

Fungorium

68 – nullpointerexception

Konzulens:

Simon Balázs

Csapattagok

Fórián-Szabó Bernát	ABIVEP	foriansiga@gmail.com
Zsigmond		
Gyárfás Réka	AFVLHM	gyarfas.reka@gmail.com
Kemecsei Kornél	HDB6X9	kornel.kemecsei@edu.bme.hu
Kuzmin Iván	U0725D	ikuzmin@edu.bme.hu
Georgijevics		
Tóth Mihály Balázs	OAYOF1	toth.misi05@gmail.com

2025.02.23

1. Követelmény, projekt, funkcionalitás

1.1 Bevezetés

1.1.1 Cél

A Fungorium egy többjátékos stratégiai játék, amelyben a játékosok különböző gombafajokat irányítva próbálják uralmuk alá vonni a rendelkezésre álló területet vagy rovarként szeretnének élelemhez jutni. A játék célja az adott környezetben való túlélés, terjeszkedés és a többi játékos legyőzése.

1.1.2 Szakterület

A szoftver egy offline többjátékos stratégiai játék, amely biológiai inspiráció alapján modellezi a gombák növekedési és terjedési mechanizmusait és a rovarok túlélési ösztöneit. A játék szórakoztató és oktatási célokra egyaránt használható, segíthet a természetes ökoszisztémák megértésében.

1.1.3 Definíciók, rövidítések

Fungorium: A játék neve

NPC: Nem játékos karakter, amelyet a játék rendszere irányít.

UI: Felhasználói felület, amelyen keresztül a játékos interakcióba léphet a játékkal.

Tárgyhonlap: Hivatkozás [1]

1.1.4 Hivatkozások

<https://www.iit.bme.hu/targyak/BMEVIIIAB02>

<https://docs.oracle.com/en/java/>

<https://docs.oracle.com/en/java/javase/22/docs/api/java.desktop/javafx/swing/package-summary.html>

https://github.com/Harcipan/Projlab_nullpointerexception

2. Összefoglalás

A dokumentum részletezi a játék célját, a funkcionalitásokat, a felhasználói jellemzőket, valamint a rendszer követelményeit. Bemutatja a projekttervet, a követelményeket, a felhasználói eseteket és egy szójegyzéket is tartalmaz.

2.1 Áttekintés

2.1.1 Általános áttekintés

A Fungorium játék egy többjátékos módban játszható kalandjáték. A játékosok lokálisan játszhatnak.

2.1.2 Funkciók

Főmenü

A játék indításakor a felhasználó egy főmenüt lát, az alábbi lehetőségekkel:

New Game (új játék)

Load Game (megkezdett állás betöltése)

Exit Game (kilépés)

New Game: A felhasználó megadja a térkép méretét és a mentés nevét. Ezután egyesével hozzáadhatók a játékosok: Megadja a nevét, választ, hogy rovarász vagy gombász szeretne lenni, majd jelölő szint választ (ez segít a játék közben felírsmerni az irányított egységet). Ezt követően be lehet állítani a térkép méretét illetve a maximális körök számát (lehet végtelen is), majd elindítani a játékmenetet. Minimum 2-2 gombász és rovarász szükséges egy játékhoz!

Load Game: A felhasználó a megjelenő listában látja a korábbi megkezdett játékait, ezek közül választ egyet és azt elindítja. Van lehetőség a mentés törlésére is, ha már nem akarja folytatni.

Exit Game: Kilép a programból.

Játék menete:

Sorsolás: A játékosok között véletlenszerűen meghatározott sorrend szerint haladunk, mindig először a gombák jönnek és csak utánuk a rovarok.

Játék felület: A térképet látja a felhasználó, illetve egyéb információkat úgy mint a kör sorszáma, a soron levő játékos neve, stb. A térkép tektonokból áll, ezek pedig kisebb mezőkből.

Kezdő kör: Az első gomba kiválasztja a kezdőmezőt, ide lerak egy gombatestet. A többi gomba hasonlóan tesz, figyelembe véve, hogy egy tektonon csak egy élő gomba lehet. A rovarok is a sorrendnek megfelelően kezdőhelyet választanak és oda lerakják a rovarjukat, itt nincs megkötés az egy tektonon tartózkodó rovarok számára.

Többi kör:

Gombák:

Fonál növesztés: A gombák körönként fix mennyiségű növekedési pontot kapnak, ezt tudják fonalak növesztésére felhasználni. Különböző típusú tektonokon, illetve annak függvényében, hogy van-e spóránk a tektonon, más költsége van ennek a műveletnek. A fonál növesztése a mezők kijelölésével történik. Csak olyan helyre lehet fonalat elhelyezni, ahol az fonalakon keresztül vagy közvetlenül egy saját gombatesthez kapcsolódik.

Spóra szórás: A gombatestek spóra pontokat termelnek minden körben, elegendő mennyiség összegyűlése után spóra szórás kezdeményezhet a játékos: A kis szórás kevesebb pontba kerül, viszont kisebb mennyiségű spórát juttat a térképre, és ezek csak szomszédos tektonokra

jutnak el. A nagy szórás több spórát juttat ki, és ezek a szomszédok szomszédjaira is eljutnak. Egy gombatest életében véges alkalommal tud spórát szórni, ezután elpusztul.

Gombatest növesztés: Ha a fonalak egy új tektonra érnek, ahol még nincs másik gombatest és elegendő spóra van a tektonon, lehetőség van gombatest növesztésére. Ez elhasználja a tektonon lévő spórákból egy mennyiséget. Gombatest is csak élő fonalakkal szomszédos mezőkre rakható.

Halott fonál: Ha egy fonál és a gombatest között megszűnik az összeköttetés akkor elkezd elhalni. Ilyen „rothadó” fonálból nem lehet tovább építkezni és gombatestet se hozhat létre. Még a teljes elhalás előtt „meg lehet menteni” a fonalat, ha egy élő fonállal elérjük.

Rovarok:

Mozgás: A rovarok körönként mozgási pontokat kapnak, ezeket felhasználva haladhatnak a mezőkön keresztül. Tektonok határát csak ott léphetik át, ahol gombafonalak mennek keresztül.

Spóra evés: A rovarok célja a spórák elfogyasztása. Különböző típusú spóráknak más-más tápértéke van, illetve módosító értékeik is lehetnek: Van ami gyorsítja a rovar, van ami lassítja, van ami megbénítja, illetve van olyan ami a fonalvágási képességét elveszi. Ezeknek a hatásoknak körökben meghatározott időtartama van. A rovar azokat a spórákat tudja elfogyasztani, ami vele azonos mezőn található.

Fonál vágás: A rovar elvághat egy tektonok között húzódó fonalat, ezzel hátráltatva más rovarokat illetve gombákat. A vágáshoz a rovarnak azon mezők egyikén kell tartózkodnia, amelyek a „híd” fonál valamely oldalát képezik.

A játék szabályai:

A rovarok csak a gombafonalakon közlekedhetnek, még a tekton belsejében is.

Ha egy rovar ott van, ahol a tekton éppen félbevágódik vagy más megszakítja a fonalat alatta, az befolyásolja a mozgását.

Játék vége:

Lejár az idő: A kezdéskor beállított számú körök után automatikusan véget ér a játék. Az utolsó játékos utolsó akciója után kiértékelődnek az eredmények, a rovarok és gombák közt külön-külön rangsort állítunk fel.

Kihal egy gomba: Egy gomba játékos ha elveszíti az összes gombatestjét, és nem tud sehová újat létrehozni, akkor ezzel „kiahlt”. A pontszámok kiértékelődnek, a rangsorokat megmutatja a játék. Már játékos halála is a játék végét eredményezi, ez összetettebb stratégiáknak ad helyet.

Kiértékelés kérése: A játékot meg lehet állítani, az ekkor megjelenő képernyőn kérhetjük a jelen állás szerinti kiértékelést. Ekkor a játék az aktuális állás alapján kiszámolja az eredményt és felállítja a rangsort.

Pontszámítás: A rovarok pontjai az elfogyasztott spórákból számolódnak: Minden spórának van egy tápanyag értéke, ebből kell a rovarnak a játék során minél többet összegyűjteni. A gombák pontjait a játék során létrehozott gombatestek száma adja, tehát nem a játék végén élő testek, hanem a valaha létrehozottak számítanak.

2.1.3 Felhasználók

A játék célcsoportja a fiatal felnőttek és az idősebb játékosok, akik érdeklődnek a természet és a kalandjátékok iránt továbbá a játék stratégiai részét is értelmezni képesek. Rendelkeznek az alapvető digitális írástudással, mint az egérhasználat, operációs rendszerben való navigálás és alkalmazások telepítése.

Felhasználói interakciók:

Főmenü

- New Game

- Load Game
- Exit Game

Játékmenet

- Kezdőhely választása
- Gomba
 - Fonál növesztés
 - Spóra szórás
 - Gombatest növesztés
- Rovar
 - Mozgás
 - Spóra evés
 - Fonál vágás

2.1.4 Korlátozások

A játék windows 10 és windows 11 rendszereken futtatható.

2.1.5 3Feltételezések, kapcsolatok

A feltételek a Tárgyhonlap weboldalról származnak.

2.2 Követelmények

2.2.1 Funkcionális követelmények

Azono-sító	Leírás	Ellenőrzés	Prio-ritás	Forrás	Use-case	Kom-ment
FUN001	'Új játék' gomb egy friss játékot indít el egy random pályán.	Megnyomjuk az 'Új játék' gombot.	MUST	nullpointerexception	Főmenü használat	
gfdgfd						

2.2.2 Erőforrásokkal kapcsolatos követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
ERO01	Java 22 runtime	pom.xml-ben ellenőrizendő a maven.compiler.*	MUST	nullpointerexception	

ERO02	Futtatható állapotban van-e a játék?	https://niif.cloud.bme.hu/ és https://fured.cloud.bme.hu/ VM-ekre feltölteni a játékot és ellenőrizni, hogy hibátlanul fut-e a játék.	MUST	nullpointerexception	
ERO03	Terjeszthető állapotban van-e a játék?	Minden erőforrás egy (fájl)könyvtárban van.	MUST	nullpointerexception	Ez úgy értelmezendő, hogy ahelyett, hogy a játékhoz szükséges erőforrások (textúrák, font-ok stb.) szét lennének szóródva az AppData-ban, temp-ben, meg egyéb máshol, helyette csak egy helyen lennének: a helyi könyvtárban. Magyarul bezippelhető és megosztható formátumban van a játék
ERO04	A programkód egy egységes archívumban van-e?	Megnézni, hogy a Maven build JAR-t ad ki	MUST	nullpointerexception	Formaiság.

ERO05	Nincsen külső könyvtár linkelve	JAR kimenetben megnézni, hogy a saját osztályainkon (és az azokba beágyazottokon) kívül van-e más olyan dolog, ami külső könyvtár.	MUST	nullpointerexception	Magyarul: milyen Maven dependency-k kerülnek be a végső JAR-ba. A pom.xml is ellenőrizhető, de az átláthatatlansága miatt nem igazán jó megoldás.
ERO06	Az összes forrásfájl egy közös GitHub repository-n van.	Nincsen olyan közös, projekthez fontos fájl, ami nem lenne a közös GitHub repository-n	MUST	nullpointerexception	
ERO07	A felmerülő problémák egy közös Trello-ban követhetőek.	-	MUST	nullpointerexception	
ERO08	Messenger-en a csapattagok egymásnak üzenhetnek.	-	MUST	nullpointerexception	
ERO09	A napló helyesen van-e kitöltve	A naplóban a névben vezetéknév van, az időtartam kb. megfelel, a dátum is helyes (még ha később is van feltöltve az adott munka a GitHub-ra)	MUST	nullpointerexception	
ERO10	A dokumentációkban ízlésesek, megfelelő stílusok vannak alkalmazva, nincsenek többsoros paragrafusok stb.	Esettől függ.	SHOULD	nullpointerexception	A lényeg, hogy a dokumentáció ne nézzen ki úgy, mint egy labor jegyzőkönyv, hanem úgy, mint egy LaTeX-ben megírt jegyzet.

ERO11	A GitHub repo-n követhető az összes csapattag munkája.	Mindenki a hivatalos nevét használja a Git név mezőben a commit-okban.	MUST	nullpointerexception	Hogy a csapattagok n kívül is lássák, hogy ki mit csinált.
ERO12	A GitHub flow alkalmazása	A különböző feature branch-ek elágaznak (tehát nem master-ba kerül a legújabb változtatás, hanem egy új branch-be), és merge/rebase helyett GitHub pull request felrakása	MUST	nullpointerexception	A merge konfliktusok minimalizálása és push konfliktusok elkerülése érdekében ez itt van.
ERO13	Számítógép erőforrások	Memória: min. 4 GB Tárhely: min. 20 GB Processzor: min. 64 bites 2 Ghz-es 2 magos	MUST	nullpointerexception	
ERO14	A dokumentációkat Google Docs-on keresztül vannak szerkesztve	-	MUST	nullpointerexception	

2.2.3 Átadással kapcsolatos követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
Átad1	A telepítés egy .exe file futtatásával történik.	Az .exe futtatásával elindul egy telepítő mely egy adott telepítési helyre létrehozza a futtatható állományt.	SHOULD	nullpointerexception	

Átad2	Az alkalmazás a .jar file-al használható	A .jar file-ra való dupla kattintással elindul az alkalmazás	SHOULD	nullpointerexception	

2.2.4 Egyéb nem funkcionális követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
NF001	Könnyen lehegyen használható	Empirikusan. Játszunk a játékkal és ha nem találunk ellentmondást elfogadjuk.	MUST	nullpointerexception	
NF002	Legyenek olvashatók a színei	Empirikusan. Játszunk a játékkal és ha nem találunk ellentmondást elfogadjuk.	MUST	nullpointerexception	
NF003	Legyenek játékosok elkülöníthetők	Megnézzük, hogy más színűek-e a játékosok.	MUST	nullpointerexception	

2.3 Lényeges use-case-ek

2.3.1 Use-case leírások

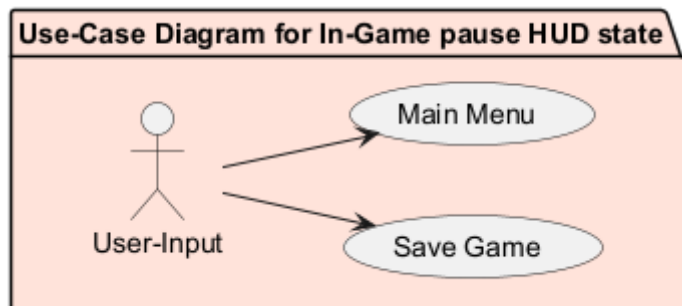
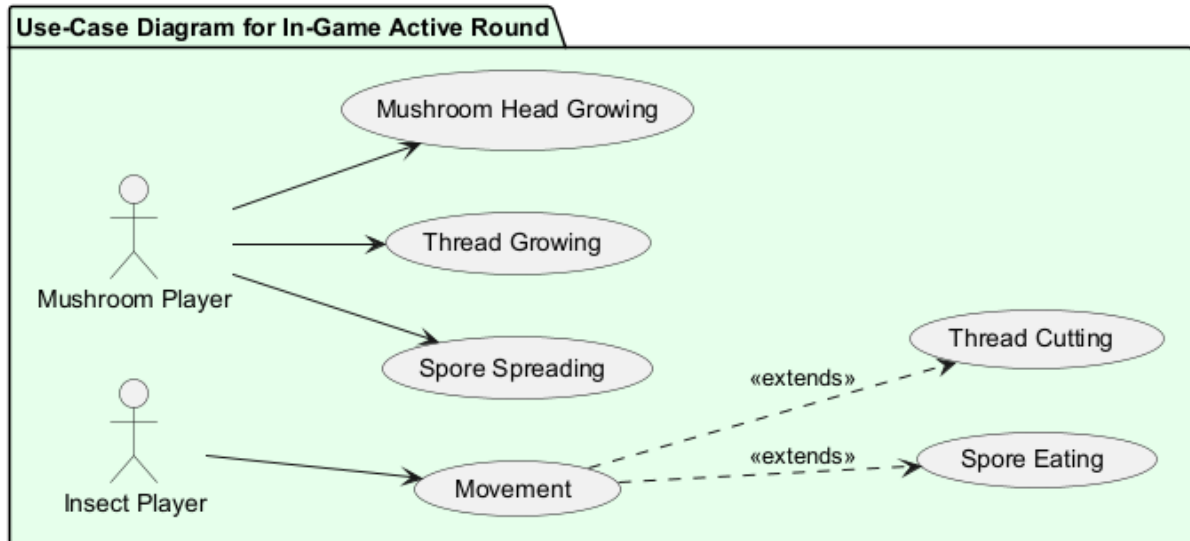
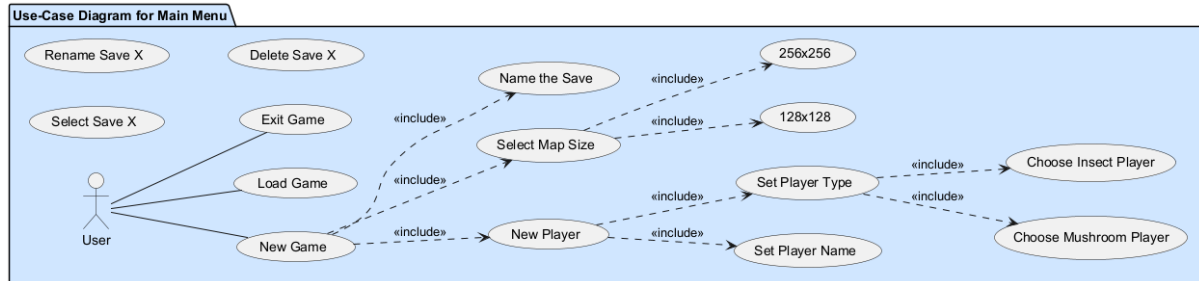
Use-case neve	Főmenü használata
Rövid leírás	Az új játék, játék betöltése és kilépés gombokkal navigálhatunk új alkalmazásfelületekre
Aktorok	játék indítója
Forgatókönyv	Valemennyi gomb megnyomása a jelenlegi ablakot eltünteti és egy másik felületet hoz elő.

Use-case neve	Játék egy körben
Rövid leírás	A játékosok közül mindenki sorra kerül és elvégzi a számára engedélyezett műveleteket.
Aktorok	játékosok (gombok, rovarok)
Forgatókönyv	A gomba növeszti fonalát a rovar pedig eszik.

Use-case neve	Pause menü használata
Rövid leírás	A játék megállítása után az általunk kiválasztott akciót tudjuk végrehajtani.

Aktorok	Felhasználó aki az egeret és billentyűzetet kezeli
Forgatókönyv	A játékot megállítjuk az 'ESC' gombbal majd a 'Main menu' gombra nyomunk ami visszavisz minket a főmenü oldalára

2.4 Use-case diagram



2.5 Szótár

Fonál – A gombák növekedésének egyik eleme, amely lehetővé teszi a terjedést és az új gombatestek létrehozását.

Fonálvágás – A rovarok azon képessége, amely lehetővé teszi számukra a gombák fonalainak megszakítását, ezáltal hátráltatva a gombák növekedését és a többi rovar mozgását.

Fungorium – A játék neve, egy többjátékos stratégiai játék, amelyben gombák és rovarok versengenek egymással.

Gombatest – A gombák által létrehozott alapvető egység, amelyből növekedés és terjeszkedés indulhat.

Mozgási pont – A rovarok mozgásához szükséges erőforrás, amely körönként megújul.

Rovar – A másik játszható frakció a játékban, amely a gombák spóráit fogyasztja és mozgásával, fonálvágással befolyásolja a játékmenetet.

Spóra – A gombák által termelt egység, amely a terjedésükhöz szükséges. A játékban a spórák összegyűlhetnek, majd a gombák felhasználhatják őket terjeszkedésre.

Tekton – A térkép egy egysége, amely kisebb mezőkből áll.

Fonálvágás – A rovarok azon képessége, amely lehetővé teszi számukra a gombák fonalainak megszakítását, ezáltal hátráltatva a gombák növekedését és a többi rovar mozgását.

Csempe - Tektonok kisebb egysége, ezeken mozognak a rovarok, fonalak.

2.6 Projekt terv

1. Lépés Követelmények, modellek elkészítése
2. Lépés: Prototípus elkészítése
3. Lépés: GUI elkészítése
4. Lépés: Tesztelés

A feladatok kezelése állapotaik nyilvántartása a trello felületén történik. Mindenkinek van saját kanban táblája amin az éppen aktuális munkafolyamatok látszódnak. Ezen felül van egy központi/közös tábla a még nem kiosztott vagy megbeszélést igénylő munkákhoz.

A kód verziókezelését a git teszi lehetővé melyet a github szolgáltatásán keresztül elérhető és szerkeszthető a csapattagok számára.

Az általános kommunikáció a discord alkalmazásban történik. Minden munkafolyamat külön thread-eken.

2.7 Napló

Kezdet	Időtartam	Résztevők	Leírás
2010.03.10. 18:00	1 óra	Fórián Gyárfás Kmecsei Kuzmin Tóth	Döntések: Csapatalakítás, csapatvezető, csapatnév választás. Meeting időpont megbeszélése, feladatlap átadó kijelölése, konzulens-el értekező kijelölése
2010.03.23. 23:00	2 óra	Fórián Kmecsei Tóth	Döntések: Alapvető programműködés megbeszélése, feladatkiírás értelmezése. (tektonok bontásának és tárolásának elve)

2025.02.15. 14:00	2 óra	Fórián Gyárfás Kemecsei Kuzmin Tóth	Döntések: Alapvető programműködés megbeszélése, tervezési kérdések megvitatása. (pl: Happy path elkészítése, spóraszórás elve)
2025.02.15. 14:00	2 óra	Fórián	Tevékenység: Meeting log elkészítése összefésülése
2025.02.15. 14:00	2 óra	Tóth	Tevékenység: Meeting log elkészítése összefésülése
2025.02.16.	2 óra	Gyárfás	Tevékenység: Use-Cases megtervezés
2025.02.16	2 óra	Fórián	Tevékenység: Use-Cases PlantUML-be
2025.02.16. 19:30	2 óra	Kemecsei	Tevékenység: Követelmények kitöltése
2025.02.18. 14:00	2 óra	Tóth	Tevékenység: Funkciók kitöltése
2025.02.18. 19:00	1 óra	Kuzmin	Tevékenység: erőforrások kitöltése
2025.02.19 19:00	30 perc	Forian	Tevékenység: feedback #1 transzkript, kiemelek, kommentek
2025.02.22 13:00	1 óra	Kemecsei	Tevékenység: Követelmények kitöltése
2025.02.22 14:00	45 perc	Fórián Gyárfás Kemecsei Kuzmin Tóth	Meeting #2 Tevékenységek: Követelmények kitöltésének befejezése Döntések: Kornél: Objektumkatalógus, dokumentálás Zsiga osztálydiagram Réka: állapot Iván: Szekvencia Misi: osztályleírás
2025.02.23. 13:10	30 perc	Kemecsei	Tevékenység: követelmények formázása, verziókezelése és nyomtatása