# Analízis modell kidolgozása

## Objektum katalógus

[Minden, a feladatban szereplő objektum rövid, egy-két bekezdés hosszú ismertetése. Meg kell jelenjen minden objektumhoz, hogy mi a felelőssége. Informális leírás, ezért nem szabad még foglalkozni az örökléssel, az interfészekkel, az absztrakt osztályokkal, a segédosztályokkal.]

### Objektum1

[Felelősség informális leírása]

### Objektum2

[Felelősség informális leírása]

[Az objektumkatalógus alapján kiindulva kell megalkotni az objektumorientált analízis modellt. A 3.2, 3.3, 3.4, 3.5 alfejezetek ugyanannak a modellnek a különböző nézetei, ezért egyidőben, egymással összefüggésben készülnek. Megtörténik az objektumkatalógusban tárgyalt objektumok felelősségének formalizálása osztályokká, asszociációkká, attribútumokká, metódusokká. Csak publikus metódusok szerepelhetnek. Megjelennek az interfészek, az öröklés, az absztrakt osztályok. Segédosztályokra még nincs szükség.]

## Statikus struktúra diagramok

[Az objektum katalógusban szereplő objektumokat megvalósító osztályok kapcsolatait és publikus metódusait bemutató osztálydiagram(ok). Tipikus hibalehetőségek: csillag-topológia, szigetek.]

## Osztályok leírása

[A diagramon szereplő osztályok bemutatása. Az osztályok ABC sorrendben kövessék egymást. Interfészek esetén az Interfészek, Attribútumok pontok kimaradnak.]

### *AridTecton* (osztály)

#### Felelősség

Az *AridTecton* olyan *FertileTecton*, amelyen legfeljebb 1 gombafonál és legfeljebb 1 gombatest növekedhet. Rajta a gombafonalak 5 kör után automatikusan felszívódnak.

#### Ősosztályok

*Tecton* → *FertileTecton* (→ *AridTecton*)

#### Interfészek

*n/a*

#### Aggregációk

*n/a*

#### Asszociációk

*n/a*

#### Kompozíciók

*n/a*

#### Függőségek

*n/a*

#### Attribútumok

*n/a*

#### Metódusok

* ***canGrowMycelium()*:** ellenőrzi, hogy a tektonon nőhet-e gombafonál.
* ***canGrowMushroomBody()*:** ellenőrzi, hogy a tektonon nőhet-e gombatest.

### *CoarseTecton* (osztály)

#### Felelősség

A *CoarseTecton* olyan *Tecton*, amelyen nem növekedhet gombatest és gombafonál.

#### Ősosztályok

*Tecton* (→ *CoarseTecton*)

#### Interfészek

*n/a*

#### Aggregációk

*n/a*

#### Asszociációk

*n/a*

#### Kompozíciók

*n/a*

#### Attribútumok

*n/a*

#### Függőségek

*n/a*

#### Metódusok

* ***canGrowMycelium()*:** ellenőrzi, hogy a tektonon nőhet-e gombafonál.
* ***canGrowMushroomBody()*:** ellenőrzi, hogy a tektonon nőhet-e gombatest

### *Entomologist* (osztály)

#### Felelősség

A rovarok (*Insect*) irányításáért felelős játékosok osztálya. Tudomással bír arról, ha ő következik.

#### Ősosztályok

*Player* → *Entomologist*

#### Interfészek

*n/a*

#### Aggregációk

*n/a*

#### Asszociációk

* Az *Entomologist* asszociatív kapcsolatban áll az *Insect*tel, mivel a rovarász irányítja a rovarát. Ennek megfelelően a kapcsolat 1:1-es számosságú.

#### Kompozíciók

*n/a*

#### Függőségek

*n/a*

#### Attribútumok

*n/a*

#### Metódusok

* ***onTurnBegin(e: Player)*:** a *TurnBeginningSubscriber* interfésztől értesítést kap arra vonatkozóan, hogy új játékos következik.
* ***calculateScore*():** számolja, hogy hány pontja van.

### *FertileTecton* (osztály)

#### Felelősség

A *FertileTecton* olyan *Tecton*, amelyen a feltételek fennállása esetén lehet gombatest vagy gombafonál:

* gombatest akkor növeszthető a tektonon, ha gombafonál és legalább 3 spóra található rajta;
* gombafonál gombatestből vagy gombafonálból nőhet ki és szomszédos tektonra növeszthető. Ha egy tektonon található gombatest, rá nem lehet gombafonalat tenni.

#### Ősosztályok

*Tecton* (→ *FertileTecton*)

(A *FertileTecton* leszármazottja az *AridTecton* és a *MultiLayeredTecton*.)

#### Interfészek

*n/a*

#### Aggregációk

*n/a*

#### Asszociációk

*n/a*

#### Kompozíciók

*n/a*

#### Függőségek

*n/a*

#### Attribútumok

*n/a*

#### Metódusok

* ***canGrowMycelium()*:** ellenőrzi, hogy a tektonon nőhet-e gombafonál.
* ***canGrowMushroomBody()*:** ellenőrzi, hogy a tektonon nőhet-e gombatest

### *GameManager* (osztály)

#### Felelősség

A *GameManager* indítja el a játékot és határozza meg a végén a győztes személyét.

#### Ősosztályok

*n/a*

#### Interfészek

*n/a*

#### Aggregációk

* A *GameManager* kompozíciós kapcsolatban áll a *Player*-rel, mivel nyilvántartja a játékosok listáját. Ennek megfelelően egy rész-egész viszony alapulvételével a *GameManager* képviseli az egészt.

#### Asszociációk

*n/a*

#### Kompozíciók

*n/a*

#### Függőségek

*n/a*

#### Attribútumok

*n/a*

#### Metódusok

* ***startNewGame()*:**új játékot indít.
* ***determineWinner()*:** meghatározza a győztes személyét.

### *Insect* (osztály)

#### Felelősség

A rovarász (*Entomologist*) irányítása alá tartozó objektum. Képes a gombafonalak mentén mozogni, gombafonalat elvágni és spórát fogyasztani.

#### Ősosztályok

*n/a*

#### Interfészek

* Az *Insect* megvalósítja a *TurnBeginningSubscriber* interfészt. Az interfész révén kap tájékoztatást arról, ha új játékos következik.

#### Aggregációk

*n/a*

#### Asszociációk

*n/a*

#### Kompozíciók

*n/a*

#### Függőségek

* Az *Insect* asszociatív kapcsolatban áll az *Entomologist*tel, mivel a rovart az azért felelős rovarász irányítja. Ennek megfelelően a kapcsolat 1:1-es számosságú.

#### Attribútumok

* ***location:Tecton*** → nyilvántartja, melyik tektonon tartózkodik a rovar.
* ***remainingMoves:int*** → számolja, hogy hány lépése van még az adott körben.
* ***sporeCount: int*** → számolja a megevett spórákat.

#### Metódusok

* ***cutMycelium()*:** elvág egy gombafonalat.
* ***eatSpore()*:** elfogyaszt egy spórát.
* ***move()*:** gombafonál mentén halad.
* ***onTurnBegin()*:** a *TurnBeginningSubscriber* interfésztől értesítést kap arra vonatkozóan, hogy új játékos következik. Ennek segítségével tud a rovar a köre végén egy véletlenszerű fonálra „elmenekülni”, ha eltűnik alatta a fonál.

### *InsectEffect* (enum)

#### Felelősség

Az *InsectEffect* révén kerül meghatározásra, hogy a spóra elfogyasztása milyen élettani hatást gyakorol a rovarra:

* *slowness*: lelassítja a rovart, azaz egy körben 1 lépést tehet meg. Ez a hatás 3 körön át tart.
* *speed*: felgyorsítja a rovart, azaz egy körben 3 lépést tehet meg. Ez a hatás 3 körön át tart.
* *prevent\_cut*: megakadályozza a rovart abban, hogy fonalat vágjon el. Ez a hatás 3 körön át tart.
* *stun*: a rovar egy körön át nem tud aktivitást kifejteni.
* *nothing*: semmilyen hatással nem jár.

#### Ősosztályok

*n/a*

#### Interfészek

*n/a*

#### Aggregációk

*n/a*

#### Asszociációk

*n/a*

#### Kompozíciók

*n/a*

#### Függőségek

* Az *InstectEffect* asszociációs kapcsolatban áll a *Spore*-ral, mivel meghatározza, milyen következménnyel jár a rovar számára a spóra elfogyasztása.
* Az *InstectEffect* asszociációs kapcsolatban áll az *Insect*-tel, mivel meghatározza, milyen következménnyel jár a rovar számára a spóra elfogyasztása.

#### Attribútumok

*n/a*

#### Metódusok

*n/a*

### *Mushroom* (osztály)

#### Felelősség

A gombász felelősségi körébe tartozó objektum, amelyből gombafonál nőhet ki.

#### Ősosztályok

(A *Mushroom* leszármazottja a *Mycelium* és a *MushroomBody*.)

#### Interfészek

*n/a*

#### Aggregációk

*n/a*

#### Asszociációk

* A *Mushroom* asszociatív kapcsolatban áll a *Mycologist*-tel, mivel a gombász felelős a gombáiért és a gombafonalaiért. Ez meghatározza a kapcsolat számosságát (*Mycologist* – *Mushroom*: 1 – 0…\*).

#### Kompozíciók

* A *Mushroom* kompozíciós kapcsolatban áll a *Tecton-*nal, mivel egy tekton létezése meghatározza a gomba létezését is. Ennek megfelelően egy rész-egész viszony alapulvételével a *Tecton* képviseli az egészt.

#### Függőségek

*n/a*

#### Attribútumok

* ***location:Tecton*** → nyilvántartja, melyik tektonon található a gombarész.

#### Metódusok

* ***growMycelium()*:** gombafonalat növeszt.
* ***sustains()***: megállapítja, hogy gombafonálról vagy gombatestről van-e szó.

### *MushroomBody* (osztály)

#### Felelősség

Spórák termelésért és kilövésért felelős gombarész.

#### Ősosztályok

*Mushroom* → *MushroomBody*

#### Interfészek

* Az *MushroomBody* megvalósítja a *TurnBeginningSubscriber* interfészt. Az interfész révén kap tájékoztatást arról, ha új játékos következik.

#### Aggregációk

* A *MushroomBody* aggregációs kapcsolatban áll a *Spore*-ral, mivel az előbbi termeli és szétszórja a spórákat. Ennek megfelelően a *MushroomBody* az aggregátor.

#### Asszociációk

* A *Mushroom* asszociatív kapcsolatban áll a *Mycologist*-tel, mivel a gombász felelős a gombáiért és a gombafonalaiért. Ez meghatározza a kapcsolat számosságát (*Mycologist* – *Mushroom*: 1 – 0…\*).

#### Kompozíciók

*n/a*

#### Függőségek

*n/a*

#### Attribútumok

*n/a*

#### Metódusok

* ***ejectSpores()*:** szétszórja a spórákat.
* ***sustains()***: megállapítja, hogy gombafonálról vagy gombatestről van-e szó.
* ***onTurnBegin(e: Player)*:** a *TurnBeginningSubscriber* interfésztől értesítést kap arra vonatkozóan, hogy új játékos következik. Erre az információra a spóratermelés miatt van szüksége.

### *Mycelium* (osztály)

#### Felelősség

Gombatest növesztésére képes gombarész. Tektontörés esetén elszakad és elsorvad.

#### Ősosztályok

*Mushroom* → *Mycelium*

#### Interfészek

*n/a*

#### Aggregációk

*n/a*

#### Asszociációk

* A *Mycelium* asszociatív kapcsolatban áll a *GameBoard*-dal, mivel a tektontörés esetén a gombafonál elszakad és elsorvad.

#### Kompozíciók

*n/a*

#### Függőségek

*n/a*

#### Attribútumok

*n/a*

#### Metódusok

* ***cut()*:** tektontörés esetén a gombafonál elszakad.
* ***delete()*:** a tektontörés esetén elszakadt gombafonál elsorvad.
* ***sustains()***: megállapítja, hogy gombafonálról vagy gombatestről van-e szó.

### *Mycologist* (osztály)

#### Felelősség

A gombákért (*Mushroom*) felelős játékosok osztálya. Tudomással bír arról, ha ő következik és számolja a pontszámát. Képes továbbá a játékmezőhöz egy új saját gombát hozzáadni.

#### Ősosztályok

*Player* → *Mycologist*

#### Interfészek

*n/a*

#### Aggregációk

*n/a*

#### Asszociációk

* Az *Mycologist* asszociatív kapcsolatban áll a *Mushroom*-mal, mivel a gombász felelős a gombáiért és a gombafonalaiért. Ez meghatározza a kapcsolat számosságát (*Mycologist* – *Mushroom*: 1 – 0…\*).

#### Kompozíciók

*n/a*

#### Függőségek

*n/a*

#### Attribútumok

* ***grows: int*:** tárolja, hogy a gombász mennyi gombafonalat tud növeszteni az adott körben.

#### Metódusok

* ***addMushroomBody*():** hozzáad egy új gombatestet ad a játékmezőhöz.
* ***calculateScore*():** számolja, hogy hány pontja van.
* ***onTurnBegin(e: Player)*:** a *TurnBeginningSubscriber* interfésztől értesítést kap arra vonatkozóan, hogy új játékos következik.

### *MultiLayeredTecton* (osztály)

#### Felelősség

A *MultiLayeredTecton* olyan *FertileTecton*, amelyen legfeljebb 1 gombatest és legfeljebb 3 gombafonál növekedhet.

#### Ősosztályok

*Tecton* → *FertileTecton* (→ *MultiLayeredTecton*)

#### Interfészek

*n/a*

#### Aggregációk

*n/a*

#### Asszociációk

*n/a*

#### Kompozíciók

*n/a*

#### Függőségek

*n/a*

#### Attribútumok

*n/a*

#### Metódusok

* ***canGrowMycelium()*:** ellenőrzi, hogy a tektonon nőhet-e gombafonál.
* ***canGrowMushroomBody()*:** ellenőrzi, hogy a tektonon nőhet-e gombatest.

### *Player* (osztály)

#### Felelősség

A játék részvevőinek (*Entomologist*, *Mycologist*) ősosztálya. Tudomással bír arról, hogy melyik játékos következik és számolja a játékosok pontszámát.

#### Ősosztályok

*n/a*

#### Interfészek

* A *Player* megvalósítja a *TurnBeginningSubscriber* interfészt. Az interfész révén kap tájékoztatást arról, ha új játékos következik.

#### Aggregációk

*n/a*

#### Asszociációk

*n/a*

#### Kompozíciók

* A *Player* kompozíciós kapcsolatban áll a *GameManager*-rel, amely nyilvántartja a játékosok listáját. Ennek megfelelően egy rész-egész viszony alapulvételével a *GameManager* képviseli az egészt.

#### Függőségek

*n/a*

#### Attribútumok

*n/a*

#### Metódusok

* ***calculateScore():*** számolja, hogy hány pontja van az adott játékosnak.
* ***onTurnBegin(e: Player)*:** a *TurnBeginningSubscriber* interfésztől értesítést kap arra vonatkozóan, hogy új játékos következik.

### *Spore* (osztály)

#### Felelősség

Gombatestnek egy adott tektonon történő növesztéséhez spóra szükséges. A spóra a gombafonal növekedését is segíti olyan tekton irányába, ahol a spóra található. A rovarok spórával táplálkoznak, és az elfogyasztása élettani hatást gyakorolhat a rovarra.

#### Ősosztályok

*n/a*

#### Interfészek

*n/a*

#### Aggregációk

* A *Spore* aggregációs kapcsolatban áll a *MushroomBody*-val, amely termeli és szétszórja a spórákat. Ennek megfelelően a *MushroomBody* az aggregátor.
* A *Spore* aggregációs kapcsolatban áll a *Tecton*-nal, mivel a tektonra spóraszórás esetén spórák eshetnek. Spóraszóráskor az egy gombatestből kilövellt összes spóra ugyanarra a tektonra esik, ami meghatározza a kapcsolat számosságát (*Tecton* – *Spore*: 1 – 0…\*).

#### Asszociációk

*n/a*

#### Kompozíciók

*n/a*

#### Függőségek

* A *Spore* függőségi kapcsolatban áll az *InstectEffect*tel, amely meghatározza, hogy milyen következménnyel jár a rovar számára a spóra elfogyasztása.

#### Attribútumok

*n/a*

#### Metódusok

* ***canGrowMycelium()*:** ellenőrzi, hogy a tektonon nőhet-e gombafonál.
* ***canGrowMushroomBody()*:** ellenőrzi, hogy a tektonon nőhet-e gombatest

### *Tecton* (osztály)

#### Felelősség

A tektöntörés keretében felelős az új tektonok létrehozásáért és a szomszédsági kérdések meghatározásáért. Ennek érdekében nyilvántartja, hogy hány kör múlva következik be tektontörés. Az ehhez szükséges körök száma véletlenszerűen kerül meghatározásra.

#### Ősosztályok

(A *Tecton* leszármazottja a *CoarseTecton* és a *FertileTecton*.)

#### Interfészek

* A *Tecton* megvalósítja a *TurnBeginningSubscriber* interfészt. Az interfész révén kap tájékoztatást arról, ha új játékos következik, amelynek segítségével a köröket számolni tudja.

#### Aggregációk

* A *Tecton* aggregációs kapcsolatban áll a *Spore*-ral, mivel a tektonra spóraszórás esetén spórák eshetnek. Spóraszóráskor az egy gombatestből kilövellt összes spóra ugyanarra a tektonra esik, ami meghatározza a kapcsolat számosságát (*Tecton* – *Spore*: 1 – 0…\*).

#### Asszociációk

*n/a*

#### Kompozíciók

* A *Tecton* kompozíciós kapcsolatban áll a *GameBoard*-dal, mivel a Tecton a játékmező atomi alapeleme. Ennek megfelelően egy rész-egész viszony alapulvételével a *Gameboard* képviseli az egészt.
* A *Tecton* kompozíciós kapcsolatban áll a *Mushroom-*mal, mivel egy tekton létezése meghatározza a gomba létezését is. Ennek megfelelően egy rész-egész viszony alapulvételével a *Tecton* képviseli az egészt.

#### Attribútumok

* ***mushroomBody***: nyilvántartja a tektonon található gombatesteket.
* ***mycelia*:** nyilvántartja a tektonon található gombafonalakat.

#### Metódusok

* ***canGrownMycelium()*:** megállapítja, hogy a tektonon növeszthető-e gombafonál.
* ***canGrowMushroomBody()*:** megállapítja, hogy a tektonon növeszthető-e gombatest.
* ***growMushroomBody()*:** a tektonon gombafonalat növeszt.
* ***growMycelium()*:** a tektonon gombatestet növeszt.
* ***onTurnBegin(e: Player)*:** a *TurnBeginningSubscriber* interfésztől értesítést kap arra vonatkozóan, hogy új játékos következik.

### *TurnManager* (osztály)

#### Felelősség

Ez az osztály irányítja, hogy mikor ér véget egy kör. Mint *publisher* tájékoztatja erről a feliratkozó *TurnBeginSubscriber* interfészt.

#### Ősosztályok

*n/a*

#### Interfészek

* A *TurnManager* megvalósítja az *OnGameBeginSubscriber* interfészt. Az interfész révén tájékoztatást kap a játék kezdetéről.

#### Aggregációk

*n/a*

#### Asszociációk

* A *TurnManager* asszociatív kapcsolatban áll a *TurnBeginningSubscriber*-rel, amelyet *publisher*ként tájékoztat a kör végéről.

#### Kompozíciók

*n/a*

#### Függőségek

*n/a*

#### Attribútumok

*n/a*

#### Metódusok

* ***subcribe(s: TurnBeginningSubscriber)***: a segítségével a *publisher* szolgáltatásaira lehet feliratkozni.
* ***unsubscribe(s: TurnBeginningSubscriber):*** a segítségével a *publisher* szolgáltatásairól lehet leiratkozni.
* ***endTurn():*** egy másik játékos körének kezdetekor meghívja a *notifySubscribers()*-t.
* ***notifySubscribers()*:** tájékoztatja a feliratkozó(ka)t *(subscribers)* a kör végéről.
* ***onGameBegin():*** tájékoztatást kap a játék kezdetéről.

## Szekvencia diagramok

[Inicializálásra, use-case-ekre, belső működésre. Konzisztens kell legyen az előző alfejezettel. Minden metódus, ami ott szerepel, fel kell tűnjön valamelyik szekvenciában. Minden metódusnak, ami szekvenciában szerepel, szereplnie kell a valamelyik osztálydiagramon. Fontos, hogy az aktor use-case-eitől bármelyik diagramhoz el lehessen jutni a metódushívások követésével.]

## State-chartok

[Csak azokhoz az osztályokhoz, ahol van értelme. Egyetlen állapotból álló state-chartok ne szerepeljenek. A játék működését bemutató state-chart-ot készíteni tilos.]

## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
|  |  |  |  |