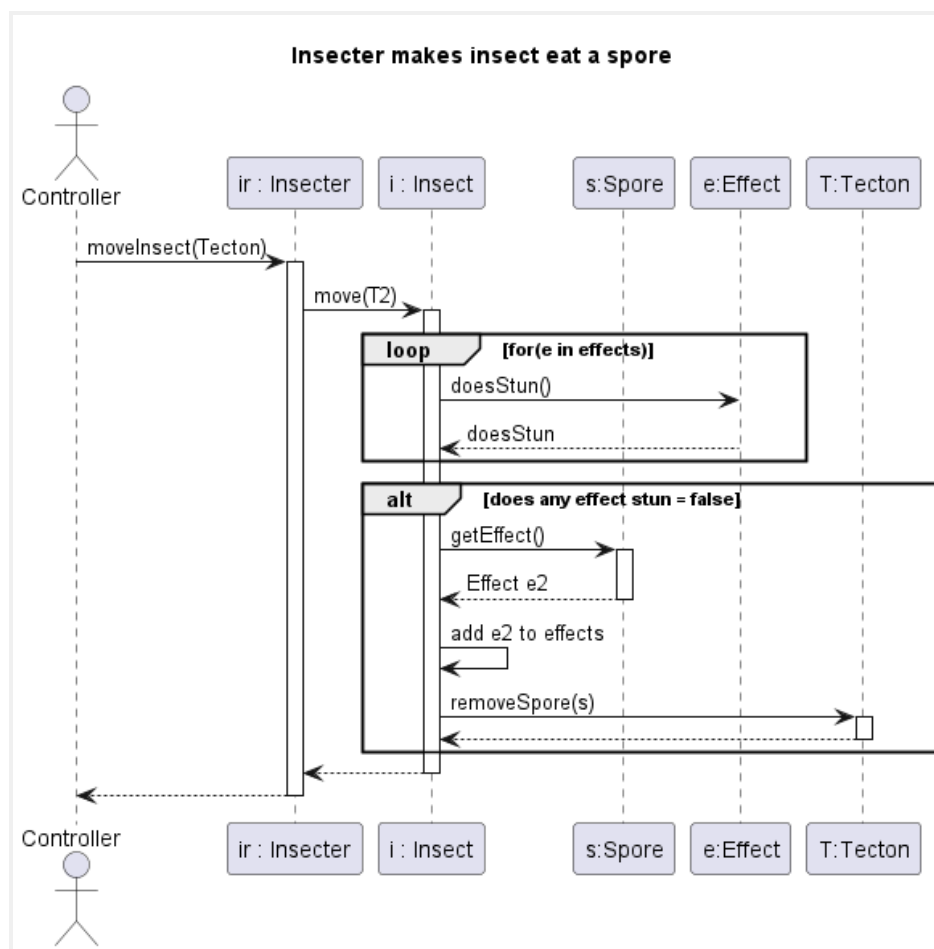
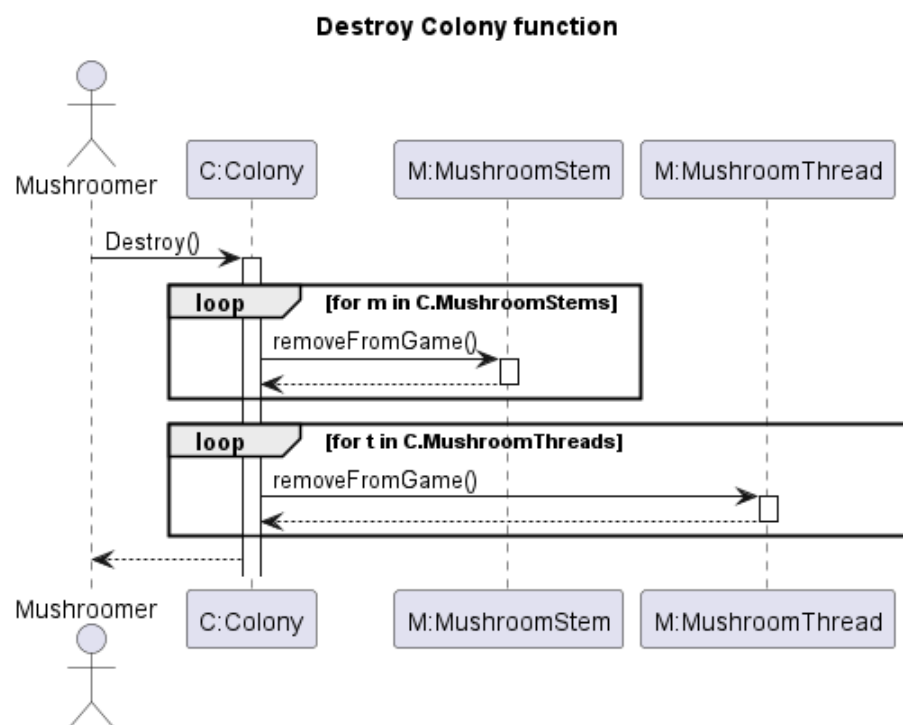
Tecton → ThreadConsumingTecton

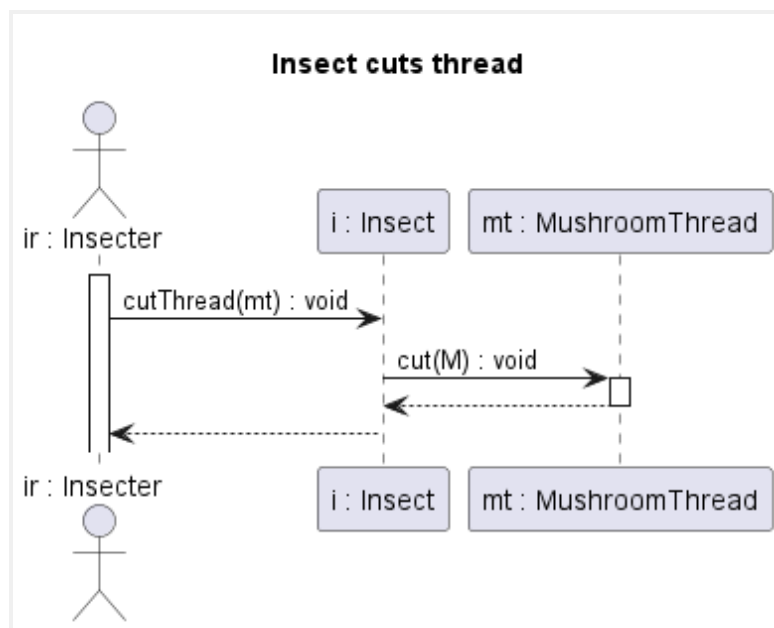
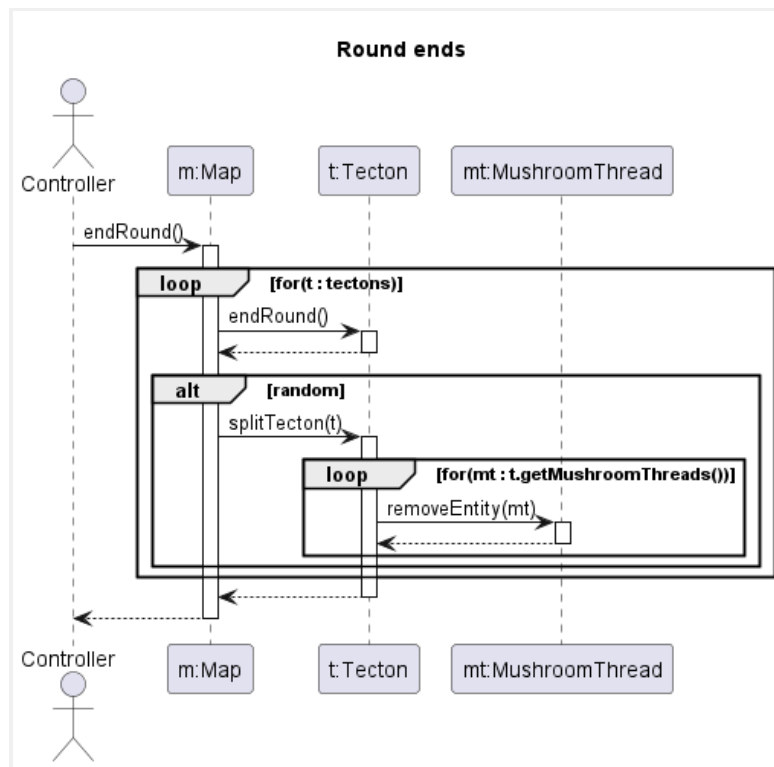
- **Interfészek**

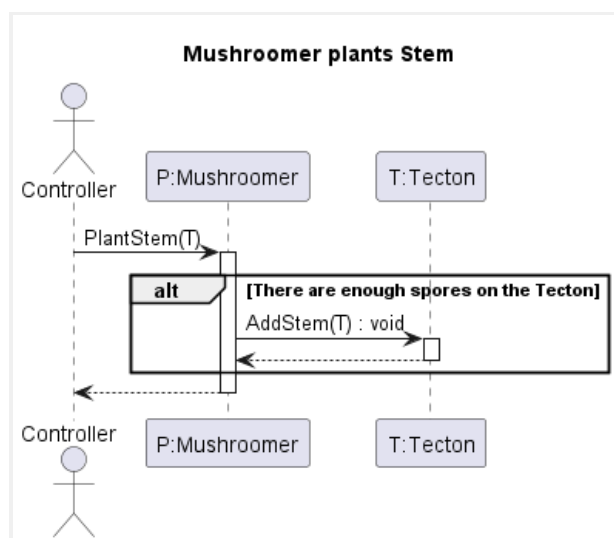
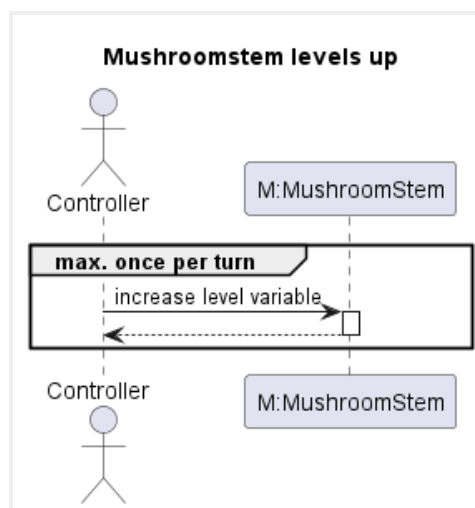
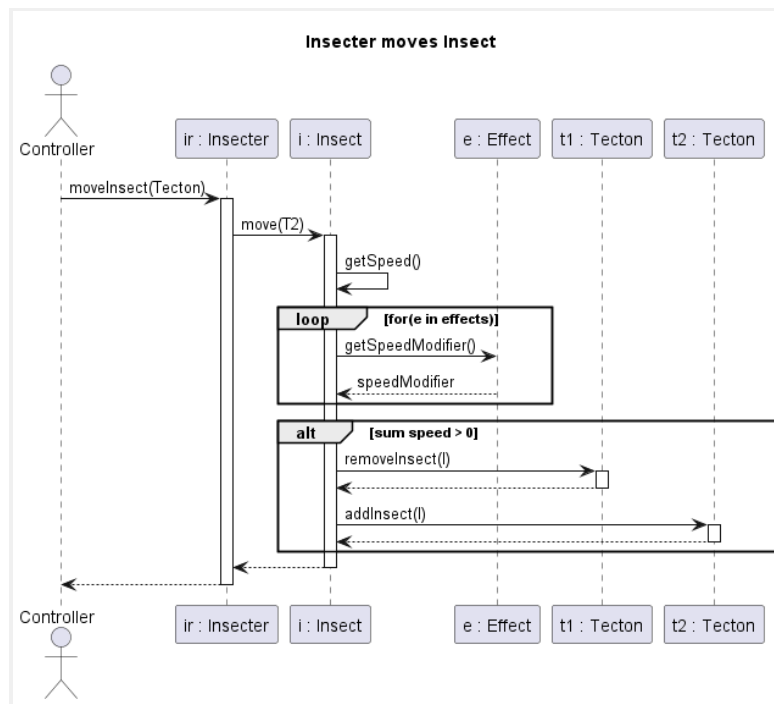
Az osztály nem valósít meg interfészt.

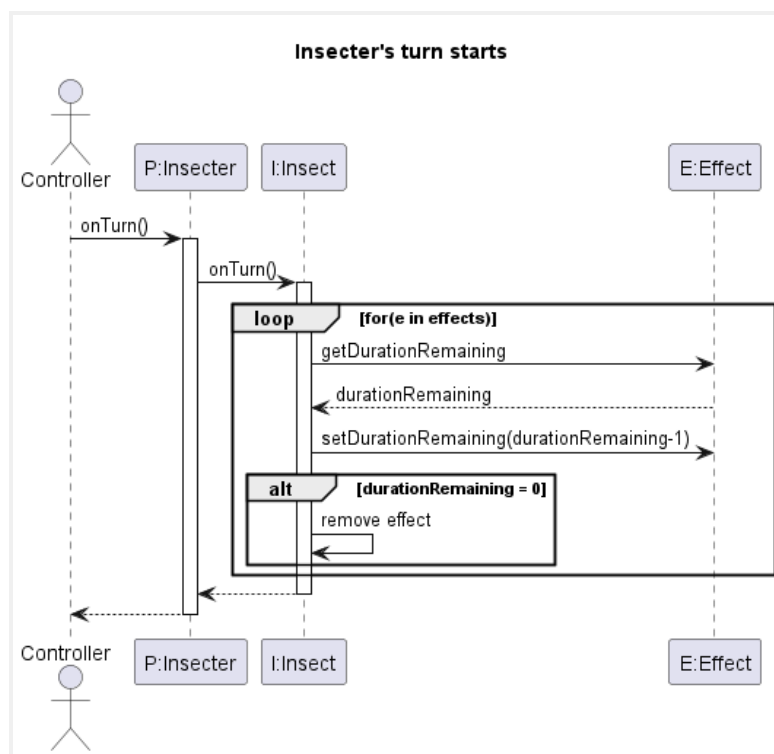
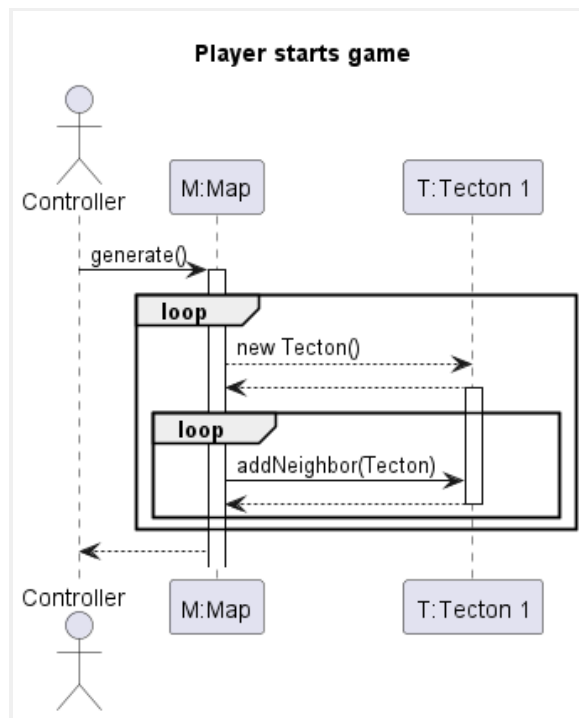
- **Asszociációk**
- **Attribútumok**
- **Metódusok**

## 4.4 Szekvencia diagramok

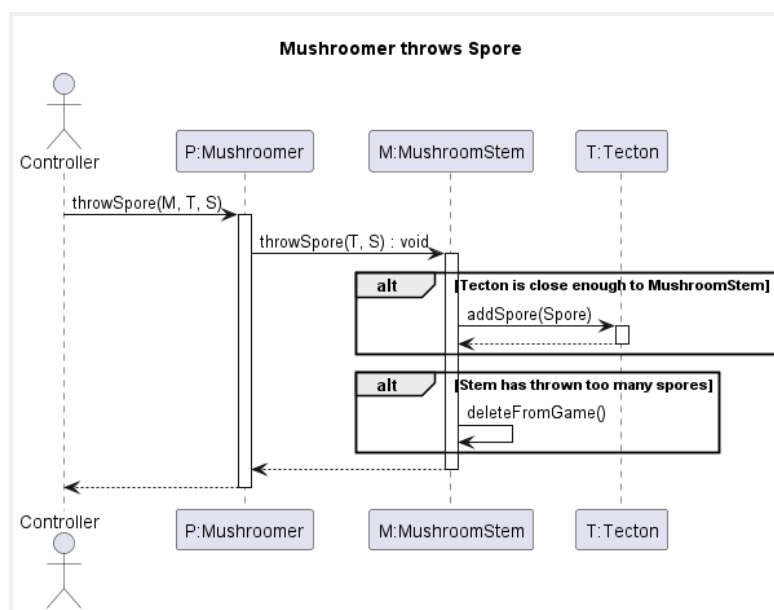
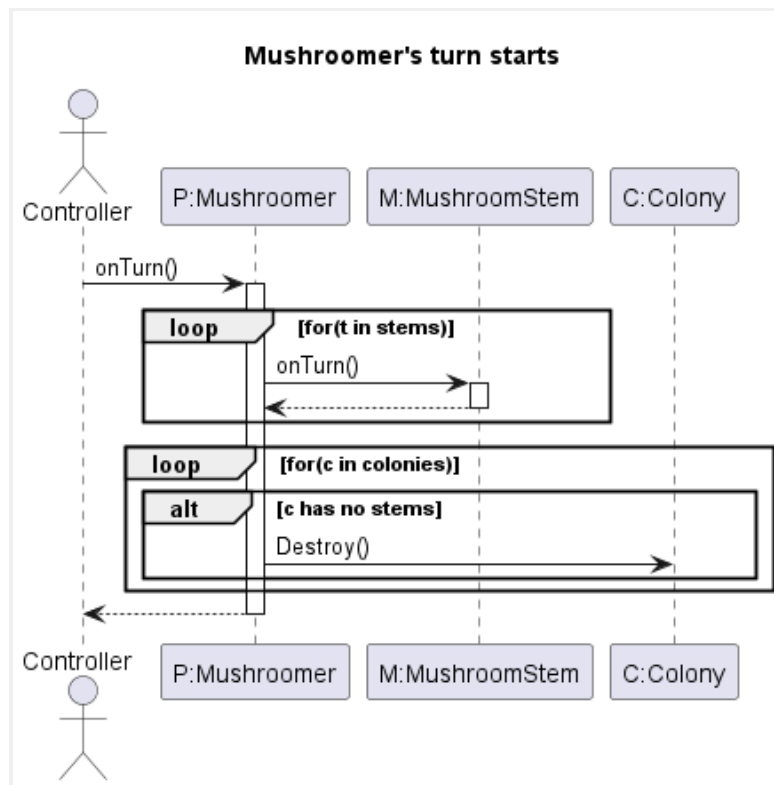






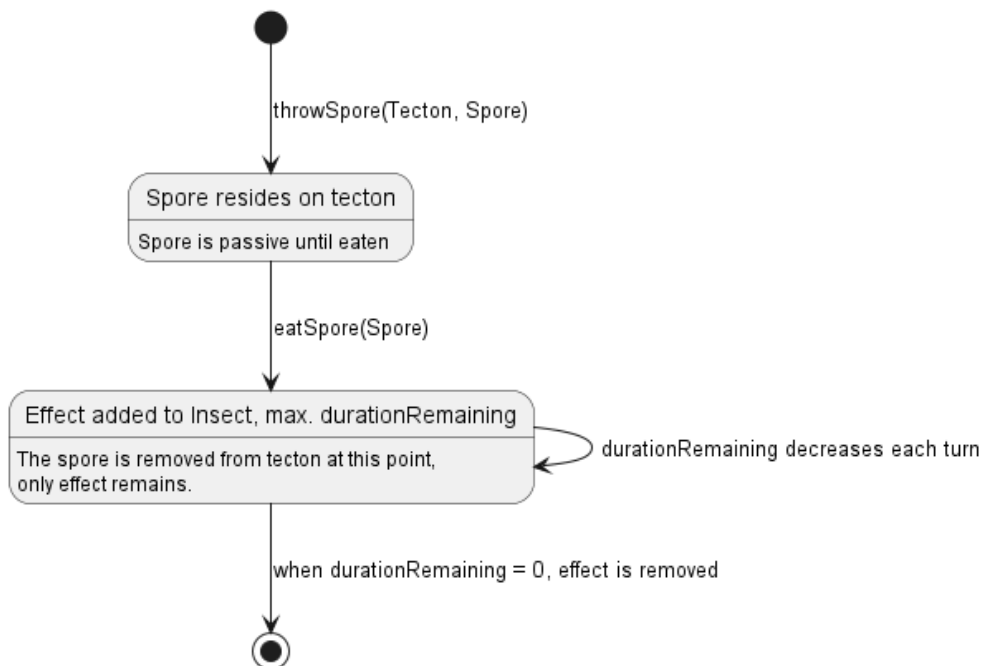




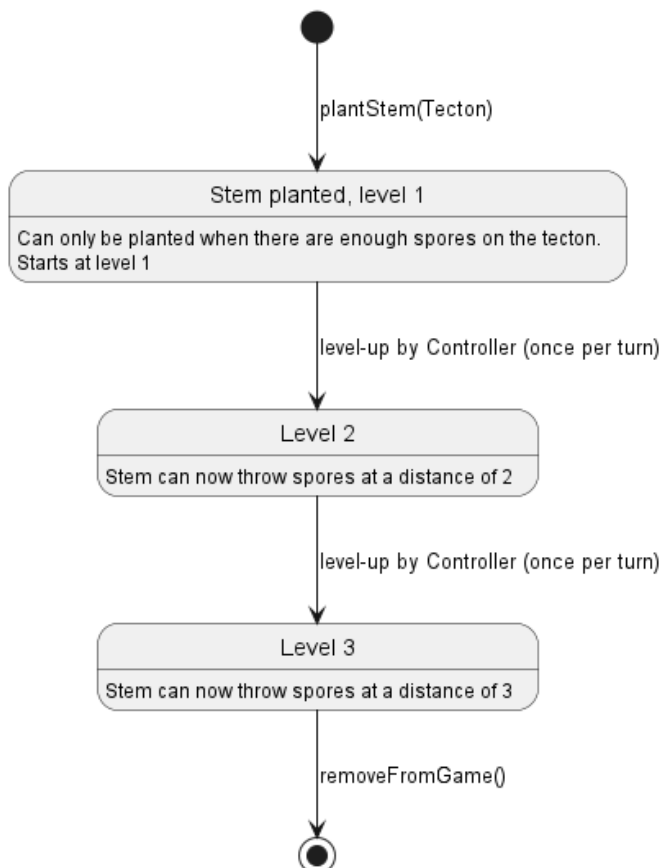


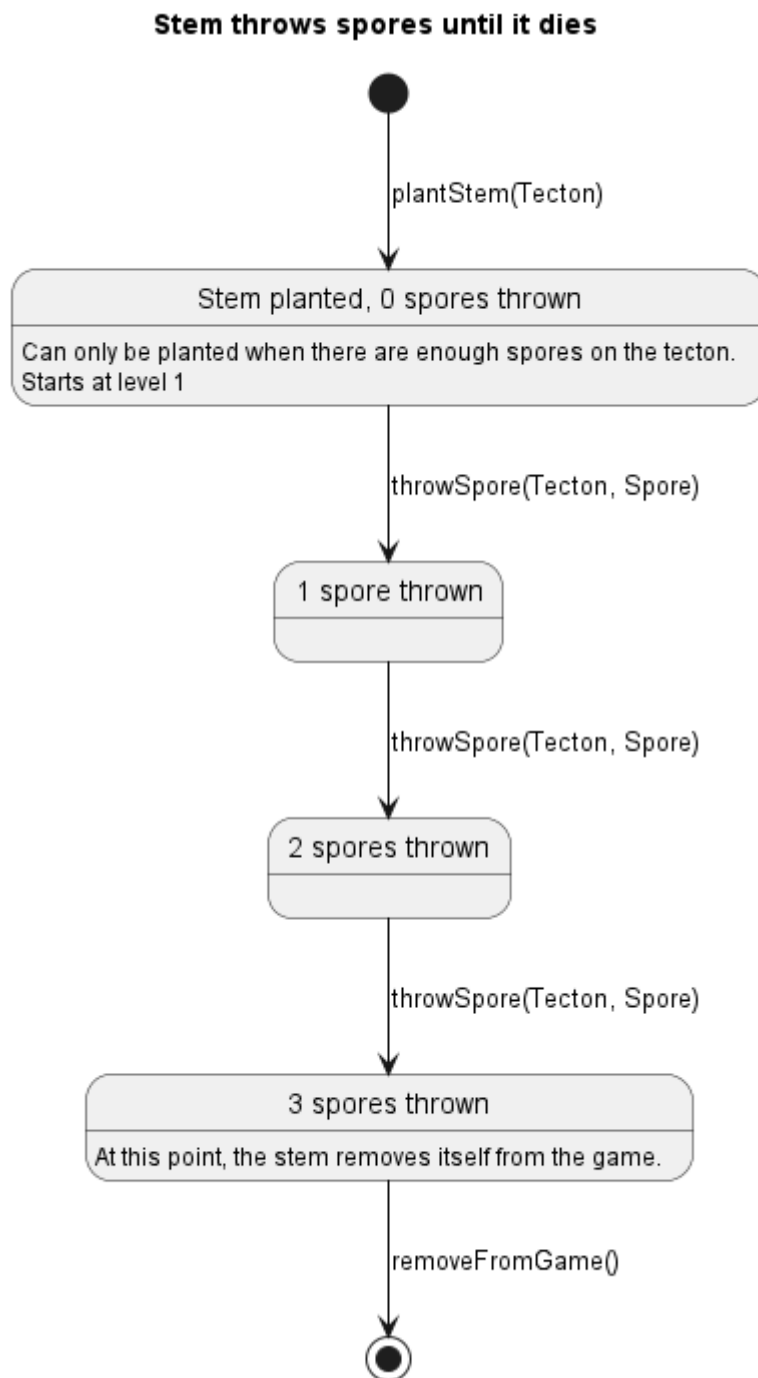
## 4.5 State-chartok

### Spore is thrown, then eaten



### Stem is upgraded by Player in their turns





**4.6 Napló**

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