**Követelmény, projekt, funkcionalitás**

55 – ubr

Konzulens:

Potyók Csaba

Csapattagok

| Bogár Dániel | I07XOI | bogardanischool@gmailcom |
| --- | --- | --- |
| Koleszár Gábor | GSQ9GD | kolesz.kole@gmail.com |
| Jánovszki András | GUFP6A | janovszki84@gmail.com |
| Zombori Péter  Varjú Zalán Péter | DSYU1P  P2938R | zombori50@gmail.com  VarjZalan@gmail.com |

2024.02.25.

1. **Követelmény, projekt, funkcionalitás**
   1. ***Bevezetés***
      1. **Cél**

E dokumentum célja a szoftver projekt laboratórium nevű tárgy során elkészítendő feladat ismertetése, a projekt követelményeinek pontosítása, meghatározása, illetve a munkafolyamat során előkerülő alapfogalmak definiálása.

* + 1. **Szakterület**

A kiadott feladat egy játék, így fő célja az emberek szórakoztatása. Remek lehetőséget nyújt a stressz feloldásához. Bárkinek módjában áll a mentális „kikapcsolódás”, ebből kifolyólag bármilyen korosztály megtalálja majd benne az örömöt.

* + 1. **Definíciók, rövidítések**
* **Use-case:** Használati eset vagy esemény, amely egy konkrét szerepkörhöz tartozik.
* **Játékos:** Minden felhasználó egyetlen játékosnak számít. Meghatározott karaktereket irányíthatnak a játékon belül.
* **Kör:** A játékosok egymás után következnek a játékmenet során. Egy menet egyenlő azzal, amikor az első játékostól kezdve sorban végig haladunk a többin, és ismét a kiválasztotthoz jutunk. Más szóval, ha minden játékos és az összes gép által irányított karakter sorrakerült, akkor vége egy körnek és új kezdődik.
* **Mező:** A karakterek irányításával bejárható pontok a játék pályáján.
* **Tárgy/Item:** A játékmenet során a mezőkön található, felvehető, lerakható és használható eszközök. Mindegyik más-más képességgel és effekttel ellátott.
* **Tároló/Inventory:** Egy olyan táblázat, amelyen 5 hely található. Minden játékos rendelkezik vele és amikor ő kerül sorra, akkor a saját akarata szerint tehet bele, illetve vehet ki belőle tárgyakat. Egy tárgy 1 db. helyet foglal. A meghatározott maximum 5 helyet nem tudják túllépni a tárgyak. Így abban az esetben ha már 5 item-mel rendelkezik a játékos és szeretne még felvenni, akkor ki kell dobnia a tárolóból valamennyit.
* **Tárgyfelvétel:** Amikor a játékos egy olyan mezőre lép, ahol tárgyak találhatóak, akkor ezeket a tárgyakat elhelyezheti a saját tárolójában, ha éppen ő került sorra. Egy item elhelyezése a tárolóban egyenlő egy felvétellel.
* **Tárgy lerakás:** Ha a játékos sorra került, akkor bármely tárgyat kiveheti a tárolójából, ezzel pedig elhelyezi azon a mezőn, amelyen éppen tartózkodik.
* **Tárgy használata:** Ha a játékos felvesz egy tárgyat, alkalmazhatja annak képességét és effektjeit.
* **Action:** A karakterek által végezhető tevékenységek, amikor éppen ők kerültek sorra. Ilyen tevékenység egy db. mező lépése, egy tárgy felvétele egy mezőről, egy tárgy lerakása egy mezőre, egy tárgy használata, idle. Az egy körben végrehajtható tevékenységek számát egy, a játékmenet során, adott intervallumon belül folyamatosan változó számérték szabályozza. Minden körben, amikor a játékos sorra kerül véletlenszerűen rendelődik hozzá. Ennek módja egy hagyományos dobókocka eldobása a játék világán belül.
* **Idle:** Ha a játékos sorrakerült és az idle nevezetű action-t választja, akkor eltekint a maradék lehetőségétől bármilyen action végrehajtására és az utána lévő karakter kerül sorra. Action-t már csak a következő körben hajthat végre.
* **Szoba:** Egy szoba egyetlen darab mezőnek tekintendő. Minden szoba rendelkezik egy konkrét számmal, amelynél több játékos nem tartózkodhat egyszerre a mezőn. Ezt a számot befogadóképességnek nevezzük. Ezzel az értékkel egyenlő limit vonatkozik a szobában lévő tárgyakra is.
* **Ajtó:** Egyik szobából egy másikba történő lépés csak ún. ajtókon keresztül történhet meg. Tehát akkor létezik ajtó egyik szobából a másikba, ha átléphetünk egyik mezőről a másikra közvetlenül. Minden szobának van legalább egy ajtaja, viszont felső korlát nincs az ajtók számára. Az ajtó lehet egy- és kétirányú. Egyirányú, ha egyik szobából átléphetünk a másikba egyetlen action felhasználásával, de vissza már nem léphetünk oda, ahonnan jöttünk egy action használatával. Kétirányú, ha a két szoba között oda-vissza léphetünk másik mezőkre való lépés nélkül.
* **Algoritmus:** Megengedett lépésekből álló módszer, utasítássorozat, részletes útmutatás, amely valamely felmerült probléma megoldására alkalmas.
* **Gráf:** A gráf a matematikai gráfelmélet és a számítógép tudomány egyik alapvető fogalma. A gráf csomópontok, csúcsok és a rajtuk értelmezett összeköttetések, az az élek halmaza.
* **User:** Az adott szoftver felhasználója.
* **GUI (Graphical User Interface):** Egy program használata során a felhasználó által látott és használt grafikus elemekből összeálló felület. Ennek segítségével a user könnyebben kapcsolatba léphet a szoftverrel.
* **Swing:** Olyan számítástechnikai értelemben vett könyvtár, amely lehetővé teszi a GUI megvalósítását.
* **OOP (object-oriented programming):** Objektumorientált vagy objektumelvű programozás.
* **Szobák egyesülése:** Két szoba egyesülése az az esemény, amikor két darab szobából egy lesz. Ebben az esetben a nagyobb befogadóképességű szoba befogadási határa lesz érvényben a létrejött mezőn. Ha az egyik gázos volt, akkor a létrejött is az lesz. Az egyesülés során kedvező esetben a két szobában lévő tárgyak mind megmaradnak. Ha a tárgyak száma túllépné a item-ek számára vonatkozó limitet, akkor véletlenszerűen megsemmisül a határt meghaladó számú tárgy. A logarléc nevezetű tárgy nem semmisül meg.
* **Szoba osztódása:** Egy szoba osztódása az az esemény, amikor egy darab szobából kettő lesz. Ilyenkor a létrejött második szoba örökli a másik tulajdonságait. A szomszédos szobák, illetve a tárgyak számának a fele az új szobához kerülnek. A két szoba között kétirányú ajtó jön létre.
  + 1. **Hivatkozások**
* **IIT szoftver projekt laboratórium weboldala:** <https://www.iit.bme.hu/oktatas/tanszeki_targyak/BMEVIIIAB02?language=hu>
* **BSz2 tantárgy jegyzete:**

<https://cs.bme.hu/bsz2/bsz2_jegyzet.pdf>

* + 1. **Összefoglalás**

A dokumentum további szakasza leírja, ezáltal meghatározza a projekt konkrét követelményeit. Általános áttekintést nyújt az elkészítendő feladatról. Bemutatja az elkészítési munkafolyamat ütemtervét. Deklarálja a use-case-eket, illetve ismerteti a szoftver felhasználóit, környezetét és korlátait.

* 1. ***Áttekintés***
     1. **Általános áttekintés**

Az elkészítendő szoftver alap megalkotási elve az OOP. Ezt az elvet a Java programozási nyelv segítségével valósítjuk majd meg. A játékhoz szükséges pályák gráfok lesznek, amelyeket fájlból olvas majd be a program. A gráfok kezelését, illetve átalakítását gráfalgoritmusokkal és az azokra vonatkozó matematikai szabályok felhasználásával fogjuk végrehajtani. A GUI fejlesztését a Swing alkalmazásával kivitelezzük. A program parancssorból futtatható lesz. A játékosok a karaktereket az egér segítségével irányíthatják majd.

* + 1. **Funkciók**

A program alap koncepciója a tanárok és a diákok egymás ellen versengése, amelyben a diákok célja a logarléc nevű tárgy megszerzése. A tanárok is játékosnak számítanak, azonban őket egy primitív mesterséges intelligencia irányítja, amely mindig “dob” egyet, vagyis kap egy random számot 1 és 6 között és ennyi akciót kap. A játékban 2-5 ember játszhat egyszerre és a tanárok száma is ennek függvényében alakul, mivel mindig 1-el több a tanár, mint a játékos. A játék a játékosok függvényében előre meghatározott körből áll és ha végig ért az összes kör és a diákok nem szerezték meg a logarlécet, akkor a tanárok nyertek. A játékban fel lehet venni bizonyos tárgyakat, amit mind a tanárok, mind a diákok megtehetnek, azonban, csak a diákok használhatják ezeket a tárgyakat. A játékban egyszerre csak egy előre meghatározott mennyiségű tárgy lehet.

A játékban minden kör, úgynevezett “Action”-okből épül fel, amelynek számát a kockával való dobás határoz meg. Az Actionnek számító események a tárgyfelvétel, a tárgyleadás, tárgy felhasználása, és az IDLE, amely lényegében, annyit jelent, hogy nem szeretnéd felhasználni a maradék Action-ödet (A tanárok esetében, csak a lépés számít action-nek).

A játék pályája szobákból épül fel. Minden szobának van egy előre meghatározott létszám limitje, amely megadja, hogy abban a szobában, hány játékos tartózkodhat egyszerre, illetve minden szoba rendelkezik egy előre megadott tárgy limittel is, amely megadja, hogy a szobában, mennyi tárgy lehet egyszerre. A létszámlimit egy random választott szám 1 és a diákok száma-1 között. A tárgy limit megegyezik a létszámlimit-tel. A szobákból speciális fajta a gázos szoba, ahova a diákok belépve elejtik az összes náluk lévő itemet. A szobák képesek osztódni, illetve egyesülni. Ha egy diák egy szobába kerül egy tanárral, vagyis egy mezőn állnak, akkor a diák meghal, vagyis kiesik a játékból. Ez ellen az esemény ellen bizonyos tárgyak képesek biztonságot nyújtani. A szobák között úgynevezett ajtók vezetnek, amelyek azt jelzik, hogy egy adott szobából, melyik másik szobába tudsz átjárni, azonban vannak olyan ajtók, amelyek csak egy irányúak. Minden szobából át lehet menni egy másik szobába, vagyis nincsenek, olyan szobák, ahonnan nem lehet kijutni.

A játékban számos tárgy megtalálható, amelyek vagy passzívan, vagyis felhasználás nélkül a felvétel után automatikusan aktiválódnak, vagy aktívan, vagyis csak felhasználás után segítik a diákokat, illetve hátráltatják az oktatókat. Minden diák rendelkezik egy inventory-val, ennek 5 férőhelye van és a felhasználható tárgyak vannak it eltárolva. Minden tárgy külön inventory helyet foglal el, akkor is ha 2 ugyanolyan tárgyal rendelkezel. Egy tárgyat a vele egy mezőre lépés után tudunk felvenni. Ha egy oktató felvesz egy tárgyat, vagyis egy mezőre kerül vele, akkor a tárgy eltörlődik, vagyis végleg eltűnik a játékból.

A “TVSZ” nevű tárgy egy passzív hatású tárgy, ha egy oktató egy mezőre kerül veled, akkor nem esel ki, azonban ez csak 3-szor lehetséges, utána eltűnik rólad a TVSZ védelme és ismét sebezhető leszel. Fontos kiemelni, hogy ha a tanár mellénk kerül, a védelem eggyel csökken, azonban, ha következő körben, amikor ismét mi jövünk még mindig azon a helyen állunk és a tanár is ott áll akkor ismét elvesz egy TVSZ védelmet. Ha egy TVSZ-el már rendelkezünk és felveszünk még egyet, akkor felvett TVSZ a nálunk lévőt vissza tölti annyival, amennyi védelemmel a felvett TVSZ rendelkezik, azonban a maximum töltöttség 3, tehát ha a nálunk lévő TVSZ 3 védelemmel rendelkezik, akkor, ha felveszünk még egy hármasat, semmi nem fog változni.

A “Söröspohár” egy aktív tárgy, vagyis el kell használnunk, hogy legyen hatása. Ez a tárgy egy körig érvényes és a felhasználás utáni körben megvéd, ha egy tanár belép egy szobába veled. Egyszerre fel lehet többet is használni, azonban ez nem növeli a védelemben töltött körök számát, így ez nem ajánlott.

A “Nedves Rongy” egy aktív tárgy, vagyis fel kell használnunk, hogy legyen hatása. Ha van nálad nedves rongy tárgy és leteszed valahova, akkor aki abban a szobában van, az kimarad egy körből. A nedves rongy a játékosnál léte alatt fogy a nedvességéből, ezért érdemes, minél hamarabb kijátszani, mivel, ha kiszárad, akkor elveszti az erejét és egy sima rongy lesz. Ha több nedves rongy tárgy van a játékosnál azok egymástól függetlenül egyszerre veszítenek a nedvességükből.

A “Dobókocka” nevű tárgy szintén egy aktív tárgy, amelyet ha elhasználsz az action-eid alatt, akkor dobhatsz még egyet a kockával és a dobott számú action hozzáadódik a meglévő action-einkhez. Fontos, hogy ennek a tárgynak az elhasználása is action-be kerül.

Az “FFP2 maszk” is egy aktív tárgy, amelyet, azonban ha elhasználunk, akkor úgymond felvesszük az arcunkra és eltűnik az inventoryból, azonban ettől nem fog elhasználódni. A maszk, akkor használódik, ha gázos szobába lépünk, ekkor pedig 2 ilyen szoba hatásától képes megvédeni.

A “Camambert” tárgy is egy aktív tárgy, amelyet, ha elhasználunk, akkor lerakjuk a szobában, ahol vagyunk és miután elvégezted az összes action-ödet aktiválódik és gázos szobát csinál a szobából, illetve minden itt lévő játékos 2 körre lefagy, vagyis 2 körből kimarad és az összes náluk lévő tárgy eldobódik.

A “Tranzisztor” egy speciális tárgy, amelyből eldobva, mindig csak felet találunk és ha felveszünk 2 felet, akkor a nálunk lévő egyedüli tranzisztorral automatikusan összekapcsolódnak. Ha leteszed a 2 fél tranzisztort 2 külön szobába, akkor letéve is össze maradnak kapcsolva és miután bekapcsolod őket át tudsz jutni a az egyik tranzisztor szobájából a másikéba. Minden utazás előtt be kell kapcsolni a tranzisztorokat, ezért egy utazás 2 Action-t igényel. A letett tranzisztorokat, bármely diák használhatja. Fontos, hogy a tanárok nem használhatnak tranzisztorokat.

A “Logarléc” nevű tárgy a legfontosabb tárgy a játékban, előre nem lehet tudni, hogy melyik szobában van csak akkor amikor egy szobába kerülsz vele, mivel ekkor a diákok megnyerik a játékot. A logarléc az egyetlen tárgy, amelyet a tanárok nem tudnak felvenni.

* + 1. **Felhasználók**

A szoftvert bármely nem és korosztály használhatja. Feltéve, hogy rendelkezik a szükséges elektronikai eszközökkel.

* + 1. **Korlátozások**

A szoftver csak számítógépen lesz használható. Csak akkor képes a felhasználó játszani, ha rendelkezik egérrel. Csak többjátékos móddal rendelkezik majd a játék.

* + 1. **Feltételezések, kapcsolatok**

A hivatkozásokban található IIT weboldalon található információk adják az elkészítendő szoftver alap követelményeit és ötleteit. Deklarálják, hogy mik azok az alap elemek, amelyekkel rendelkeznie kell majd a játéknak. A weboldal előírja a program fejlesztésének ütemtervét. A másik hivatkozás a Bevezetés a Számításelméletbe 2 nevezetű tárgy jegyzete. Ez a jegyzet segíti majd a pálya megalkotását, ezt követően pedig annak folyamatos formálását, illetve átalakítását. A jegyzetből áttekinthetőek az efféle komplex feladatok megvalósításához szükséges gráfelméleti ismeretek.

* 1. ***Követelmények***
     1. **Funkcionális követelmények**

| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Use-case** | **Komment** |
| --- | --- | --- | --- | --- | --- | --- |
| KOCKA1 | A dobókocka dobásával a játékos akciókhoz jut. | Dobunk dobókockával és megszámoljuk, hogy hány helyre léphetünk. | Alapvető | Első csapat megbeszélés | Mozgás, körök kezelése |  |
| LEC1 | Ha felvesszük a logarlécet vége a játéknek és nyertünk. | Felvesszük a logarlécet | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Tárgy felvétele, körök kezelése |  |
| AJTO1 | Az ajtókon egy másik szobába jutunk | Átmegyünk egy ajtón | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Mozgatás |  |
| AJTO2 | Vannak olyan ajtók, amik nem nyílnak mind a két irányba | Megpróbálunk átmenni egy ilyen ajtón a rossz irányba és nem enged | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Mozgatás |  |
| AJTO3 | Vannak olyan ajtók, amik kis időre eltűnnek | Találunk egy ilyen ajtót és figyeljük a viselkedését | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Mozgatás | Nehéz ellenőrizni a randomitási faktor miatt |
| INVI1 | Az inventoryba maximum 5 tárgy fér el | Több tárgyat nem tudunk belerakni | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Tárgy felvétele |  |
| TANAR1 | Ha a tanár felvesz egy tárgyat, az eltűnik | A tanár előtt lévő tárgyat megfigyeljük | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Tárgyak felvétele |  |
| TVSZ1 | A TVSZ-t felvéve 3 alkalommal védve vagyunk a tanárok ellen | TVSZ-szel a kézben a tanárok előtt védve leszünk | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Tárgy használata |  |
| SOR1 | Ha felvesszük, akkor egyszer védelmet biztosít a tanárok lennen. | A tárggyal a kézben védve vagyunk a tanárok előtt | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Tárgy használata |  |
| RONGY1 | Ha felvesszük, akkor 3 körig védelmet biztosít a tanárok ellen | A tárggyal a kézben védve vagyunk a tanárok előtt | Alapvető | A tárggyal a kézben védve vagyunk a tanárok előtt | Tárgy használata |  |
| KOCKA2 | Ha használjuk, újra dobhatunk. | A tárgat használva a körünk újraindul | Opcionális | Első csapat megbeszélés | Körök kezelése |  |
| SAJT1 | Ha használjuk a Camambert, gázos lesz a szoba a körünk végén | A tárgyat használva a gáz hatása érvéynesül | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Tárgyak használata |  |
| FFP2 | Ha használjuk, védve vagyunk két gázos szobától | Gázos szobába belépve nem érvényesül a gáz hatása rajtunk | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Tárgyak használata |  |
| PNP1 | A letett, aktivált tranzisztorpár segítségével képesek vagyunk teleportálni | A helyes teleportálás ellenőrzése | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Mozgatás |  |
| TANAR2 | Ha a tanár elkap, kibukik a hallgató | A tanár a hallgatót elkapva, vége a hallgató játékának | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Játék szabályainak betartása |  |
| SZOBA1 | Minden szobában van egy létszámlimitje és item limitje | A szobában nem lehet kidobni több itemet | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Szobák kezelése |  |
| SZOBA2 | Bizonyos szobák tudnak osztódni | A szobák ketté válnak | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Szobák osztódása |  |
| SZOBA3 | Bizonyos szobák tudnak egyesülni | A szobák egyesülnek | Alapvető | IIT: Projlab weboldala: Feladatkiírás | Szobák egyesülése |  |

* + 1. **Erőforrásokkal kapcsolatos követelmények**

| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| --- | --- | --- | --- | --- | --- |
| OS1 | A programnak több JVM-et tartalmazó, grafikus megjelenítésre képes operációs rendszeren is futnia kell | A program fut Linuxon is | Alapvető | IIT: Projlab. weboldala: Feladatkiírás |  |
| CLOUD1 | A programnak egy felhőben lévő gépen is működnie kell | A BME virtuális gépén működik a program | Alapvető | IIT: Projlab. weboldala: Feladatkiírás |  |
| ELEKTRO1 | A programhoz szükséges informatikai eszközök működtetéséhez elengedhetetlen az áram ellátás | Az eszközök megfelelően működnek | Alapvető | Csapat értekezlet |  |
| DEVICE1 | A szoftver használatához szükséges egy megfelelő számítógép, egér, illetve a program grafikus interfészét megjelenítő monitor | A felhasználó zökkenőmentesen irányítja a programot, illetve kapcsolatba lép vele az imént megnevezett eszközök segítségével | Alapvető | Csapat értekezlet |  |

* + 1. **Átadással kapcsolatos követelmények**

| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| --- | --- | --- | --- | --- | --- |
| JARRUN1 | A program futtatható JAR fájlja elengedhetetlen | A szoftver elindul a JAR fájlból | Alapvető | Csapat értekezlet |  |
| MAPFILES1 | A pályák külön fájlban tárolt adatok, tehát ezek átadása elengedhetetlen | A program hibátlanul működik | Alapvető | Csapat értekezlet |  |
| FILES1 | A program által megjelenített képek átadása elengedhetetlen | A program nem szenved képek hiányában a futása során | Alapvető | Csapat értekezlet |  |
| CMD1 | A program elindítható command line-ból | A szoftver sikeresen elindul parancssorból | Alapvető | IIT: Projlab. követelmények |  |

* + 1. **Egyéb nem funkcionális követelmények**

| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| --- | --- | --- | --- | --- | --- |
| JAR1 | A hordozhatóság eléréséhez a program JAR formátumban csomagolva is elérhető | A program fut a JAR fájl futtatásával | Opcionális | Első csapat megbeszélés |  |
| SEC1 | A program nem tárolja a felhasználók adatait | Nem került mentésre egyetlen felhasználói adat sem | Alapvető | Csapat értekezlet |  |
| TRUST1 | A felhasználók bízhatnak a program hibamentes működésében | A szoftver hibamentesen működik, illetve minden teszten átmegy | Alapvető | IIT: Projlab. követelmények |  |
| TEST1 | A szoftver tesztelése egyszerűen megvalósítható | A tesztelési folyamat nagyobb problémáktól mentes és gördülékeny | Alapvető | Csapat értekezlet |  |
| EXP1 | A szoftver használata nem nyújt rossz élményt a felhasználóknak, könnyen átlátható és kezelhető | A teszt felhasználók nem tapasztalnak rossz élményeket | Alapvető | Csapat értekezlet |  |

* 1. ***Lényeges use-case-ek***
     1. **Use-case leírások**

| **Use-case neve** | **Pálya megtekintése** |
| --- | --- |
| **Rövid leírás** | A játékosok megtekintik a pályát. |
| **Aktorok** | Játékos |
| **Forgatókönyv** | **1.** A program megjeleníti a pályát a játék állapotától függően. |

| **Use-case neve** | **Hallgató irányítása** |
| --- | --- |
| **Rövid leírás** | A játékos végrehajt egy akciót az alábbiak közül: (külön use casek) Mozgatás, Tárgyak felvétele, Tárgyak letevése, Tétlenül maradás, Tárgyak használata |
| **Aktorok** | Játékos |
| **Forgatókönyv** | **1.** A hallgató kiválaszt egy akciót a lehetőségek közül.  **2.** Az akció a forgatókönyvének megfelelően halad tovább a játék |

| **Use-case neve** | **Mozgatás** |
| --- | --- |
| **Rövid leírás** | A játékosok és a játékvezérlő mozgathatja a hozzá tartozó karakter(eke)t. |
| **Aktorok** | Játékos, játékvezérlő |
| **Forgatókönyv** | **1.** A játékos vagy a játékvezérlő kiválasztja a mozgás akciót választja  **2.** Az irányító fél kiválasztja melyik szobába szeretne lépni.  **3.A.1.** A karakter melletti szoba lett választva és az ajtó iránya is helyes, így a karakter pozíciója az új szoba.  **3.B.1.** Nem egy karakter mellti mező lett választva, vagy az ajtó iránya miatt nem tudott a játékos az adott szobába lépni. |

| **Use-case neve** | **Tárgyak felvétele** |
| --- | --- |
| **Rövid leírás** | A játékosoknak és a játék vezérlőnek lehetősége van a szobák közt elszórt tárgyak felvétele az adott karakterük inventoryjába |
| **Aktorok** | Játékos, játék vezérlő |
| **Forgatókönyv** | **1.** A játékos a tárgy felvétele akciót választja  **2.** A játékos a szobában lévő tárgyak közül választ egyet  **3.A.1.** A játékos inventoryjában van hely, így a tárgy belekerül  **3.A.2.** A tárgy eltűnik a szobából.  **3.B.1.** Nincs hely a játékos inventoryjában, így a tárgy felvétele sikertelen. Ilyenkor nem használt el action-t a játékos. |

| **Use-case neve** | **Tárgyak letevése** |
| --- | --- |
| **Rövid leírás** | A játékosoknak lehetősége van az hallgató tárgyainak letevése a jelenlegi pozícióban. |
| **Aktorok** | Játékos |
| **Forgatókönyv** | **1.** A játékos a tárgy letevése akciót választja  **2.A.1.** A játékos szobájában van elég hely, így leteszi a tárgyat a szobába  **2.A.2.** A játékos inventoryjából eltűnik az adott tárgy  **2.B.1.** Nincs hely a játékos szobájában, így a tárgy letevése sikertelen. Ilyenkor nem használt el action-t a játékos. |

| **Use-case neve** | **Tárgyak használata** |
| --- | --- |
| **Rövid leírás** | A játékosoknak lehetőségük van az invntoryjukban lévő tárgyak használatára. |
| **Aktorok** | Játékos |
| **Forgatókönyv** | **1.** A játékos kiválasztja a tárgyak használata akciót.  **2.** A játékos kiválasztja a használni kívánt tárgyat az inventoríjából.  **3.** A játékos, a tárgy vagy a pálya állapota az adott tárgy hatása alapján változik. |

| **Use-case neve** | **Tétlenül maradás** |
| --- | --- |
| **Rövid leírás** | A hallgatóknak és a tanároknak lehetőségük van a körükben tétlenül maradni. |
| **Aktorok** | Játékos, játék vezérlő |
| **Forgatókönyv** | **1.** A játékos kiválasztja a tétlenül maradás akciót.  **2.** A játékos az adott akcióban nem tesz semmilyen aktív cselekvést |

| **Use-case neve** | **Tanárok irányítása** |
| --- | --- |
| **Rövid leírás** | A játék vezérlő irányítja a pályán lévő tanárokat |
| **Aktorok** | Játék vezérlő |
| **Forgatókönyv** | **1.** A játékos kiválaszt egy akciót a lehetőségek közül.  **2.** Az akció forgatókönyvének megfelelően halad tovább a játék. |

| **Use-case neve** | **Körök kezelése** |
| --- | --- |
| **Rövid leírás** | A játék vezérlő kezeli, és betartatja a játék köreire vonatkozó szabályokat |
| **Aktorok** | Játék vezérlő |
| **Forgatókönyv** | **1.** A játék vezérlő megkeresi a következő játékost  **2.** A játék vezérlő dob egy dobókockával (kisorsol egy random számot 1 és 6 között)  **3.** A soron következő játékosnak kioszt annyi akciót, ahányat dobott a kockával. |

| **Use-case neve** | **Hallgató lelkének elvétele** |
| --- | --- |
| **Rövid leírás** | A tanár elveszi az egyik hallgatónak a lelkét, amitől a hallgató kibukik az egyetemről (azaz kiesik a játékból). |
| **Aktorok** | Játék vezérlő |
| **Forgatókönyv** | **1.** A tanár kiválasztja a hallgatót.  **2.A.1.** Egy szobában vannak, így el tudja venni a lelkét.  **2.B.1.** A tanár és a hallgató nem volt egy szobában így a léleklopás sikertelen. |

| **Use-case neve** | **Szobák kezelése** |
| --- | --- |
| **Rövid leírás** | A játék vezérlő random időközönként alakítja a szobák elrendezését, osztódás és egyesülés révén. |
| **Aktorok** | Játék vezérlő |
| **Forgatókönyv** | **1.** A játék vezérlő kiválasztja, hogy osztódjon 1 szoba, vagy egyesüljön 2 szoba.  **2.A.1.** A játék vezérlő az osztódás forgatókönyve szerint jár el.  **2.B.1.** A játék vezérlő az egyesülés forgatókönyvre szerint jár el. |

| **Use-case neve** | **Szobák osztódása** |
| --- | --- |
| **Rövid leírás** | A játék vezérlő kiválaszt egy szobát, ami ketté osztódik. |
| **Aktorok** | Játék vezérlő |
| **Forgatókönyv** | **1.** A játék vezérlő kiválaszt egy szobát  **2.** Egy új szobát hoz létre a játék vezérlő, ugyanolyan tulajdonsággal, mint a kiválasztott szoba.  **3.** A kiválasztott szoba szomszédainak fele szomszédja lesz az új szobának. Ilyenkor a régi szobából, ezekbe a szobákba már nem lehet eljutni. Ugyanez igaz a kiválasztott szobában lévő tárgyakra is.  **4.A.1** Amennyiben volt játékos a kiválasztott szobában, az random eséllyel átkerül az új szobába |

| **Use-case neve** | **Szobák egyesülése** |
| --- | --- |
| **Rövid leírás** | A játék vezérlő kiválaszt két szobát, melyek egyesülnek |
| **Aktorok** | Játék vezérlő |
| **Forgatókönyv** | **1.** A játék vezérlő kiválaszt két szobát, melyekben nincsen lhelyezve tranzisztor.  **2.** A játék vezérlő létrehoz egy új szobát, mely a két választott szoba szomszédjaikkal szomszédos, és a befogadóképessége a kiválasztott szobák befogadóképességének maximumával egyenlő.  **3.A.1.** Bármelyik kiválasztott szoba gázos volt, az új szoba is gázos lesz.  **3.B.1.** Nincsen gázos szoba a kiválasztottak között, a szoba nem lesz gázos.  **4.** A két szoba tárgyainak összesét átteszi az új szobába.  **5.A.1.** A tárgyak száma meghaladja a limitet, így a logarléc kivételével eldob a játék vezérlő annyi tárgyat, hogy a limitnek megfeleljen.  **5.B.1.** A tárgyak száma nem haladja meg a limitet, így az összes tárgy átkerül.  **6.** A két kiválasztott szobát eldobja a játék, és csak a régi szoba marad meg. |

* + 1. **Use-case diagram**



* 1. ***Szótár***

Action - Egy cselekvés a játékosok által

Ajtó - Átmenet 2 szoba között

Aktív tárgy - El kell használni a tárgyat, ahhoz, hogy hatása legyen

Camambert - A diákokat segítő tárgy

Command line - A Parancssor

Diák - A játékosok által irányított karakter

Egyesülés - 2 szobából egy új szoba keletkezik

Felhasználás/Elhasználás - Egy tárgy képességének aktiválása

FFP2 maszk - Egy diákokat segítő tárgy

Felvétel - Egy tárgy felvétele

Gázos szoba - speciális szoba

GUI - A felhasználói felület

Idle - Egy körben a játékos nem él az action-ei felhasználásával

Inventory - A tárgyakat tároló, úgymond tárhely

Játék - A program

Játékos - A szoftver felhasználói

Kiszárad - A nedves rongy, a megadott körig használható ezután elveszti az erejét

Kör - A játékosok egymás után dobása, addig ameddig vissza nem érünk, ahhoz a játékoshoz aki kezdte a kört

Lefagyás - A lefagyó játékos adott számú körből kimarad

Eldobás/Lerakás - Egy tárgy lerakása használat nélkül

Logarléc - A diákok győzelméhez szükséges tárgy

Mező - A pályát alakító gráf egy csúcsa

Nedves rongy - A diákokat segítő tárgy

OOP - objektum orientált programozás

Osztódás - 1 szobából 2 új szoba keletkezik

Passzív tárgy - Felvétel után, automatikusan aktiválódik a hatása

Pálya - Egy irányított gráf, élei az ajtók, csúcsai a szobák

Rongy - A nedves rongy kiszáradva

Sörös pohár - Egy diákokat segítő tárgy

Szoba - A játékban található mezők

Tanár/Oktató - Egy egyszerű mesterséges intelligencia által vezérelt karakter

Tárgy - A játék menete során felvehető és használható, különböző képességekkel bíró eszközök.

Tranzisztor - A diákat segítő speciális tárgy

TVSZ - egy diákokat segítő tárgy a játékban

Use-Case - Használati eset vagy esemény, amely konkrét szerepkörhöz tartozik

* 1. ***Projekt terv***

**2.6.1. Projektütemterv**

| **Azonosító** | **Határidő** | **Feladat** | **Felelős** | **Erőforrás** |
| --- | --- | --- | --- | --- |
| PU0 | feb. 16. 12:00 | Csapatalapítás | A csapat |  |
| PU1 | feb. 26. 14:15 | Követelmény, projekt, funkcionalitás | A csapat | Számítógép, Teams |
| PU2 | márc. 4. 14:15 | Analízis modell kidolgozása (I. változat) - beadás | A csapat | Számítógép, Teams |
| PU3 | márc. 11. 14:15 | Analízis modell kidolgozása (II. változat) - beadás | A csapat | Számítógép, Teams |
| PU4 | márc. 18. 14:15 | Szkeleton tervezése - beadás | A csapat | Számítógép, Teams |
| PU5 | márc. 25. 14:15 | Szkeleton elkészítése | A csapat | Számítógép, Teams |
| PU6 | ápr. 8. 12:00 | Szkeleton bemutatása és herculesre feltöltése | A csapat | Számítógép, Teams |
| PU7 | ápr. 8. 14:15 | Prototípus koncepciója - beadás | A csapat | Számítógép, Teams |
| PU8 | ápr. 15. 14:15 | Prototípus részletes tervezése - beadás | A csapat | Számítógép, Teams |
| PU9 | ápr. 29. 14:15 | Prototípus elkészítése és felöltése herculesre | A csapat | Számítógép, Teams |
| PU10 | máj. 6. 14:15 | Grafikus változat megtervezése - beadás | A csapat | Számítógép, Teams |
| PU11 | máj. 22. 12:00 | Prototípus bemutatása | A csapat | Számítógép, Teams |
| PU12 | máj. 22. 14:15 | Grafikus változat elkészítése | A csapat | Számítógép, Teams |
| PU13 | máj. 24. 12:00 | Egyesített dokumentáció, és grafikus verzió, azaz a végső program beadása | A csapat | Számítógép, Teams |

**2.6.2. Erőforrások, eszközök**

A fejlesztés során felhasznált segédeszközök:

* Dokumentáció: Microsoft Word, Google Docs
* Kommunikáció: Microsoft Teams, Meta messenger
* Modellező eszközök: Lucidchart, Microsoft Visio
* Programozási nyelv: Java
* Főbb programozási elv: OOP
* Fejlesztő környezet: IntelliJ
* Programozási segédeszköz: Sonarlint
* Forráskód megosztás, verziókezelés: GitHub
* Feladatok kezelése, felosztása: Trello
* Egyéb fájlok megosztása: Google Drive
  1. ***Napló***

| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| --- | --- | --- | --- |
| 2024.02.21. 19:00 | 3,5 óra | Bogár  Jánovszki  Koleszár  Varjú | Értekezlet.  Döntés: Bogár a csapatkapitány; platformok mellett döntés (Github, Trello, Teams, Drive)  Tevékenység: Feladatkiírás pontosítása |
| 2023.02.23. 15:00 | 1 óra | Bogár | Tevékenység: Bogár létrehozza, és személyre szabja a használt felületeket |
| 2024.02.24. 13:00 | 1,5 óra | Bogár  Koleszár  Varjú  Zombori | Értekezket.  Döntés: Feladatkiírás pontosítása, feladatok kiosztása |
| 2024.02.24. 20:00 | 4 óra | Koleszár | Tevékenység: A “Követelmény, projekt, funkcionalitás” című dokumentum fedlapjának és öt db. pontjának megírása. |
| 2024.02.25 12:00 | 4 óra | Varjú | Tevékenység: A “Követelmény, projekt, funkcionalitás” című dokumentumban a “Funkciók” című rész elkészítése. |
| 2024.02.25. 14:00 | 3 óra | Zombori | Tevékenység: A “2.3 Követelmények” rész kidolgozása |
| 2024.02.25. 14:00 | 4 óra | Bogár | Tevékenység: A “2.4 Lényeges use-case-ek” és “2.6 Projekt terv” kidolgozása, és napló vezetése |
| 2024.02.25. 20:30 | 1,5 óra | Varjú | Tevékenység: A “Követelmény, projekt, funkcionalitás” című dokumentumban a “Szótár” című rész elkészítése. |
| 2024.02.25. 20:30 | 1,5 óra | Koleszár | Tevékenység: A “Követelmény, projekt, funkcionalitás” című dokumentumban a 2.3-as rész szerkesztése és kiegészítése |
| 2024.02.25. 21:00 | 1 óra | Bogár  Koleszár  Varjú | Értekezlet:  Döntés: A kimaradt feladatrészek megoldásának megbeszélése.  Tevékenység: Ugyanezen feladatok kidolgozása |