**3.** **Analízis modell kidolgozása**

5. – Pandora

Konzulens:

Elekes Márton

Csapattagok

|  |  |  |
| --- | --- | --- |
| Szabó András | GA9BCR | sbasiliscus@gmail.com |
| Katona Benedek | LNU506 | kbeniboy2004@gmail.com |
| Lipóczi Levente László | SUSXSW | lipoczi.levente@gmail.com |
| Fekete Botond | VZ6YKQ | botond.fekete.010@gmail.com |
| Kőműves Marcell | LLDPZI | kommarci03@gmail.com |

2025.03.03

**3.1** **Objektum katalógus**

**3.1.1** **Game (Játék) ⚙️**

Felelőssége a játékosok kezelése (tárolása és léptetése), a körök számolása, a játékmenet irányítása és a győztes meghatározása a játék legvégén.

**3.1.2** **Player (Játékos) 🎮**

Felelőssége (a saját pontszámának nyilvántartása,) a körben a saját lépésének végrehajtása és az állapota.

**3.1.3** **Shroomer (Gombász) 🍄🕹️**

Felelőssége a saját fonál- és gombatest-hálózat nyilvántartása, valamint a spórák szórása.

**3.1.4** **Bugger (Rovarász) 🐜**

Nyilvántartja az állapotát. Spórák evésével különböző hatások érhetik.

**3.1.5** **Tekton 🗺️**

Felelőssége a rajta található gombák, spórák, hozzá csatlakozó gombafonalak, valamint szomszédos tektonok tárolása. *Meghatározza a gombák növekedésének és a spórák tárolásának feltételeit.*

***Tekton-típusok:***

**3.1.6** **Stone (Kő)** 🗿

Gombatest nem fejlődhet ki rajta, csak gombafonál mehet át rajta.

**3.1.7 Dirt (Föld) 🌱**

Semmilyen külön felelőssége sincs.

**3.1.8 Peat (Tőzeg) 🍂**

Több gombafonál is átmehet rajta.

**3.1.9 Swamp (Mocsár) 🐊**

A gombafonalak 3 kör után felszívódnak.

**3.1.10 Mushroom (Gombatest) 🍄**

Maradék eldobható a spórák számát tárolja. Felelőssége a korának kezelése (fiatal/idős), ami meghatározza a spóraszórás távolságát.

**3.1.11 Hypa (Gombafonál) 🧵**

Felelőssége a kapcsolatok fenntartása a tektonok között. (referenciával rendelkezik)

**3.1.12 Spore (Spóra) 🦠**

Felelőssége a különböző hatások kiváltása a rovarászokra és az interakciók szabályozása. A rovarokra kifejtett hatások időtartamát és típusát kezeli.

***Spóratípusok:***

**3.1.13 Booster (Gyorsító spóra) 🏃‍♂️⚡**

A rovar 2 kör erejéig kétszer mozoghat egy körben.

**3.1.14 Slower (Lassító spóra) 🐌**

A rovar 2 kör erejéig csak egyszer mozoghat.

**3.1.15 Paralyzer (Bénító spóra) 🚷♿**

A rovar 2 körből kimarad.

**3.1.16 BiteBlocker (Fonálvágást gátló spóra) ✂️❌**

A rovar 2 kör erejéig nem vághat el gombafonal.

**3.2** **Statikus struktúra diagramok**

A képen szöveg, diagram, Tervrajz, Műszaki rajz látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

A képen szöveg, diagram, képernyőkép, Párhuzamos látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

A képen szöveg, diagram, Párhuzamos, sor látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

**3.3** **Osztályok leírása**

**3.3.1 BiteBlocker (Fonalvágást gátló spóra)**

* **Felelősség**

Ha a rovar a hatása alatt áll, akkor nem tud ketté harapni fonalat.

* **Ősosztályok**

Spore osztály > BiteBlocker osztály

* **Interfészek**

Nincs.

* **Asszociációk**

Nincs.

* **Attribútumok**

Nincs.

* **Metódusok**
  + **haveEffect(chosenAction: int)** – Kifejti a hatását a rovarra ami megette.

**3.3.2 Booster (Gyorsító spóra)**

* **Felelősség**

Ha a rovar a hatása alatt áll, akkor két tektont tud lépni.

* **Ősosztályok**

Spore osztály > Booster osztály

* **Interfészek**

Nincs.

* **Asszociációk**

Nincs.

* **Attribútumok**

Nincs.

* **Metódusok**
  + **haveEffect(chosenAction: int)** – Kifejti a hatását a rovarra ami megette.

**3.3.3** **Bugger (Rovarász)**

* **Felelősség**

Nyilvántartja a hatást, ami alatt éppen áll. (Megevett spóra típusától függ.) Mozog a térképen, spórákat eszik és fonalakat vág el.

* **Ősosztályok**

Player osztály > Bugger osztály

* **Interfészek**

Nincs.

* **Asszociációk**
  + **Tekton:** Kapcsolatban van azzal a tektonnal, amin éppen áll. (-position)
  + **Spore:** Kapcsolat a legutóbb megevett spórával. (-ateLast)
* **Attribútumok**
  + **underEffectSince(int) :** itt tartja nyilván, hogy mióta (hány köre) van hatás alatt.
* **Metódusok**
  + **executeRound(int roundNumber): void** – Végrehajtja a játékos lépését az adott körben, meghívja a selectAction metódusát.
  + **selectAction(): int** – Kiválasztja a végrehajtandó akciót a körben.
  + **eat(s: Spore)** – Egy adott spórát fogyaszt el.
  + **bite(h: Hypha)** – Egy adott gombafonalat vág el.
  + **move(t: Tekton)** – Egy szomszédos mezőre lép. Meg kell adni, hogy melyik tekton a cél.

**3.3.4** **Game (Játék)**

* **Felelősség**

Számolja a köröket és lépteti a játékosokat. A játékmenetet irányítja és a győztest határozza meg. FIFO módon lépteti a játékosokat.

* **Ősosztályok**

Nincs.

* **Interfészek**

Nincs.

* **Asszociációk**
  + **Player**: A Game tartalmazza a Player típusú objektumokat, minimum 4-et.
* **Attribútumok**
  + **round: int** – A jelenlegi kör száma.
  + **map: List<Tekton>** – Listában tárolja a tektonokat.
* **Metódusok**
  + **gameLoop(): void**  - Lefuttatja a játék fő ciklusát, amelyben a játékosok egy FIFO listába kerülnek és a gameLoop metódus lépteti őket sorban.

**3.3.5 Hypa (Gombafonál)**

* **Felelősség**

Kapcsolatokat tart fennt a tektonok között és köti össze őket.

* **Ősosztályok**

Nincs.

* **Interfészek**

Nincs.

* **Asszociációk**
  + **Tekton**: Referenciával rendelkezik a hozzá kapcsolódó tektonokra, ezek a fonál végei.
* **Attribútumok**
  + **age()** – Tárolja a fonál életkorát.
  + **isDying()** – Tárolja, hogy a fonál nem kötődik gombatesthez, és emiatt haldoklik.
* **Metódusok**
  + **die()** – Elhalt/elharapott fonál törlése.
  + **age()** – Shroomer hívja minden kör után, és növeli az életkorát.

**3.3.6  Mushroom (Gombatest) - Absztrakt**

* **Felelősség**

Nyilvántartja az élettartamát, és spórákat termel, valamint tárolja azokat.

* **Ősosztályok**

Nincs.

* **Interfészek**

Nincs.

* **Asszociációk**
  + **Tekton** – A gomba egy adott mezőn nő.
  + **Shroomer** – A gomba egy gombászhoz tartozik.
* **Attribútumok**
  + **age: int** – A gomba kora, amióta megszületett.
  + **numberOfSpores: int** – Spórák száma, amit el tud még szórni.
  + **sporesThrown: int** – Eddig eldobott spórák száma.
* **Metódusok**
  + **sporeThrown()** – Spóra eldobása. A tekton storeSpore függvényét hívja meg.
  + **die()** – Gomba meghalásakor hívja a Shroomer.
  + **age()** – Shroomer hívja minden kör után, így növeli az életkorát.

**3.3.7**   **MushroomFactoryRegistry**

* **Felelősség**

A különböző gombafajok létrehozásáért felel.

* **Ősosztályok**

Nincs.

* **Interfészek**

Nincs.

* **Asszociációk**
  + **Shroomer** – A Shroomer osztály elemeit példányosítja.
* **Attribútumok**
  + **registry: Map<String,Function>** – Map-ban tárolja a különböző gombafajokat(String) és a konstruktorokat(Function).
* **Metódusok**
  + **createMushroom(String, Shroomer, Tekton): Mushroom** – Példányosítja a paraméterként kapott gombát, és hívja a Mushroom konstruktorát.
  + **registerMushroom(type: String, constructor: Function)** – Új gombafaj felvételére szolgál.

**3.3.8 Paralyzer (Bénító spóra)**

* **Felelősség**

Ha a rovar a hatása alatt áll, akkor nem tud semmiféle akciót végrehajtani.

* **Ősosztályok**

Spore osztály > Paralyzer osztály

* **Interfészek**

Nincs.

* **Asszociációk**

Nincs.

* **Attribútumok**

Nincs.

* **Metódusok**
  + **haveEffect(chosenAction: int)** – Kifejti a hatását a rovarra ami megette.

**3.3.9**   **Peat (Tőzeg)**

* **Felelősség**

Több gombafonal is átmehet rajta.

* **Ősosztályok**

Tekton osztály > Peat osztály

* **Interfészek**

Nincs.

* **Asszociációk**

Nincs.

* **Attribútumok**

Nincs.

* **Metódusok**
  + **acceptHypa(shroomer: Shroomer)** – Több gombafonal is átmehet rajta, tehát nem bool.

**3.3.10**   **Player (Játékos) - Absztrakt**

* **Felelősség**

A játékos osztály reprezentálja a játékosokat, tárolja pontszámukat, és végrehajtja a körükhöz tartozó akciókat.

* **Ősosztályok**

Nincs.

* **Interfészek**

Nincs.

* **Asszociációk**
  + **Tekton**: A játékos aktuális pozíciója. Minden játékos egy adott mezőn áll.
  + **Game**: A játékhoz tartozik. Game osztály tartalmazza.
* **Attribútumok**
  + **score**: int – A játékos által gyűjtött pontok száma.
* **Metódusok**
  + **increaseScore(value: int)** – Növeli a játékos pontszámát a kapott értékkel.

**3.3.11**   **Shroomer (Gombász)**

* **Felelősség**

A gombász feladata a gombafonalak növesztése, spórák szórása és a pontszerzés.

* **Ősosztályok**

Player osztály > Shroomer osztály

* **Interfészek**

Nincs.

* **Asszociációk**
  + **Mushroom**:  Függőségben áll vele. Létrehozza a gombatesteket a throwSpore függvényével, ha elégszer hívódott meg. A gombász több gombát is kezelhet, több is tartozhat hozzá.
  + **Hypha**: Gombafonalakat növeszt a growHypa függvényével.
  + **Spore**: Spórákat szór a throwSpore függvényével.
  + **Tekton**: Függőségben áll vele. A gombász tekton mezőkön mozog.
* **Attribútumok**

Nincs.

* **Metódusok**
  + **growHypa(from: Tekton, to: Tekton)** – Gombafonal növesztése. Argumentumban meg kell adni, melyik tektonról hova növesztünk.
  + **throwSpore(with: Mushroom, to: Tekton)** – Spóra szórása. Meg kell adni, melyik gombatestből szórunk és melyik tektonra.
  + **increaseScore()** – Növeli a pontszámát az adott játékosnak, amelyik meghívja.
  + **executeRound(int roundNumber): void** – Végrehajtja a játékos lépését az adott körben, meghívja a selectAction metódusát. Ezt a Player osztályból örökli.
  + **selectAction(): int** – Kiválasztja a végrehajtandó akciót a körben. Ezt a Player osztályból örökli.

**3.3.12 Slower (Lassító spóra)**

* **Felelősség**

Ha a rovar a hatása alatt áll, akkor két kör alatt egy tektont tud csak lépni.

* **Ősosztályok**

Spore osztály > Slower osztály

* **Interfészek**

Nincs.

* **Asszociációk**

Nincs.

* **Attribútumok**

Nincs.

* **Metódusok**
  + **haveEffect(chosenAction: int)** – Kifejti a hatását a rovarra ami megette.

**3.3.13  Spore (Spóra) – Absztrakt**

* **Felelősség**

Különböző hatásokat vált ki a rovarászokra, és szabályozza az interakciókat.

* **Ősosztályok**

Nincs.

* **Interfészek**

Strategy

* **Asszociációk**
  + **Tekton**: A spóra egy adott mezőn van tárolva.
  + **Bugger**: A rovarász spórákat ehet meg. Ilyenkor a spóra haveEffect függvénye hívódik meg, így váltja ki a rovarra a hatását.
  + **Shroomer**: Minden spóra egy gombászhoz tartozik, akinek a gombája elszórta azt.
* **Attribútumok**

Nincs.

* **Metódusok**
  + **haveEffect()** – Aktiválja egy adott spóra hatását, meghívja egy spóratípus adott függvényét.

**3.3.14**   **Stone (Kő)**

* **Felelősség**

Gombatest nem nőhet rajta.

* **Ősosztályok**

Tekton osztály > Stone osztály

* **Interfészek**

Nincs.

* **Asszociációk**

Nincs.

* **Attribútumok**

Nincs.

* **Metódusok**
  + **growMushroom()** – Nem nőhet rá gomba.

**3.3.15**   **Swamp (Mocsár)**

* **Felelősség**

A gombafonalak egy idő után(3 kör) elhalnak rajta.

* **Ősosztályok**

Tekton osztály > Swamp osztály

* **Interfészek**

Nincs.

* **Asszociációk**

Nincs.

* **Attribútumok**

Nincs.

* **Metódusok**
  + **checkForDeleteHypa()** – Minden kör végén végigmegy a tárolt fonalak listáján, ha van 3 körös,  akkor töröljük.

**3.3.16**   **Tekton**

* **Felelősség**

A pálya alapeleme, amely tárolja a rajta lévő gombákat, spórákat, fonalakat, rovarokat, valamint a szomszédait. Meghatározza, milyen feltételekkel történhet rajta a növekedés és a kapcsolatépítés.

* **Ősosztályok**

Nincs.

* **Interfészek**

Nincs.

* **Asszociációk**
  + **Mushroom:** tárolja a rajta lévő gombatestet, ha van ilyen. Van erre lekérdező függvénye.
  + **Spore:** spórákat tárolja és ha összegyűlik elegendő, akkor gombává növeszti a growMushroom függvényével.
  + **Hypa:** Bármennyi gombafonal kapcsolódhat hozzá. A fonalak végpontja a tektonok.
  + **Bugger:** A pozícióját tárolja az adott rovarnak.
  + **Tekton:** Másik tekton objektumokkal szomszédságban áll.
* **Attribútumok**
  + **hasBug: bool** – Tárolja, hogy tartózkodik-e rajta rovar.
* **Metódusok**
  + **break()** – Meghívja a lekérdező függvényeket, hogy van-e rajta valami. Ha nincs rajta gombatest, akkor a mező kettétörik.
  + **hasMushroom()** – Ellenőrzi, van-e gomba a mezőn.
  + **<virtual>growMushroom()** – Gombát növeszt az adott tektonon, ami meghívja.
  + **storeSpore(s: Spore)** – Spóra tárolása.
  + **getNeighbours(): List<Tekton>** – Szomszédos mezők lekérése.
  + **removeSpore(s: Spore)** – Eltávolítja a spórákat.
  + **hasBug()** – Le lehet kérni, hogy van-e rajta rovar.
  + **<virtual>acceptHypa(shroomer: Shroomer): bool** – Gombafonál növésének elfogadása az adott tektonra.

**3.3.17**   **Strategy – Absztrakt Interfész**

* **Felelősség**

Megvalósítja a különböző viselkedéseket attól függően, hogy milyen hatás alatt áll a rovar.

* **Ősosztályok**

Nincs

* **Asszociációk**
  + **Spore** – A különböző spórák által kiváltott hatásokat implementálja.
* **Metódusok**
  + eat(s: Spore) – Spóra evés.
  + bite(h: Hypa) – Harapás.
  + move(t: Tekton) – Mozgás.

**3.4** **Szekvencia diagramok**

Bugger executeRound() szekvenciája fonal harapás akció kiválasztásakor

A képen szöveg, képernyőkép, diagram, sor látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

A nem BiteBlocked típusú stratégiák move() metódusának szekvenciája:

A képen szöveg, képernyőkép, sor, szám látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

A BiteBlocked típusú stratégiák move() metódusának szekvenciája:

A képen szöveg, képernyőkép, Betűtípus, sor látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

Bugger excdcuteRound() Spóra evés szekvencia

Shroomer gombafonál szórása:

A képen szöveg, képernyőkép, sor, diagram látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

 Gombász spóra szórása, fiatal gombatest esetében

A képen szöveg, képernyőkép, diagram, sor látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

Gombász spóra szórása, old gombatest esetébenA képen szöveg, képernyőkép, Párhuzamos, sor látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.**3.5** **State-chartok*A képen diagram, szöveg, vázlat, rajz látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.***

**3.6** **Napló**

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2025.02.25. 21:00 | 1,5 óra | Katona  Fekete  Lipóczi  Kőműves  Szabó | Értekezlet.  Döntés: Katona és Kőműves elkészíti az osztályidagrammokat 02.27-ig, Fekete elkezdi a szekvencia diagrammokat, Lipóczi  Szabó elkészíti az állapotgépeket, és végiggondolja a játék működését 02.27-ig |
| 2025.02.26 10:00 | 2 óra | Szabó | kezdetleges állapotgép elkészítése |
| 2025.02.26 10:00 | 1,5 óra | Fekete  Kőműves | konzultáción részvétel |
| 2025.02.26 12:00 | 2 óra | Katona | UML Class diagramm tervezés |
| 2025.02.26 18:00 | 4 óra | Kőműves | kezdetleges UML Class diagram tervezés |
| 2025.02.26 19:00 | 1,5 óra | Lipóczi | objektum katalógus elkészítése |
| 2025.02.27 16:00 | 2óra | Katona | Class Diagramm |
| 2025.02.27 12:00 | 2 óra | Katona  Fekete  Lipóczi  Kőműves  Szabó | Értekezlet  játék alapvető működéséhez kapcsolódó oo megvalósítások átbeszélése  Döntés: |
| 2025.02.27 16:00 | 2 óra | Lipóczi | Osztályok leírása |
| 2025.02.27 18:00 | 2 óra | Fekete | Osztályok leírása 2 |
| 2025.03.01. 16:00 | 1,5 óra | Fekete  Kőműves  Lipóczi  Szabó | Értekezlet  Végső osztály diagramm átbeszélése |
| 2025.03.02. 21:00 | 3 óra | Lipóczi | Szekvencia diagramok készítése |
| 2025.03.02. 21:00 | 3 óra | Kőműves | Szekvencia diagramok készítése |
| 2025.03.02 21:00 | 3 óra | Fekete | Osztályleírások frissítése, rendezése |
| 2025.03.02 21:00 | 3 óra | Katona | Szekvenciadiagrammok elkészítése |
| 2025.03.02 21:00 | 3 óra | Szabó | megadott szekvenciadiagrammok elkészítése |