# Követelmény, projekt, funkcionalitás

18 – don

Konzulens:

Ferencz Endre

Csapattagok

|  |  |  |
| --- | --- | --- |
| Borbély László Benedek | V19RPS | [laci.dj@gmail.com](mailto:laci.dj@gmail.com) |
| Gáspár Vilmos | GF4UZL | [gaspar.vilmos24@gmail.com](mailto:gaspar.vilmos24@gmail.com) |
| Makki Viktor | I0UA10 | [deldemonho@gmail.com](mailto:deldemonho@gmail.com) |
| Ritter Alex | GS9ERJ | [ritt.alex@gmail.com](mailto:ritt.alex@gmail.com) |
| Papp Albert | F9YTAY | [albertxpapp@gmail.com](mailto:albertxpapp@gmail.com) |
| Zöld Máté | GQH7FW | [mate951006@gmail.com](mailto:mate951006@gmail.com) |
|  |  |  |

2016. február 28.

# Követelmény, projekt, funkcionalitás

## Bevezetés

### Cél

A dokumentum célja a feladatkiírás szerinti „Csillagkapu” logikai játék specifikálása, munkafolyamatainak leírása. Későbbiekben a program tervének, megvalósításának és tesztelésének dokumentálása.

### Szakterület

A szoftver tulajdonképpen egy játék, melynek célja a szórakoztatás. Alkalmas logikai készségek fejlesztésére is. Célközönsége az egyszerű logikai játékkedvelők, minden korosztályban.

### Definíciók, rövidítések

|  |  |
| --- | --- |
| **kifejezés** | **magyarázat** |
| bemenet | A fájlból beolvasott adatok, vagy a felhasználó által nyújtott valós idejű információ. |
| bemutatás | Magyarázathoz kötött szemléltetés, mely bizonyos működések létezését és lépéseit demonstrálja. |
| billentyűzet | Egy bemeneti eszköz, mely a felhasználó által nyújtott információ valós idejű érzékelésére alkalmas, gombok segítségével. |
| háttértár | Hosszú távú adattárolásra alkalmas eszköz. |
| HSZK | Hallgatói Számítógép Központ |
| jar fájl | Olyan fájl, ami java programozási nyelvű elemeket tartalmaz. |
| játékmotor | Olyan programrészlet, mely a játékprogram bemeneteit kiértékeli. |
| Java Runtime Environment 1.8 | Programozási eszközök, környezet alkalmazások fejlesztéséhez Java programnyelven. |
| kiértékelés | Szélsőséges értékekkel tesztelés, működés számszerű értékelése, mérése. |
| kimenet | A program által előállított, a felhasználó számára közölt adatok. |
| monitor | Vizuális kimeneti eszköz. |
| parancssori fordítás | A program futtathatóvá alakításának módja. |
| standard I/O elemek | Java Runtime Environment 1.8-ban ki- és bemenetek kezelésére használt eszközkészlet. |

### Hivatkozások

Feladatkiírás:

<https://www.iit.bme.hu/~projlab/>

Segédlet, hasznos tanácsok:

<https://wiki.sch.bme.hu/Szoftver_projekt_laborat%C3%B3rium>

Java Runtime Environment 1.8:

<http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>

### Összefoglalás

Jelen dokumentum további részeiben lehetőség nyílik a feladatkiírás szerint elkészítendő program alapvető követelményeinek megismerésére. Ezentúl megismerkedhetünk a program leendő funkcióival, use-case-eivel, valamint a megvalósító csapat felépítésével és használt eszközeivel.

## Áttekintés

### Általános áttekintés

A szoftver a legmagasabb architekturális szinten 3 alrendszerből áll: Játékmotor, Pálya és Grafikus felület.

A szoftver legfontosabb alrendszere a Játékmotor, amiben zajlik maga a játék. A felhasználó utasításainak megfelelően mozgatja *O’Neill ezredest*, azaz az egyetlen játékos által irányított karaktert. Figyeli, hogy a két különböző színű csillagkapuból mindig csak egy, a legutoljára kilőtt legyen nyitva, illetve csak a megfelelő speciális falakon jöjjön létre. Valamint hogy két csillagkapu között létrejövő féregjárat mindkét irányba átjárható legyen. Nyomon követi a kilőtt lövedékek helyzetét. Gondoskodik róla, hogy az egyes ajtók csak a hozzájuk tartozó mérlegek lenyomásával nyíljanak ki, számon tartja a dobozokat, figyeli a mozgatásukat és az esetleges megsemmisülésüket. A pályát a Pálya alrendszertől kéri le, ami egy labirintus. A Grafikus felület számára szolgáltatja a szükséges információkat a játék pillanatnyi állásának kirajzolásához.

A Pálya alrendszer a pálya fájlokból olvassa be a pálya felépítésére vonatkozó információkat.

A Grafikus felület a Játékmotor által szolgáltatott információk alapján jeleníti meg a játék pillanatnyi állását. A felhasználó a Grafikus felület segítségével tud tájékozódni a labirintusban és ezáltal tudja *O’Neill ezredest* irányítani.

Hálózati elvárások nincsenek, ugyanis a szoftver nem használ hálózatot. A háttértáron jar fájl(ok)nak. a pálya fájloknak és a grafikus felület által felhasznált képeknek kell helyet biztosítani.

### Funkciók

A feladat egy labirintus játék elkészítése, amelyben a játékos célja a labirintusban található összes ZPM modul megszerzése, amelyek szétszórva helyezkednek el a pályán. A játékos O’Neill ezredes és Jaffa karakterét irányítja, akik a Földet akarják megvédeni Anubistól a gonosz goa’uldtól. Egy ZPM modult úgy tud megszerezni a játékos, ha odaviszi az ezredest vagy Jaffát, aki ha közvetlenül a modul előtt áll, akkor fel tudja venni azt.

A labirintust és a benne haladó utakat falak határolják. Két különböző típusú fal található a pályán. Az egyik közönséges fal, amin a játékos nem tud áthatolni. A másik pedig egy speciális fal, ami rendelkezik a közönséges fal minden tulajdonságával és az teszi speciálissá, hogy létre lehet hozni rajta csillagkaput. A két faltípus szemmel is jól megkülönböztethető.

A játékos nem csak az ezredes és Jaffa mozgatására képes, hanem használhatja azok speciális fegyverét is, amellyel csillagkapukat tudnak létrehozni. A fegyverrel egyenes vonalban tudnak lőni, a lövedék pedig addig halad előre, amíg falba vagy Replikátorba nem ütközik (aki a pályán véletlenszerűen mozog, és ha szakadékba lép, betemeti azt maga után). Ha a játékos a labirintus speciális falát találja el, akkor azon létrejön egy csillagkapu. Ha nem speciális falra lő, akkor a lövedék megsemmisül, és nem hoz létre csillagkaput. A fegyver kétféle lövedéket tud kilőni, amelyek a színükben térnek el, sárga és kék, illetve piros és zöld színűek. A felhasználó dönti el, hogy milyen színű lövedéket lő. A lövedék a színének megfelelő színű csillagkaput hoz létre.

Ha mindkét színű csillagkapu létrejött, akkor a kettő között kialakul egy féregjárat, amely mindkét irányban átjárható. Egyszerre csak egy féregjárat létezhet, így ha a játékos lő, miközben egy féregjárat nyitva van, akkor a lövedék színével megegyező színű korábban létrehozott csillagkapu bezárul. Ha sikeresen megnyílt az új csillagkapu, egy új féregjárat jön létre az új kapu és a megmaradt között.

A féregjáraton nem csak a játékos tud átmenni, hanem tárgyakat is magával tud vinni. Ha a játékos egy tárgyat cipel, akkor nem tud lőni a fegyverével. A labirintusban található tárgyak közül egyedül a dobozokat lehet felvenni és hordozni. A játékos egyszerre csak egy dobozt tud mozgatni. Csak akkor tudja felvenni a dobozt, ha közvetlenül előtte áll, a mozgás során a doboz végig az ezredessel megegyező mezőkön van és le is tuja tenni maga elé.

A pályák teljesítéshez szükség van a féregjáratok használatára, mert a labirintusban különböző akadályok zárjak el a játékost a ZPM moduloktól. Ezek az akadályok többfélék lehetnek: szakadékok, falak, dobozok és ajtók.

A szakadékokon a játékos nem tud közvetlenül átkelni, és ha túlmegy a szélén, akkor beleesik és meghal. Ha egy dobozt ejt a szakadékba, ami úgy lehetséges, hogy a szakadék fölé teszi le, vagy a dobozzal a kezében esik a szakadékba, akkor a doboz megsemmisül. A szakadékok felett át lehet lőni a fegyverrel.

A labirintusban találhatók még ajtók, amelyek a játék indulásakor mindig zárva vannak és falként viselkednek, azaz nem lehet áthatolni rajtuk semmilyen módon. A zárt ajtókon nem lehet elhelyezni csillagkapukat. Minden ajtóhoz tartozik egy mérleg, amivel ki lehet nyitni, még pedig úgy, hogy a mérlegre súlyt helyezünk, és ezzel lenyomjuk. Ez a súly lehet maga a játékos, vagy pedig egy doboz. Amíg súly van a mérlegen, addig a hozzá tartozó ajtó kinyílik, ekkor át lehet menni vagy lőni rajta, viszont amit lekerül róla, az ajtó bezárul. A mérlegek nem félétlenül a hozzájuk tartozó ajtók közvetlen közelében találhatóak.

A játék minden pályája teljesíthető. Az egyes pályák nehézsége változó. A játék a legkönnyebben megoldható pályától halad a legnehezebben megoldhatóig.

A játék akkor ér véget, ha a játékos összeszedte az összes ZPM modult, ami a pályán található vagy a játékos meghal. Illetve a felhasználó bármikor bezárhatja a programot, ezzel leállítva szoftver futását.

### Felhasználók

A felhasználóval szemben nincsenek előzetes követelmények. A szoftver használata gyorsan megtanulható és a fenti szabályok és definíciók megértése után pedig élvezhető minden felhasználó számára.

### Korlátozások

* A játéknak valós idejűnek kell lennie, nem fogadható el másodperces nagyságrendű késleltetés.
* A szoftver színes grafikai felülettel fog rendelkezni, amiről egyértelműen le kell tudni olvasni a játék aktuális állását.
* A játék billentyűzet segítségével irányítható. A billentyűzetkiosztás logikus legyen, pl. nyilakkal lehessen mozgatni a játékost.
* A szoftver fejlesztését dokumentálni kell.
* „*A forrássorok legalább 20 %-ának kommentárnak kell lennie.*”
* A játékban a pályák nehézsége szintenként fog nőni, így folyamatosan egyre nagyobb kihívást állítva a játékos elé.

### Feltételezések, kapcsolatok

Feladatkiírás:

<https://www.iit.bme.hu/~projlab/>

Nem ismert kifejezések:

<https://hu.wikipedia.org/wiki/Goa%E2%80%99uld>

<https://hu.wikipedia.org/wiki/Csillagkapu_%28telev%C3%ADzi%C3%B3s_sorozat%29>

Felhasznált eszközök:

<https://bitbucket.org/>

<https://www.sourcetreeapp.com/>

<http://www.skype.com/hu/>

<https://www.facebook.com/>

## Követelmények

### Funkcionális követelmények

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Use-case** | **Komment** |
| F1. | Játékost lehessen a billentyűzet segítségével irányítani | bemutatás | alapvető | feladatkiírás | Ezredes irányítása, Readkey | A játékosnak (O’Neill ezredesnek) irányíthatónak kell lennie, különben az egész játék értelmét veszti. |
| F2. | A játékos fel tudja szedni a ZPM-eket, ha közvetlenül előttük áll | kiértékelés | alapvető | feladatkiírás | ZPM felvétele, Ezredes irányítása | Mindenképpen alapvető fontosságú, hiszen pontosan ez a játék célja. |
| F3. | A dobozok mozgathatóak, a játékos fel tudja venni őket, majd azokat egy tetszőleges másik helyen le tudja tenni | bemutatás | fontos | feladatkiírás | Doboz mozgatása, Ezredes irányítása | Nem alapvető fontosságú, hiszen elképzelhető olyan pálya, mely ajtók kinyitása nélkül is teljesíthető. |
| F4. | A falakon a játékos ne tudjon keresztülmenni | bemutatás | fontos | feladatkiírás | Ezredes irányítása | Fontos, de nem végzetes. A játékot jelentősen megkönnyítené a hiánya. |
| F5. | A mérlegek lenyomhatóak, melyek a nyomás időtartamáig egy ajtót nyitnak | bemutatás, kiértékelés | fontos | feladatkiírás | Doboz mozgatása, Ezredes irányítása | Fontos, de ugyanúgy, mint a dobozok mozgatásánál, itt is elképzelhető olyan pálya, mely e kritérium nélkül is teljesíthető marad. |
| F6. | Egy ajtót csak és kizárólag egy mérleggel lehet kinyitni | kiértékelés | opcionális | csapat | Doboz mozgatása | A feladatkiírás nem mondta ki explicit, hogy minden ajtóhoz csak egy mérleg tartozhat. |
| F7. | A játékos a fegyverrel két különböző színű csillagkaput tud nyitni | bemutatás | alapvető | feladatkiírás | Lövés fegyverrel | Sárga és kék portál nyitható a fegyverekkel, ha a lövedékek arra alkalmas falat érnek. (zárt ajtó nem minősül annak) |
| F8. | Az ellentétes színű csillagkapuk bármelyikén áthaladva a játékos a másik csillagkapunál jelenik meg | kiértékelés | alapvető | feladatkiírás | Lövés fegyverrel, Doboz mozgatása, Ezredes irányítása | Minden féreglyuk mindkét irányban átjárható, azokon tetszőleges tárgy (doboz) átvihető. |
| F9. | Bármelyik színű csillagkapuból mindig csak és kizárólag a legfrissebben nyitott aktív | kiértékelés | alapvető | feladatkiírás | Lövés fegyverrel | Ha ez nem teljesülne, nem lehetne egyértelműen eldönteni, hogy hova kerül a játékos. |
| F10. | A szakadék nem átjárható, de át lehet lőni rajta | bemutatás | fontos | feladatkiírás | Lövés fegyverrel, Ezredes irányítása | Ha a szakadék átjárható lenne, a játék csak akkor érhetne véget, ha a játékos az összes ZPM-et összeszedte. |
| F11. | A szakadékba esett játékos meghal | kiértékelés | fontos | feladatkiírás | Ezredes irányítása | A játék így is véget érhet. |
| F12. | A szakadékba esett tárgy megsemmisül | kiértékelés | fontos | feladatkiírás, csapat | Doboz mozgatása | A szakadékba esett doboz megsemmisül, nem kerül vissza az eredeti helyére, de ettől függetlenül a pálya továbbra is teljesíthető maradhat. (nem minden esetben) |
| F13. | Új játék kezdése | bemutatás | alapvető | feladatkiírás | Új játék kezdése | Alapvető funkció a menüben. |
| F14. | Kilépési lehetőség a játékból | bemutatás | fontos | csapat | Kilépés | Nem kritikus funkció, de kétségkívül hasznos. |
| F15. | A játéknak akkor van vége, ha a játékos összegyűjti az összes ZPM modult vagy meghal | kiértékelés | alapvető | feladatkiírás | Új játék kezdése | A feladatkiírás szerinti végkifejlet. |
| F16. | Ha a lövedék doboznak ütközik, megsemmisül | kiértékelés | opcionális | csapat | Lövés fegyverrel | De ZPM modul esetén nem semmisül meg. |
| F17. | A kilőtt lövedék színével megegyező színű csillagkapu megsemmisül a fegyver elsütésének pillanatában. | kiértékelés | fontos | csapat | Lövés fegyverrel | Amennyiben létezik. |
| F18. | A játékos dobozzal a kezében nem képes a fegyverével lőni | bemutatás | opcionális | csapat | Doboz mozgatása, Lövés fegyverrel | A nem tisztázott kérdésben a csapat így döntött. |
| F19. | Lövedék korlátlan számban lőhető, de egy időben csak egy lehet mozgásban egy színből | kiértékelés | fontos | feladatkiírás | Lövés fegyverrel | Soha nem fogy ki a lövedék. |
| F20. | Minden második ezredes által felvett ZPM modul után keletkezik egy új | bemutatás | fontos | feladatkiírás | Ezredes irányítása | Jaffa esetén nem |

### Erőforrásokkal kapcsolatos követelmények

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| E1. | Java Runtime Environment 1.8 | bemutatás | alapvető | feladatkiírás | Mivel a szoftvernek a HSZK gépein futtathatónak kell lenniük, ezért az említettnél frissebb és régebbi verzió használata nem ajánlott. |
| E2. | Grafikus megjelenítő (monitor) | bemutatás | fontos | feladatkiírás | A játék alapvetően grafikus felületen működik, az említett eszköz nélkül a játékélmény drasztikusan lecsökken. |
| E3. | Billentyűzet | bemutatás | alapvető | csapat | Bár a feladatkiírás nem mondja ki, hogy mivel kell a játékost irányítani, a csapat a hagyományos billentyűzet mellett döntött. |

### Átadással kapcsolatos követelmények

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| Á1. | A dokumentáció és a programrészletek átadására az előre meghatározott időpontokban kerül sor | bemutatás | alapvető | tárgy honlap | A pontos időpontok a dokumentum végén találhatók |
| Á2. | A program parancssori fordítás után futtatható | bemutatás | alapvető | tárgy honlap | Az előző pontban meghatározott kritériumok teljesítése esetén. |

### Egyéb nem funkcionális követelmények

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| NF1. | Újrafelhasználhatóság, objektumorientált elkészítés a későbbi bővítések/változtatások miatt | bemutatás | fontos | csapat | Törekedjünk a kódduplikáció elkerülésére. |
| NF2. | Az adatok központi tárolása | bemutatás | fontos | csapat | Lehetővé kell tenni a játék változtatását, minimális kódolással. |
| NF3. | A végső program tesztelését a részegységek individuális, majd pedig az összetartozó részek együttes tesztelésével végezzük el | bemutatás | fontos | tárgy honlap | A hibák elkerülése miatt muszáj a lehető legrészletesebben és legalaposabban az összes részegységet letesztelni. |
| NF4. | A játék egyszerűségéből fakadóan, a felhasználónak csak alapvető felhasználói ismeretekre legyen szüksége | bemutatás | opcionális | csapat | A játékhoz nincs szükség semmilyen szakmai előképzettségre. |

## Lényeges use-case-ek

### Use-case leírások

|  |  |
| --- | --- |
| **Use-case neve** | Új játék kezdése |
| **Rövid leírás** | Új játék kezdése |
| **Aktorok** | Játékos |
| **Forgatókönyv** | A játékos kiválasztja az új játék menüpontot.  Betöltődik az 1. pálya, az ezredes kezdő pozícióba áll. |

|  |  |
| --- | --- |
| **Use-case neve** | Kilépés |
| **Rövid leírás** | Kilépés a játékból |
| **Aktorok** | Játékos |
| **Forgatókönyv** | A játékos kiválasztja a kilépés menüpontot, a szoftver bezáródik. |

|  |  |
| --- | --- |
| **Use-case neve** | ZPM felvétele |
| **Rövid leírás** | Az ezredes felvesz egy ZPM modult. |
| **Aktorok** | Játékos |
| **Forgatókönyv** | Az ezredes rálép egy ZPM modul melletti mezőre, ekkor azt automatikusan begyűjti. |

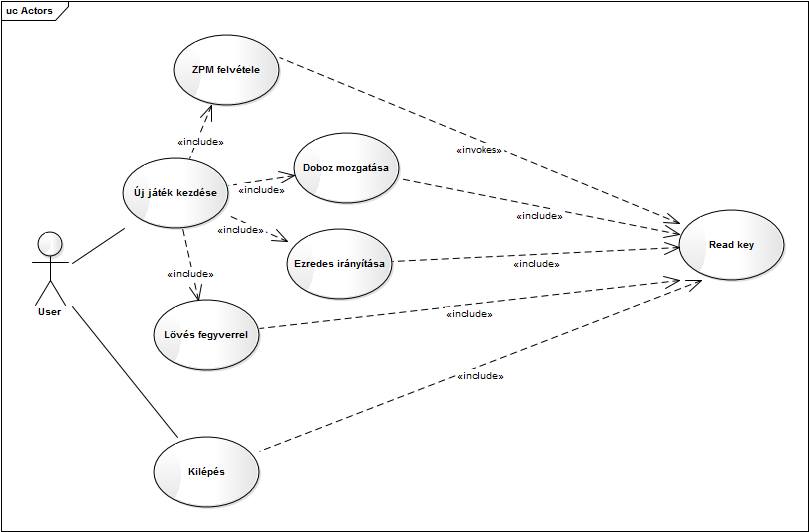
|  |  |
| --- | --- |
| **Use-case neve** | Doboz mozgatása |
| **Rövid leírás** | Az ezredes egy másik helyre helyez egy dobozt. |
| **Aktorok** | Játékos |
| **Forgatókönyv** | Az ezredes egy doboz elé áll, és egy gomb megnyomására felveszi.  Utána a doboz és az ezredes együtt fog mozogni, e gomb újra megnyomásáig.  Ekkor leteszi maga elé az ezredes a dobozt, ami ott marad. |

|  |  |
| --- | --- |
| **Use-case neve** | Ezredes irányítása |
| **Rövid leírás** | Ezredes irányítása a pályán |
| **Aktorok** | Játékos |
| **Forgatókönyv** | Az ezredes billentyűlenyomás hatására elmozdul a pályán a megfelelő irányba. |

|  |  |
| --- | --- |
| **Use-case neve** | Lövés fegyverrel |
| **Rövid leírás** | Az ezredes féregjáratot nyit a falon keresztül a fegyverével |
| **Aktorok** | Játékos |
| **Forgatókönyv** | A játékos megnyom egy gombot, ennek hatására az ezredes lő egy lövedéket a fegyverével.   * Ha olyan falat vagy dobozt talál el, amin nem lehet féregjárat, akkor semmi nem történik. * Ha olyan falat talál el, amin lehet féregjárat   + Ha még nincs csillagkapu a falon, akkor elhelyez rajta egyet.   + Ha már meg volt egy helyen jelölve a fal egy másik színű csillagkapuval, akkor a két megjelölt fal rész között féregjárat fog keletkezni, amelyen keresztül az ezredes szabadon tud közlekedni.   + Ha már létrejött a féreg járat a régebbi csillagkapu felcserélődik azzal a csillagkapuval amelyet legutóbb hozott létre az ezredes. |

|  |  |
| --- | --- |
| **Use-case neve** | Readkey |
| **Rövid leírás** | Olvassuk a billentyűzetet. |
| **Aktorok** | Játékos |
| **Forgatókönyv** | Standard I/O elemeket felhasználva olvassuk a billentyűzetet. |

### Use-case diagram



## Szótár

|  |  |
| --- | --- |
| **kifejezés** | **magyarázat** |
| ajtó | Egy olyan elem a pályán, melynek két állapota létezik: nyitott és zárt. Nyitott állapotban áthaladhat rajta O’Neill ezredes, a fegyvere és lövedékei, valamint dobozok is. Zárt állapotban közönséges falként működik. Az állapotát mindig pontosan egy mérleg segítségével lehet megváltoztatni. |
| alkalmas fal | lásd: speciális fal |
| Anubis | Egy gonosz goa'uld, aki a játékprogramban megvalósított történet alapján el akarja pusztítani a Földet, O’Neill ezredes főellensége. |
| átjárható | Egy pálya eleme átjárható, ha azo(ko)n a mező(kö)n, ami(ke)n az adott elem van, doboz, O’Neill ezredes és fegyvere vagy annak lövedéke is egyidejűleg helyet foglalhat. |
| csillagkapu | Egy olyan különleges elem, melyből minden időpillanatban maximum egy kék és egy sárga színű létezhet. Amennyiben csak egy csillagkapu létezik, az nem átjárható. Ha mindkét csillagkapu létrejött, a kettő között féregjárat alakul ki. |
| doboz | Egy olyan elem a pályán, melyet O’Neill ezredes, ha szomszédos mezőn áll, képes felvenni, és a pálya másik részeibe elmozdítani. |
| fal | Gyűjtőneve a közönséges és speciális falaknak. Nem átjárható elemek. |
| fegyver | O’Neill ezredes különleges eszköze, mellyel két különböző (sárga és kék) lövedékek lőhetőek. |
| féregjárat | Két különböző színű csillagkapu között kialakuló kapcsolat, mely biztosítja, hogy az egyik csillagkapuba beérkező mozgatható elemek (O’Neill ezredes, doboz) a másik csillagkapuból fognak kijönni. Ha az egyik csillagkapu megsemmisül, a féregjárat is. |
| goa'uld | Egy idegen bolygóról érkezett parazita faj, melyek legtöbbször emberi gazdatestet választanak. |
| Jaffa | A játékos által bemenetekkel irányított karakter, melynek megfelelő kezelésével teljesíthetőek a program által nyújtott logikai kihívások. |
| játékos | A játékprogramot futtató, és bemenetekkel irányító felhasználó. A kifejezést, ahol egyértelmű gyakran használjuk O’Neill ezredes helyett. |
| játékos meghal | A kezelt karakter végzetes cselekedetének eredménye, mely a játék végét eredményezi. |
| közönséges fal | Egy olyan elem a pályán, mely nem átjárható, a játékos által el nem mozdítható. |
| különleges fal | lásd: speciális fal |
| labirintus | lásd: pálya |
| lövedék | Egy olyan elem, mely a fegyvertől a választott irányban egyenesen halad az átjárható mezőkön, mindaddig, míg nem átjárhatóba ütközik. Amennyiben a nem átjárható mező speciális fal, a becsapódás helyén a lövedék színével (sárga vagy kék) egyező színű csillagkapu jön létre. Minden más esetben a lövedék megsemmisül. |
| mérleg | Egy olyan elem a pályán, melyre súlyt helyezve kinyílik pontosan egy pályán lévő ajtó, a súly mérlegről eltávolítása után pedig az ajtó visszazárul. |
| mező | A pálya olyan alapegysége, melyen vagy teljesen rajta van egy elem, vagy egyáltalán nem. |
| O’Neill ezredes | A játékos által bemenetekkel irányított karakter, melynek megfelelő kezelésével teljesíthetőek a program által nyújtott logikai kihívások. |
| pálya | A játékprogramban megvalósított logikai fejtörő, melyet a játékos O’Neill ezredes megfelelő irányításával oldhat meg. A pályán több, annak nehézségét és teljesíthetőségét befolyásoló elem található (pl.: falak, mérleg, doboz) |
| Replikátor | A gép által irányított karakter. |
| speciális fal | Egy olyan elem a pályán, mely nem átjárható, a játékos által el nem mozdítható. Ha ilyen típusú falba lövedék csapódik, csillagkapu nyílik rajta. |
| súly | Maga O’Neill ezredes vagy a pályán elhelyezett és elmozgatott doboz(ok) is szolgálhatnak súlyként. |
| szakadék | Egy olyan elem a pályán, melyre ha O’Neill ezredessel rálépünk, meghal, a pálya teljesítése nem sikerült. Amennyiben doboz kerül a szakadékkal azonos mezőre, a doboz megsemmisül. Lövedékek áthaladhatnak azokon a mezőkön, amelyen szakadék van. |
| ZPM modul | Energiaforrás, a játék célja ezek összegyűjtése. A ZPM modulok pályán elhelyezett elemek, melyek mindegyikét össze kell szednie a játékosnak a pálya teljesítéséhez. |

## Projekt terv

## Csapat

## A csapat 6 főből áll. A feladatot úgy próbáljuk felosztani, hogy lehetőleg mindenki azonos nehézségű részt kapjon, miközben az egyéni preferenciáit is betartjuk.

|  |  |
| --- | --- |
| **Név** | **Felelősségek** |
| Borbély László Benedek | kódolás, UML, teszt |
| Gáspár Vilmos | dokumentáció, UML, teszt |
| Makki Viktor | dokumentáció, kódolás, grafika |
| Papp Albert | kódolás, grafika, teszt |
| Ritter Alex | UML, kódolás, grafika |
| Zöld Máté | dokumentáció, UML, kódolás |

## Kommunikáció

## Verziókezelés: Mivel ezen a projekten többen dolgozunk, a dokumentumokat és programkódokat meg kell osztanunk egymással. Nem jó, ha valakinél még régebbi változat vagy hiányos fájlok vannak. Hogy minden egy helyen, a legújabb verziójában legyen.

**Facebook:** A csapatnak nyitottunk egy beszélgetést ahol a sürgősebb dolgokat kommunikálhatjuk le, illetve hogy ki hogy áll az adott feladatával.

**Megbeszélések:** Hetente kétszer tartunk megbeszélést. Az elsőn a feladatokat osztjuk ki, másodikon meg egyeztetjük őket illetve az esetleges problémákat orvosoljuk. Ha valaki nem tud részt venni ezeken a megbeszéléseken Skype segítségével hidaljuk át a problémát

**Levelezőlista:** Mindenkit felvettünk egy levelezőlistára ahol a találkozási időpontokat, helyszíneket és a kulcsfontosságú információkat osszuk meg egymással

## Használt programok

**Verziókezelés:** SourceTree programot használunk, bitbucket központi tárhellyel.

**Dokumentáció** Microsoft Wordöt használjuk, mert mindenki ismeri a csapatból  
Az UML szerkesztésére az Enterprise Architect programot használjuk

**Fejlesztőkörnyezet** Eclipse és IntelliJ IDEA szoftverekkel dolgozunk

## Feladat lépései, határidei

|  |  |
| --- | --- |
| **Dátum** | **Leírás** |
| febr. 29. | Követelmény, projekt, funkcionalitás - beadás |
| márc. 7. | Analízis modell kidolgozása 1. - beadás |
| márc. 14. | Analízis modell kidolgozása 2. - elektronikus feltöltés. A nyomtatott változatot a márc. 16-án kell a laborvezetőnek odaadni. |
| márc. 21. | Szkeleton tervezése - beadás |
| márc. 29. | **Szkeleton** - beadás |
| ápr. 4. | Prototípus koncepciója - beadás |
| ápr. 11. | Részletes tervek - beadás |
| ápr. 25. | **Prototípus** - beadás |
| máj. 2. | Grafikus felület specifikációja - beadás |
| máj. 17. | **Grafikus változat** - beadás |
| máj. 20. | Összefoglalás - beadás |

A feladat főbb lépései (vastagon jelölt) kifejtve (*forrás: tárgyhonlap*):

A **szkeleton** változat célja annak bizonyítása, hogy az objektum és dinamikus modellek a definiált feladat egy modelljét alkotják. A szkeleton egy program, amelyben már valamennyi, a végső rendszerben is szereplő business objektum szerepel. Az objektumoknak csak az interfésze definiált. Valamennyi metódus az indulás pillanatában az ernyőre szöveges változatban kiírja a saját nevét, majd meghívja azon metódusokat, amelyeket a szolgáltatás végrehajtása érdekében meg kell hívnia. Amennyiben a metódusból valamely feltétel fennállása esetén hívunk meg más metódusokat, akkor a feltételre vonatkozó kérdést interaktívan az ernyőn fel kell tenni és a kapott válasz alapján kell a továbbiakban eljárni. A szkeletonnak alkalmasnak kell lenni arra, hogy a különböző forgatókönyvek és szekvencia diagramok ellenőrizhetők legyenek. Csak karakteres ernyőkezelés fogadható el, mert ez biztosítja a rendszer egyszerűségét.

A **prototípu**s program célja annak demonstrálása, hogy a program elkészült, helyesen működik, valamennyi feladatát teljesíti. A prototípus változat egy elkészült program kivéve a kifejlett grafikus interfészt. A változat tervezési szempontból elkészült, az ütemezés, az aktív objektumok kezelése megoldott. A business objektumok - a megjelenítésre vonatkozó részeket kivéve - valamennyi metódusa a végleges algoritmusokat tartalmazza. A megjelenítés és működtetés egy alfanumerikus ernyőn követhető, ugyanakkor a megjelenítés fájlban is logolható, ezzel megteremtve a rendszer tesztelésének lehetőségét. Különös figyelmet kell fordítani az interfész logikájára, felépítésére, valamint arra, hogy az mennyiben tükrözi és teszi láthatóvá a program működését, a beavatkozások hatásait.

A teljes ( **grafikus** ) változat a prototípustól elvileg csak a kezelői felület minőségében különbözhet. Ennek változatnak az értékelésekor a hangsúlyt sokkal inkább a megvalósítás belső szerkezetére, semmint a külalakra kell helyezni.

***Napló***

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2016.02.24. 12:00 | 1 óra | Gáspár, Borbély | Értekezlet.  Dokumentálás alapelvei |
| 2016.02.24. 19:30 | 1,5 óra | Gáspár, Borbély, Papp, Zöld, Ritter | Verziókezelő szoftverek előkészítése,  feladatok szétosztása |
| 2016.02.26. 17:30 | 4 óra | Borbély | Követelmények meghatározása |
| 2016.02.27. 20:00 | 2,5 óra | Ritter | Projekt terv |
| 2016.02.27. 21:00 | 2,5 óra | Zöld | Áttekintés, Funkciók, Felhasználók |
| 2016.02.27. 21:00 | 3 óra | Papp | Use-casek, korlátozások |
| 2016.02.28. 9:00 | 3,5 óra | Gáspár | Cél, Szótár, definíciók kitöltése |
| 2016.02.28. 16:00 | 3 óra | Gáspár, Borbély, Papp, Zöld, Makki, Ritter | Beadás előtti értekezlet, dokumentum egységesítése |

# 3. Analízis modell kidolgozása

18 – don

Konzulens:

Ferencz Endre

Csapattagok

|  |  |  |
| --- | --- | --- |
| Borbély László Benedek | V19RPS | [laci.dj@gmail.com](mailto:laci.dj@gmail.com) |
| Gáspár Vilmos | GF4UZL | [gaspar.vilmos24@gmail.com](mailto:gaspar.vilmos24@gmail.com) |
| Makki Viktor | I0UA10 | [deldemonho@gmail.com](mailto:deldemonho@gmail.com) |
| Ritter Alex | GS9ERJ | [ritt.alex@gmail.com](mailto:ritt.alex@gmail.com) |
| Papp Albert | F9YTAY | [albertxpapp@gmail.com](mailto:albertxpapp@gmail.com) |
| Zöld Máté | GQH7FW | [mate951006@gmail.com](mailto:mate951006@gmail.com) |
|  |  |  |

2016. március 6.

# Analízis modell kidolgozása

## Objektum katalógus

### Abyss

A pályán található szakadék. Azok az objektumok, amelyek beleesnek, mint a játékos vagy a doboz megsemmisülnek. Tetszőlegesen sok lehet belőle a pályán.

### Box

A pályán elhelyezkedő doboz.

### BoxInventory

A pályán elhelyezkedő mozgatható objektumokat tárolja. Ilyen objektum a Box. Ezeket az objektumokat a pályán elfoglalt koordinátájukkal együtt tárolja.

### Button

A mérleget megvalósító objektum, ami egy nyomógomb. Az ajtók számának megfelelő számú gomb található a pályán. Minden Button-hoz tartozik egy Door, a Button tartja nyilván ezt a hozzárendelés. Feladata, hogy az ajtó állapotát megváltoztassa.

### ButtonInventory

A pályán elhelyezkedő Button-okat tárolja a koordinátájukkal együtt.

### CollectableInventory

A pályán elhelyezkedő gyűjthető objektumokat tárolja. Ilyen objektum a ZPM. Ezeket az objektumokat a pályán elfoglalt koordinátájukkal együtt tárolja.

### Colonel

Ezredes objektum

### Coordinate

Tárolja a pozíciót.

### CVector

Tárolja a pozíciót és irányt.

### DataAccessPoint

A StepHandler, a ShotHandler és a GrabHandler számára biztosít hozzáférést a FieldObjectInventory-hoz, BoxInventory-hoz és a CollectableInventory-hoz. Egy ilyen objektum van.

### Door

Ajtó, ami a pályán található. Két állapota van. Létrejöttekor csukott állapotban van és csak a hozzátartozó gomb lenyomásakor kerül nyitott állapotba. Feladata ezen állapotok nyilvántartása, azaz, hogy tudja, hogy átjárható-e vagy nem.

### FieldObjectInventory

Tárolja azokat az objektumokat, amik a pályát alkotják. Ezek közé tartozik a Wall, a SpecialWall, az Abyss, a Way és a Door. Az objektumokat a pályán elfoglalt koordinátájukkal együtt tárolja.

### GrabHandler

Kezeli a játékos egy objektum felvétele illetve lerakását.

### Jaffa

Jaffa objektum.

### Player

Nyilvántartja a játékos állapotát (hely, irány, dobozkézben).

### Replicator

Replikátor objektum.

### ShotHandler

Kezeli a játékos lövéseit, csak akkor akar Csillagkaput létrehozni, ha az eltalált mezőn létrehozható csillagkapu.

### SpecialWall

Speciális fal, amelyen létrejöhet csillagkapu. Egy ilyen falon csak egy csillagkapu lehet.

### StarGate

Nyilvántartja a csillagkapu színét illetve pozícióját.

### StarGateInventory

Tárolja a csillagkapukat, kezeli a helyüket és megmondja, hogy rálépve melyik mezőre kerülünk.

### StepHandler

Kezeli a játékos lépéseit, csak akkor lép az egyik irányba, ha ott nincs olyan mező illetve objektum, ami ezt nem engedné.

### Wall

Ez az objektum a pályán elhelyezkedő egységnyi fal. Tetszőleges számú lehet belőle. Feladata, hogy nem lehet áthatolni rajta semmilyen módon, így csillagkapu sem jöhet létre rajta, mert akkor féregjárattal át lehetne menni rajta.

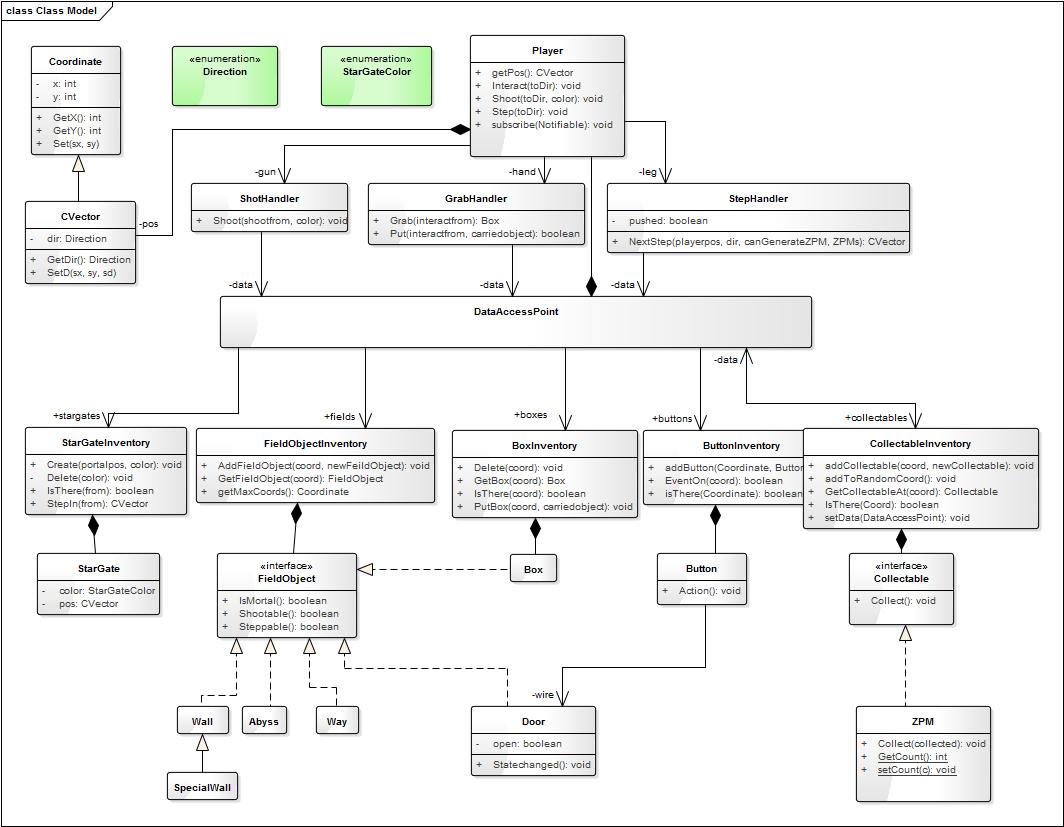
### Way

A pálya járható útja. Egyetlen feladata, hogy biztosítsa, hogy rá lehessen lépni. Szintén tetszőlegesen sok lehet belőle.

### ZPM

Összegyűjtendő objektum, amelyből tetszőleges számú lehet. A játék célja ezen objektumok összegyűjtése. Feladat, hogy biztosítsa a játékosnak, hogy feltudja venni.

## Statikus struktúra diagramok



## Osztályok leírása

### Abyss

#### Felelősség

Szakadékot megvalósító osztály. Átjárható, ezáltal át lehet lőni felett. Nem jöhet létre rajta csillagkapu. Halálos.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok:

#### boolean Steppable(): Rá lehet-e lépni az objektumra.

#### boolean Shootable(): Lőhető-e az objektumra csillagkapu.

#### boolean IsMortal(): Halálos-e az objektum.

#### boolean isDoor(): Ajtó-e az objektum.

### Box

#### Felelősség

A dobozt megvalósító osztály. Mozgatható, nem átjárható, nem lehet átlőni felette, nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek: FieldObject

#### Attribútumok: Nincs

#### Metódusok:

#### boolean Steppable(): Rá lehet-e lépni az objektumra.

#### boolean Shootable(): Lőhető-e az objektumra csillagkapu.

#### boolean IsMortal(): Halálos-e az objektum.

#### boolean isDoor(): Ajtó-e az objektum.

### BoxInventory

#### Felelősség

Tárolja a doboz objektumokat.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **boxes: HashMap<Coordinate, Box>**

#### Metódusok

* **void Delete(Coordinate coord)**: Törli a paraméterként átvett koordinátán található Box interface-t megvalósító objektumot.
* **Box GetBox(Coordinate coord)**: Visszaadja a paraméterként átvett koordinátán található objektumot a HashMap-ből, ha nincs azon a koordinátán objektum, akkor null-al tér vissza.
* **bool IsThere(Coordinate coord)**: Visszaadja, hogy van-e a keresett koordinátán objektum.
* **void addBox(Coordinate coord, Box newBox)**: A paraméterként átvett koordinátával elhelyezi a HashMap-ban a newBox-ban található objektumot, ha az nem null.

### Button

#### Felelősség

Megvalósítja a mérleget, ami egy nyomógomb. Feladata, hogy tudja, melyik ajtó tartozik hozzá, hogy annak meg tudja változtatni az állapotát.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **wire: Door**

#### Metódusok

* **Door getDoor():** Visszadja a nyomólaphoz tartozó ajtót.
* **void lockDoor():** Ajtó zárása.
* **void unlockDoor():** Ajtó nyitása.
* **bool isLocked():** Zárva van-e az ajtó.
* **int getRequired Weight():** Visszaadja az ajtó nyitásához szükséges súlyt.

### ButtonInventory

#### Felelősség

Tárolja a pályán elhelyezkedő Button objektumokat.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **buttons: HashMap<Coordinate, Button>**

#### Metódusok

* **void addButton(Coordinate coordinate, Button button):** Nyomólap hozzáadása listához.
* **void EventOn(Coordinate coordinate, int weightOnCoordinate):** Nyomólapra történő dobozlerakás vagy rálépés esetén lefutó függvény.
* **Button getButtonAt(Coordinate coordinate):** Visszaadja a megadott koordinátán levő nyomólapot.
* **boolean isOpenAt(Coordinate coordinate):** Megadja, hogy van-e nyitott állapotban lévő nyomólap az adott koordinátán.
* **boolean isThere(Coordinate coordinate):** Megadja, hogy van-e nyomólap az adott koordinátán.

### CollectableInventory

#### Felelősség

Tárolja a gyűjthető objektumokat, azaz azokat, amelyek megvalósítják a Collectable interface-t.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **collectables: HashMap<Coordinate, Collectable>**

#### Metódusok

* **void addCollectable(Coordina coordinate Collectable newCollenctable):** Gyűjthető elem hozzáadása.
* **void addToRandomCoordinate():** Gyűjthető elem hozzáadása a pálya egy random pontjára.
* **void GetCollectableAt(Coordinate coordinate):** Gyűjthető elem felvétele.
* **boolean isThere(Coordinate coordinate):** Megadja, hogy van-e az adott koordinátán gyűjthető elem.
* **void setData(DataAccessPoint dataAccessPoint):** A GameObjects.ZPM generáláshoz ismernie kell a fieldObjectek és box-ok helyét.
* **int ZPMCount():** Visszaadja a pályán lévő ZPM-ek számát.

### Collectable

#### Felelősség

Interface a gyűjthető osztályok számára.

#### Ősosztályok: Nincs

#### Metódusok

* **void Collect()**: Az interface-t megvalósító objektum begyűjtésekor hívódik meg.

### Coordinate

#### Felelősség

*Tárolja x, y koordinátákat.*

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **x**: int: x koordinátát tárolja.
* **y**: int: y koordinátát tárolja.

#### Metódusok

* **void Set(int sx, int sy)**: Beállítja erre a pozícióra koordinátákat.
* **int GetX()**: Visszaadja x értékét.
* **void GetY()**: Visszaadja y értékét.

### CVector

#### Felelősség

*Tárolja x, y koordinátákat, és az irányt ahova néz az adott objektum.*

#### Ősosztályok: Coordinate

#### Interfészek: Nincs

#### Attribútumok

* **dir**: Direction: Az irányt tárolja.

#### Metódusok

* **void SetDir(int sx, int sy, Direction sd)**: Beállítja erre a pozícióra koordinátákat.
* **Direction GetDir()**: Visszaadja az irányt.

### DataAccessPoint

#### Felelősség

Hozzáférés biztosítása a StepHandler, a ShotHandler és a GrabHandler számára a FieldObjectInventory-hoz, BoxInventory-hoz és a CollectableInventory-hoz.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **field**: FieldObjectInventory: Referencia a FieldObject-ek tárolójára
* **Box**: BoxInventory: Referencia a mozgatható objektumok tárolójára.
* **collectable**: CollectableInventory: Referencia gyűjthető objektumok tárolójára

#### Metódusok: Nincs

### Door

#### Felelősség

Az ajtót megvalósító osztály. Felelőssége, hogy tudja, hogy milyen állapotban van. Zárt állapotban nem lehet átmenni rajta, így átlőni sem. Nyílt állapotban pedig át lehet menni rajta és így át is lehet lőni a nyitott ajtón. Semelyik állapotában nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok

* **open: boolean**

#### Metódusok

* **void** **Statechanged()**: Megváltoztatja az ajtó állapotát. Ha nyitva volt, akkor becsukja, ha zárva volt, akkor kinyitja.

#### boolean Steppable(): Rá lehet-e lépni az objektumra.

#### boolean Shootable(): Lőhető-e az objektumra csillagkapu.

#### boolean IsMortal(): Halálos-e az objektum.

#### boolean isDoor(): Ajtó-e az objektum.

### FieldObject

#### Felelősség

Interface, amely gondoskodik arról, hogy azokról az osztályokról, amik megvalósítják, el lehessen dönteni, hogy átjárhatók-e, át lehet-e lőni felettük, létre lehet-e hozni felette csillagkaput, illetve halálosak-e.

#### Ősosztályok: Nincs

#### Metódusok

#### boolean Steppable(): Rá lehet-e lépni az objektumra.

#### boolean Shootable(): Lőhető-e az objektumra csillagkapu.

#### boolean IsMortal(): Halálos-e az objektum.

#### boolean isDoor(): Ajtó-e az objektum.

### FieldObjectInventory

#### Felelősség

FieldObject interface-et megvalósító objektumok tárolása.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **fieldobj: HashMap<Coordinate, FieldObject>**

#### Metódusok

* **FieldObject GetFieldObject(Coordinate coord)**: A paraméter listán átvett koordinátán található FieldObject-et visszaadja a HashMap-ből, ha nincs a megodott koordinátán FieldObject, akkor null-t ad vissza.

### GrabHandler

#### Felelősség

A mozgatható objektumok mozgatásáról gondoskodik.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **sgates**: StarGateInventory: Nyilvántartja a csillagkapukat és pozíciójukat.
* **data**: DataAcessPoint: Adat-összeköttetést biztosít.
* **weight:** ButtonHandler: Kezeli a gombok (mérlegek) lenyomását.

#### Metódusok

* **boolean Grab(CVector interactfrom)**: CVectorral adjuk meg, hogy honnan milyen irányból akarunk felvenni, true-t ad vissza, ha a felvétel sikerült.
* **boolean Put(CVector interactfrom)**: CVectorral adjuk meg, hogy honnan milyen irányból akarunk letenni, true-t ad vissza, ha a letételsikerült.

### Player

#### Felelősség

Tárolja a Player (O’Neill ezredes) állapotát: pozícióját, irányát és hogy van-e valami a kezében. Ezeket változtatja a megfelelő függvényekkel.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **handfull**: boolean: Van-e valami mozgatható a Player kezében.
* **pos:** CVector: Megadja a Player pozícióját és irányát.

#### Metódusok

* **CVector GetPos()**: Visszaadja a Player pozícióját, irányát.
* **void SetPos(CVector ppos)**: Megváltoztatja a Player pozícióját, irányát.
* **boolean GetInHand()**: Visszaadja a handfull attribútum értékét.
* **void SetInHand(boolean b)**: Megváltoztatja a handfull attribútum értékét.

### PlayerActionHandler

#### Felelősség

Bemenet szerint továbbítja az események vezérlését, melyek a játékoshoz köthetőek.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **leg**: StepHandler: Kezeli a Player lépéseit.
* **gun**: ShootHandler: Kezeli a Player lövését.
* **hand**: GrabHandler: Kezeli a mozgatható objektumok felszedését.
* **pushing**: ButtonHandler: Kezeli a gombok (mérlegek) lenyomását.
* **data:** Player: Tárolja a Player állapotát.

#### Metódusok

* **void Interact(Direction toDir)**: Megpróbál felvenni vagy lerakni abba az irányba.
* **void shoot(Direction toDir, StarGatColor color)**:Lő abba az irányba ilyen színű Csillagkaput.
* **void Step(Direction toDir)**: Megpróbál lépni abba az irányba.

### ShootHandler

#### Felelősség

Lekezeli a játékos lövéseit.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **sgates**: StarGateHandler: Kezeli a csillagkapukat.
* **data**: DataAcessPoint: Adat-összeköttetést biztosít.

#### Metódusok

* **void Shoot(CVector shootfrom, Direction color)**: CVectorral átadjuk a pozíciónkat és hogy onnan milyen irányból akarunk lőni. Kiszámolja a becsapódás helyét, ha shootable, vezérléstovábbadással gondoskodik az esetleges csillagkapukról.

### SpecialWall

#### Felelősség

Speciális falat megvalósító osztály. Biztosítja, hogy nem átjárható, nem lehet átlőni felette. Nem halálos. Abban különbözik a Wall-tól, hogy létrejöhet rajta csillagkapu.

#### Ősosztályok: Wall

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok: Nincs

### StarGate

#### Felelősség

Maga a csillagkapu objektum, nyilvántartja a színét.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **color**: StarGateColor: Tárolja a csillagkapuk színét.
* **pos:** CVector: Tárolja a csillagkapuk pozícióját.

#### Metódusok

### StarGateInventory

#### Felelősség

Nyilvántartja a csillagkapukat.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **sgates**: StarGate: Tárolja a csillagkapukat.
* **weight:** ButtonHandler: Kezeli a gombok (mérlegek) lenyomását.

#### Metódusok

* **boolean IsThere(CVector from)**: Visszaadja, hogy az adott helyről érkezve van-e csillagkapu.
* **CVector StepIn(CVector from)**: Adott helyről belép a csillagkapuba, visszaadja a következő pozíciót.

### StepHandler

#### Felelősség

Lekezeli a játékos lépéseit, információt szolgáltat a mezők járhatóságáról.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok:

* **sgates**: StarGateInventory: Nyilvántartja a csillagkapukat és pozíciójukat.
* **data**: DataAcessPoint: Adat-összeköttetést biztosít.

#### Metódusok

* **CVector Nextstep(CVector playerpos, Direction dir)**: Visszaadja a Player pozícióját, annak függvényében, hogy tudott-e lépni vagy sem.

### Wall

#### Felelősség

A falat megvalósító osztály. Biztosítja, hogy nem lehet átmenni, átlőni rajta. Nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok: Nincs

### Way

#### Felelősség

Az utat megvalósító osztály. Átjárható, át lehet lőni felette, nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok: Nincs

### ZPM

#### Felelősség

A játék során begyűjtendő ZPM-et megvalósító osztály.

#### Ősosztályok: Nincs

#### Interfészek

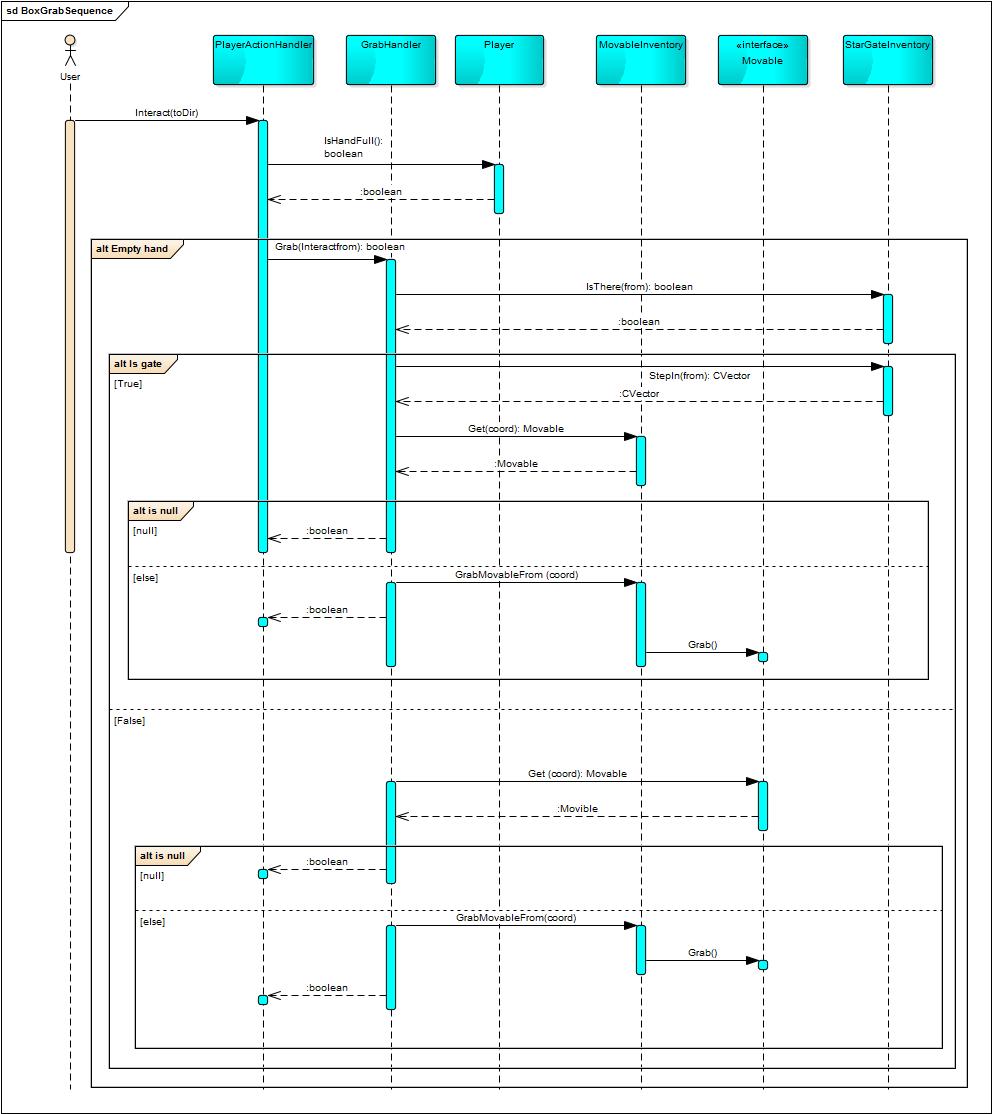
* Collectable

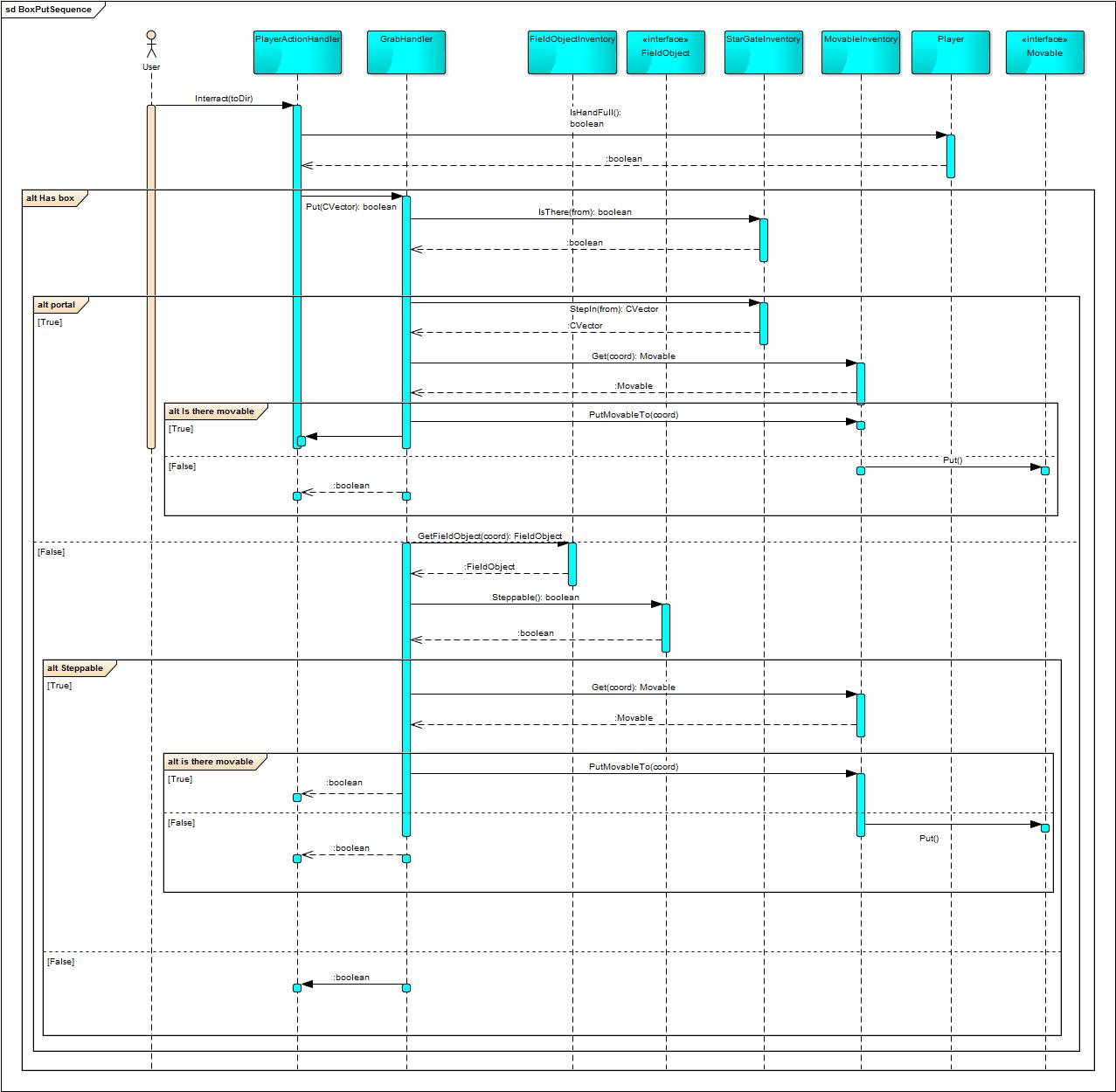
#### Attribútumok: Nincs

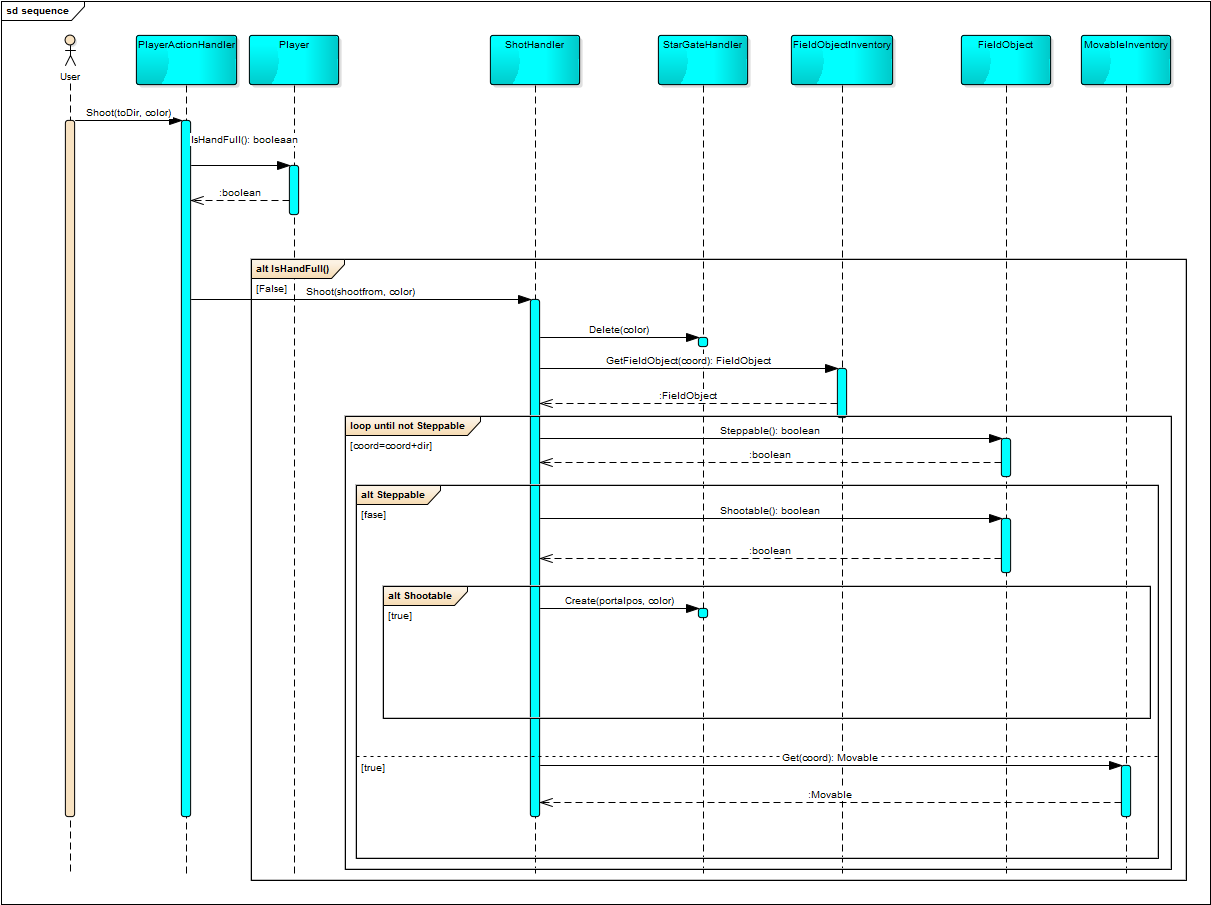
#### Metódusok: Nincs

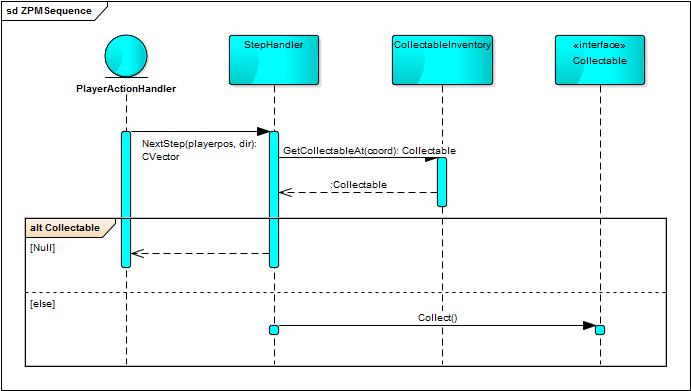
## Szekvencia diagramok

## StepSequence

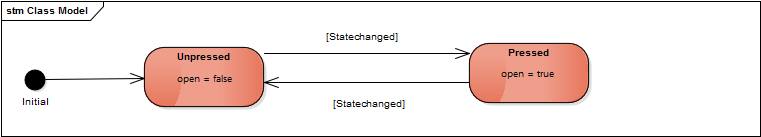








## State-chartok



## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2016.02.29 18:00 | 4 óra | Makki, Gáspár, Ritter, Papp, Borbély | Gondolkodás a felépítésen |
| 2016.03.03 16:30 | 3 óra | Makki, Ritter, Zöld, Gáspár, Papp | Osztálydiagram |
| 2016.03.04 14:30 | 4 óra | Ritter, Zöld, Gáspár, Papp | Osztálydiagram 2.0 |
| 2016.03.05 13:00 | 1 óra | Makki Zöld, Gáspár, Papp, Ritter | Osztálydiagram véglegesítése |
| 2016.03.05 14:00 | 5 óra | Gáspár | Osztálydiagram elkészítése, szekvenciák leírása |
| 2016.03.06 16:00 | 3 óra | Zöld | Objektuok és osztályok leírásai: DataAcsessPoint, FieldObjectInventory, BoxInventory, CollectableInventory, ButtonInventory, Button, FieldObject, Wall, SpecialWall, Abyss, Way, Door, Box, Box, Collectable, ZPM |
| 2016.03.06 16:00 | 6 óra | Makki | Sekvencia diagramok elkészítése |
| 2016.03.07.18:00 | 4 óra | Ritter | A maradék osztályok és objektumok leírásai |
| 2016.03.07 19:00 | 2 óra | Papp | Shoot szekvencia diagram |
| 2016.03.07.19:00 | 15 perc | Borbély | Állapotgép |

# Analízis modell kidolgozása 2

18 – don

Konzulens:

Ferencz Endre

Csapattagok

|  |  |  |
| --- | --- | --- |
| Borbély László Benedek | V19RPS | [laci.dj@gmail.com](mailto:laci.dj@gmail.com) |
| Gáspár Vilmos | GF4UZL | [gaspar.vilmos24@gmail.com](mailto:gaspar.vilmos24@gmail.com) |
| Makki Viktor | I0UA10 | [deldemonho@gmail.com](mailto:deldemonho@gmail.com) |
| Ritter Alex | GS9ERJ | [ritt.alex@gmail.com](mailto:ritt.alex@gmail.com) |
| Papp Albert | F9YTAY | [albertxpapp@gmail.com](mailto:albertxpapp@gmail.com) |
| Zöld Máté | GQH7FW | [mate951006@gmail.com](mailto:mate951006@gmail.com) |
|  |  |  |

2016. március 14.

# Analízis modell kidolgozása 2

## Objektum katalógus

### Abyss

A pályán található szakadék. Azok az objektumok, amelyek beleesnek, mint a játékos vagy a doboz megsemmisülnek. Tetszőlegesen sok lehet belőle a pályán.

### Box

A pályán elhelyezkedő doboz.

### BoxInventory

A pályán elhelyezkedő mozgatható objektumokat tárolja. Ilyen objektum a Box. Ezeket az objektumokat a pályán elfoglalt koordinátájukkal együtt tárolja.

### Button

A mérleget megvalósító objektum, ami egy nyomógomb. Az ajtók számának megfelelő számú gomb található a pályán. Minden Button-hoz tartozik egy Door, a Button tartja nyilván ezt a hozzárendelés. Feladata, hogy az ajtó állapotát megváltoztassa.

### ButtonInventory

A pályán elhelyezkedő Button-okat tárolja a koordinátájukkal együtt.

### CollectableInventory

A pályán elhelyezkedő gyűjthető objektumokat tárolja. Ilyen objektum a ZPM. Ezeket az objektumokat a pályán elfoglalt koordinátájukkal együtt tárolja.

### Colonel

Ezredes objektum

### Coordinate

Tárolja a pozíciót.

### CVector

Tárolja a pozíciót és irányt.

### DataAccessPoint

A StepHandler, a ShotHandler és a GrabHandler számára biztosít hozzáférést a FieldObjectInventory-hoz, BoxInventory-hoz és a CollectableInventory-hoz. Egy ilyen objektum van.

### Door

Ajtó, ami a pályán található. Két állapota van. Létrejöttekor csukott állapotban van és csak a hozzátartozó gomb lenyomásakor kerül nyitott állapotba. Feladata ezen állapotok nyilvántartása, azaz, hogy tudja, hogy átjárható-e vagy nem.

### FieldObjectInventory

Tárolja azokat az objektumokat, amik a pályát alkotják. Ezek közé tartozik a Wall, a SpecialWall, az Abyss, a Way és a Door. Az objektumokat a pályán elfoglalt koordinátájukkal együtt tárolja.

### GrabHandler

Kezeli a játékos egy objektum felvétele illetve lerakását.

### Jaffa

Jaffa objektum.

### Player

Nyilvántartja a játékos állapotát (hely, irány, dobozkézben).

### Replicator

Replikátor objektum.

### ShotHandler

Kezeli a játékos lövéseit, csak akkor akar Csillagkaput létrehozni, ha az eltalált mezőn létrehozható csillagkapu.

### SpecialWall

Speciális fal, amelyen létrejöhet csillagkapu. Egy ilyen falon csak egy csillagkapu lehet.

### StarGate

Nyilvántartja a csillagkapu színét illetve pozícióját.

### StarGateInventory

Tárolja a csillagkapukat, kezeli a helyüket és megmondja, hogy rálépve melyik mezőre kerülünk.

### StepHandler

Kezeli a játékos lépéseit, csak akkor lép az egyik irányba, ha ott nincs olyan mező illetve objektum, ami ezt nem engedné.

### Wall

Ez az objektum a pályán elhelyezkedő egységnyi fal. Tetszőleges számú lehet belőle. Feladata, hogy nem lehet áthatolni rajta semmilyen módon, így csillagkapu sem jöhet létre rajta, mert akkor féregjárattal át lehetne menni rajta.

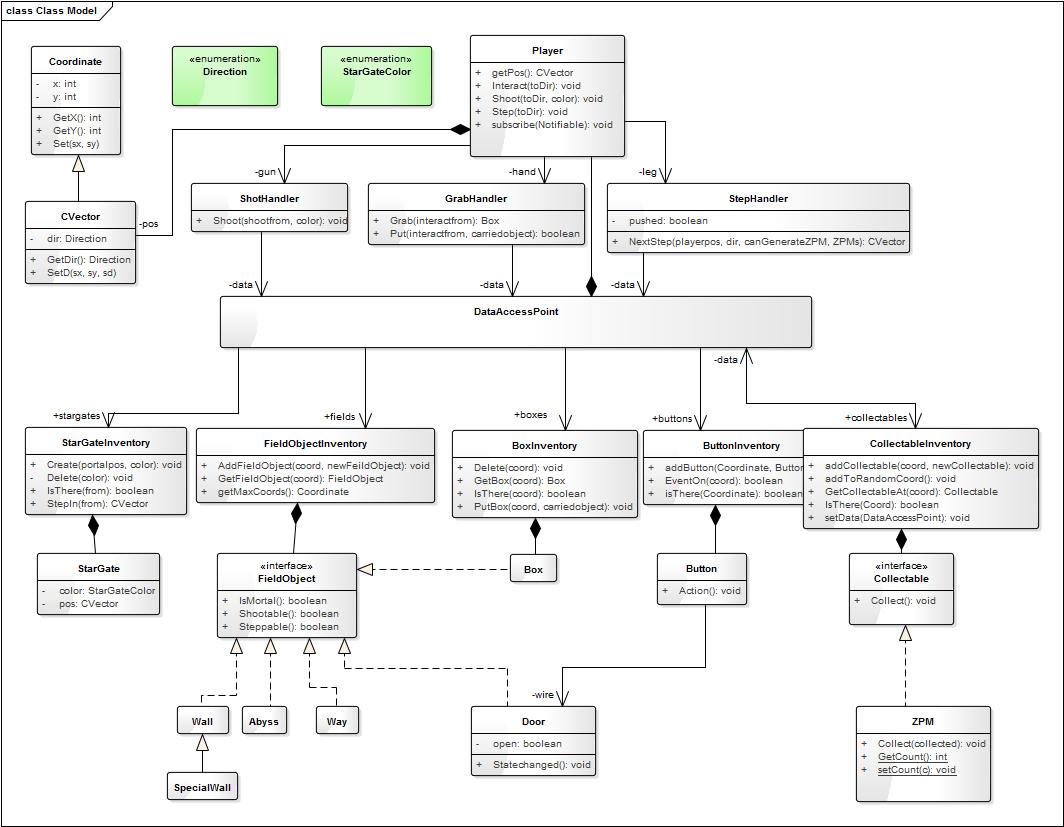
### Way

A pálya járható útja. Egyetlen feladata, hogy biztosítsa, hogy rá lehessen lépni. Szintén tetszőlegesen sok lehet belőle.

### ZPM

Összegyűjtendő objektum, amelyből tetszőleges számú lehet. A játék célja ezen objektumok összegyűjtése. Feladat, hogy biztosítsa a játékosnak, hogy feltudja venni.

## Statikus struktúra diagramok



## Osztályok leírása

### Abyss

#### Felelősség

Szakadékot megvalósító osztály. Átjárható, ezáltal át lehet lőni felett. Nem jöhet létre rajta csillagkapu. Halálos.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok:

#### boolean Steppable(): Rá lehet-e lépni az objektumra.

#### boolean Shootable(): Lőhető-e az objektumra csillagkapu.

#### boolean IsMortal(): Halálos-e az objektum.

#### boolean isDoor(): Ajtó-e az objektum.

### Box

#### Felelősség

A dobozt megvalósító osztály. Mozgatható, nem átjárható, nem lehet átlőni felette, nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek: FieldObject

#### Attribútumok: Nincs

#### Metódusok:

#### boolean Steppable(): Rá lehet-e lépni az objektumra.

#### boolean Shootable(): Lőhető-e az objektumra csillagkapu.

#### boolean IsMortal(): Halálos-e az objektum.

#### boolean isDoor(): Ajtó-e az objektum.

### BoxInventory

#### Felelősség

Tárolja a doboz objektumokat.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **boxes: HashMap<Coordinate, Box>**

#### Metódusok

* **void Delete(Coordinate coord)**: Törli a paraméterként átvett koordinátán található Box interface-t megvalósító objektumot.
* **Box GetBox(Coordinate coord)**: Visszaadja a paraméterként átvett koordinátán található objektumot a HashMap-ből, ha nincs azon a koordinátán objektum, akkor null-al tér vissza.
* **bool IsThere(Coordinate coord)**: Visszaadja, hogy van-e a keresett koordinátán objektum.
* **void addBox(Coordinate coord, Box newBox)**: A paraméterként átvett koordinátával elhelyezi a HashMap-ban a newBox-ban található objektumot, ha az nem null.

### Button

#### Felelősség

Megvalósítja a mérleget, ami egy nyomógomb. Feladata, hogy tudja, melyik ajtó tartozik hozzá, hogy annak meg tudja változtatni az állapotát.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **wire: Door**

#### Metódusok

* **Door getDoor():** Visszadja a nyomólaphoz tartozó ajtót.
* **void lockDoor():** Ajtó zárása.
* **void unlockDoor():** Ajtó nyitása.
* **bool isLocked():** Zárva van-e az ajtó.
* **int getRequired Weight():** Visszaadja az ajtó nyitásához szükséges súlyt.

### ButtonInventory

#### Felelősség

Tárolja a pályán elhelyezkedő Button objektumokat.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **buttons: HashMap<Coordinate, Button>**

#### Metódusok

* **void addButton(Coordinate coordinate, Button button):** Nyomólap hozzáadása listához.
* **void EventOn(Coordinate coordinate, int weightOnCoordinate):** Nyomólapra történő dobozlerakás vagy rálépés esetén lefutó függvény.
* **Button getButtonAt(Coordinate coordinate):** Visszaadja a megadott koordinátán levő nyomólapot.
* **boolean isOpenAt(Coordinate coordinate):** Megadja, hogy van-e nyitott állapotban lévő nyomólap az adott koordinátán.
* **boolean isThere(Coordinate coordinate):** Megadja, hogy van-e nyomólap az adott koordinátán.

### CollectableInventory

#### Felelősség

Tárolja a gyűjthető objektumokat, azaz azokat, amelyek megvalósítják a Collectable interface-t.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **collectables: HashMap<Coordinate, Collectable>**

#### Metódusok

* **void addCollectable(Coordina coordinate Collectable newCollenctable):** Gyűjthető elem hozzáadása.
* **void addToRandomCoordinate():** Gyűjthető elem hozzáadása a pálya egy random pontjára.
* **void GetCollectableAt(Coordinate coordinate):** Gyűjthető elem felvétele.
* **boolean isThere(Coordinate coordinate):** Megadja, hogy van-e az adott koordinátán gyűjthető elem.
* **void setData(DataAccessPoint dataAccessPoint):** A GameObjects.ZPM generáláshoz ismernie kell a fieldObjectek és box-ok helyét.
* **int ZPMCount():** Visszaadja a pályán lévő ZPM-ek számát.

### Collectable

#### Felelősség

Interface a gyűjthető osztályok számára.

#### Ősosztályok: Nincs

#### Metódusok

* **void Collect()**: Az interface-t megvalósító objektum begyűjtésekor hívódik meg.

### Coordinate

#### Felelősség

*Tárolja x, y koordinátákat.*

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **x**: int: x koordinátát tárolja.
* **y**: int: y koordinátát tárolja.

#### Metódusok

* **void Set(int sx, int sy)**: Beállítja erre a pozícióra koordinátákat.
* **int GetX()**: Visszaadja x értékét.
* **void GetY()**: Visszaadja y értékét.

### CVector

#### Felelősség

*Tárolja x, y koordinátákat, és az irányt ahova néz az adott objektum.*

#### Ősosztályok: Coordinate

#### Interfészek: Nincs

#### Attribútumok

* **dir**: Direction: Az irányt tárolja.

#### Metódusok

* **void SetDir(int sx, int sy, Direction sd)**: Beállítja erre a pozícióra koordinátákat.
* **Direction GetDir()**: Visszaadja az irányt.

### DataAccessPoint

#### Felelősség

Hozzáférés biztosítása a StepHandler, a ShotHandler és a GrabHandler számára a FieldObjectInventory-hoz, BoxInventory-hoz és a CollectableInventory-hoz.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **field**: FieldObjectInventory: Referencia a FieldObject-ek tárolójára
* **Box**: BoxInventory: Referencia a mozgatható objektumok tárolójára.
* **collectable**: CollectableInventory: Referencia gyűjthető objektumok tárolójára

#### Metódusok: Nincs

### Door

#### Felelősség

Az ajtót megvalósító osztály. Felelőssége, hogy tudja, hogy milyen állapotban van. Zárt állapotban nem lehet átmenni rajta, így átlőni sem. Nyílt állapotban pedig át lehet menni rajta és így át is lehet lőni a nyitott ajtón. Semelyik állapotában nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok

* **open: boolean**

#### Metódusok

* **void** **Statechanged()**: Megváltoztatja az ajtó állapotát. Ha nyitva volt, akkor becsukja, ha zárva volt, akkor kinyitja.

#### boolean Steppable(): Rá lehet-e lépni az objektumra.

#### boolean Shootable(): Lőhető-e az objektumra csillagkapu.

#### boolean IsMortal(): Halálos-e az objektum.

#### boolean isDoor(): Ajtó-e az objektum.

### FieldObject

#### Felelősség

Interface, amely gondoskodik arról, hogy azokról az osztályokról, amik megvalósítják, el lehessen dönteni, hogy átjárhatók-e, át lehet-e lőni felettük, létre lehet-e hozni felette csillagkaput, illetve halálosak-e.

#### Ősosztályok: Nincs

#### Metódusok

#### boolean Steppable(): Rá lehet-e lépni az objektumra.

#### boolean Shootable(): Lőhető-e az objektumra csillagkapu.

#### boolean IsMortal(): Halálos-e az objektum.

#### boolean isDoor(): Ajtó-e az objektum.

### FieldObjectInventory

#### Felelősség

FieldObject interface-et megvalósító objektumok tárolása.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **fieldobj: HashMap<Coordinate, FieldObject>**

#### Metódusok

* **FieldObject GetFieldObject(Coordinate coord)**: A paraméter listán átvett koordinátán található FieldObject-et visszaadja a HashMap-ből, ha nincs a megodott koordinátán FieldObject, akkor null-t ad vissza.

### GrabHandler

#### Felelősség

A mozgatható objektumok mozgatásáról gondoskodik.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **sgates**: StarGateInventory: Nyilvántartja a csillagkapukat és pozíciójukat.
* **data**: DataAcessPoint: Adat-összeköttetést biztosít.
* **weight:** ButtonHandler: Kezeli a gombok (mérlegek) lenyomását.

#### Metódusok

* **boolean Grab(CVector interactfrom)**: CVectorral adjuk meg, hogy honnan milyen irányból akarunk felvenni, true-t ad vissza, ha a felvétel sikerült.
* **boolean Put(CVector interactfrom)**: CVectorral adjuk meg, hogy honnan milyen irányból akarunk letenni, true-t ad vissza, ha a letételsikerült.

### Player

#### Felelősség

Tárolja a Player (O’Neill ezredes) állapotát: pozícióját, irányát és hogy van-e valami a kezében. Ezeket változtatja a megfelelő függvényekkel.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **handfull**: boolean: Van-e valami mozgatható a Player kezében.
* **pos:** CVector: Megadja a Player pozícióját és irányát.

#### Metódusok

* **CVector GetPos()**: Visszaadja a Player pozícióját, irányát.
* **void SetPos(CVector ppos)**: Megváltoztatja a Player pozícióját, irányát.
* **boolean GetInHand()**: Visszaadja a handfull attribútum értékét.
* **void SetInHand(boolean b)**: Megváltoztatja a handfull attribútum értékét.

### PlayerActionHandler

#### Felelősség

Bemenet szerint továbbítja az események vezérlését, melyek a játékoshoz köthetőek.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **leg**: StepHandler: Kezeli a Player lépéseit.
* **gun**: ShootHandler: Kezeli a Player lövését.
* **hand**: GrabHandler: Kezeli a mozgatható objektumok felszedését.
* **pushing**: ButtonHandler: Kezeli a gombok (mérlegek) lenyomását.
* **data:** Player: Tárolja a Player állapotát.

#### Metódusok

* **void Interact(Direction toDir)**: Megpróbál felvenni vagy lerakni abba az irányba.
* **void shoot(Direction toDir, StarGatColor color)**:Lő abba az irányba ilyen színű Csillagkaput.
* **void Step(Direction toDir)**: Megpróbál lépni abba az irányba.

### ShootHandler

#### Felelősség

Lekezeli a játékos lövéseit.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **sgates**: StarGateHandler: Kezeli a csillagkapukat.
* **data**: DataAcessPoint: Adat-összeköttetést biztosít.

#### Metódusok

* **void Shoot(CVector shootfrom, Direction color)**: CVectorral átadjuk a pozíciónkat és hogy onnan milyen irányból akarunk lőni. Kiszámolja a becsapódás helyét, ha shootable, vezérléstovábbadással gondoskodik az esetleges csillagkapukról.

### SpecialWall

#### Felelősség

Speciális falat megvalósító osztály. Biztosítja, hogy nem átjárható, nem lehet átlőni felette. Nem halálos. Abban különbözik a Wall-tól, hogy létrejöhet rajta csillagkapu.

#### Ősosztályok: Wall

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok: Nincs

### StarGate

#### Felelősség

Maga a csillagkapu objektum, nyilvántartja a színét.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **color**: StarGateColor: Tárolja a csillagkapuk színét.
* **pos:** CVector: Tárolja a csillagkapuk pozícióját.

#### Metódusok

### StarGateInventory

#### Felelősség

Nyilvántartja a csillagkapukat.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **sgates**: StarGate: Tárolja a csillagkapukat.
* **weight:** ButtonHandler: Kezeli a gombok (mérlegek) lenyomását.

#### Metódusok

* **boolean IsThere(CVector from)**: Visszaadja, hogy az adott helyről érkezve van-e csillagkapu.
* **CVector StepIn(CVector from)**: Adott helyről belép a csillagkapuba, visszaadja a következő pozíciót.

### StepHandler

#### Felelősség

Lekezeli a játékos lépéseit, információt szolgáltat a mezők járhatóságáról.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok:

* **sgates**: StarGateInventory: Nyilvántartja a csillagkapukat és pozíciójukat.
* **data**: DataAcessPoint: Adat-összeköttetést biztosít.

#### Metódusok

* **CVector Nextstep(CVector playerpos, Direction dir)**: Visszaadja a Player pozícióját, annak függvényében, hogy tudott-e lépni vagy sem.

### Wall

#### Felelősség

A falat megvalósító osztály. Biztosítja, hogy nem lehet átmenni, átlőni rajta. Nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok: Nincs

### Way

#### Felelősség

Az utat megvalósító osztály. Átjárható, át lehet lőni felette, nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok: Nincs

### ZPM

#### Felelősség

A játék során begyűjtendő ZPM-et megvalósító osztály.

#### Ősosztályok: Nincs

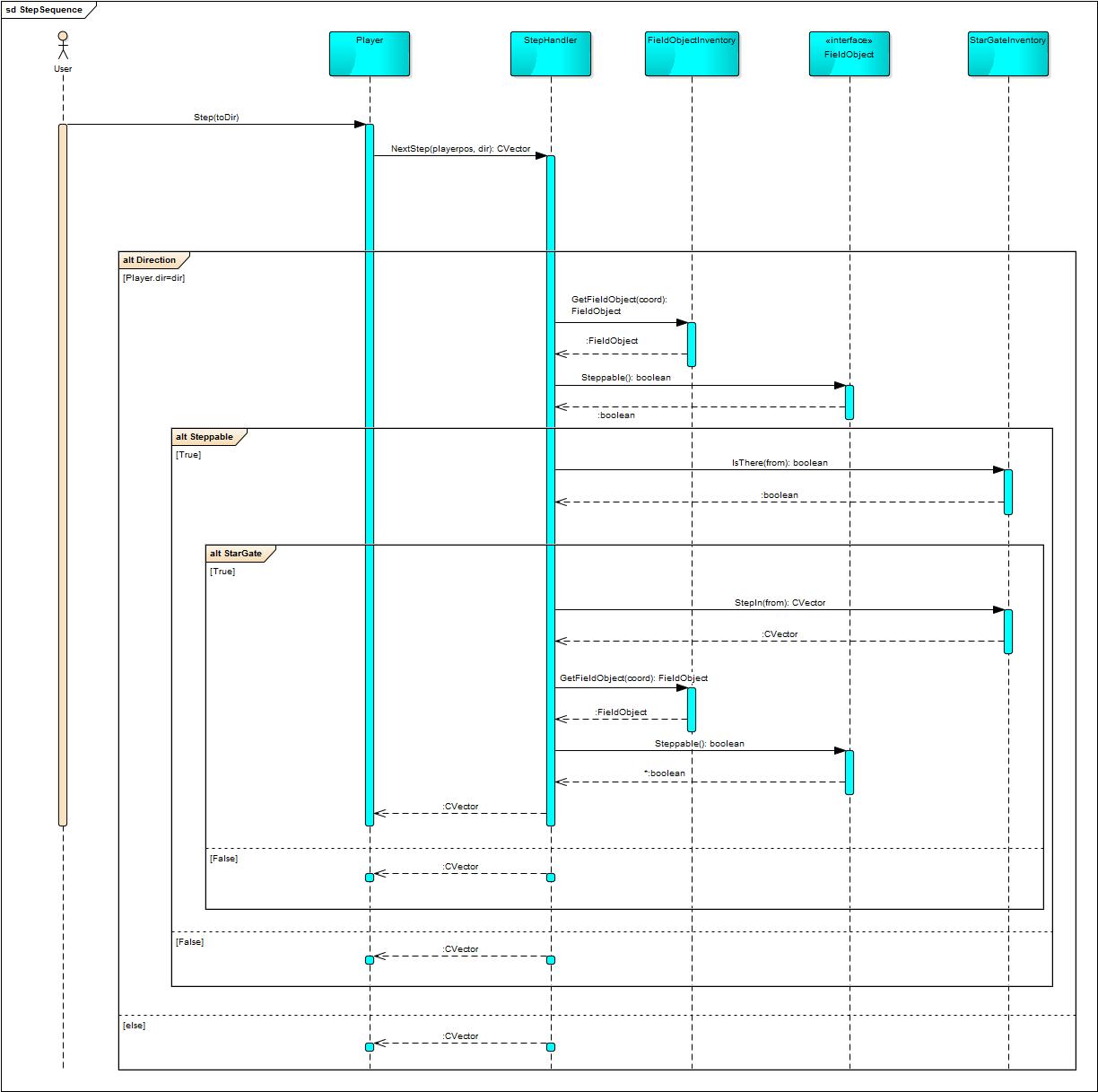
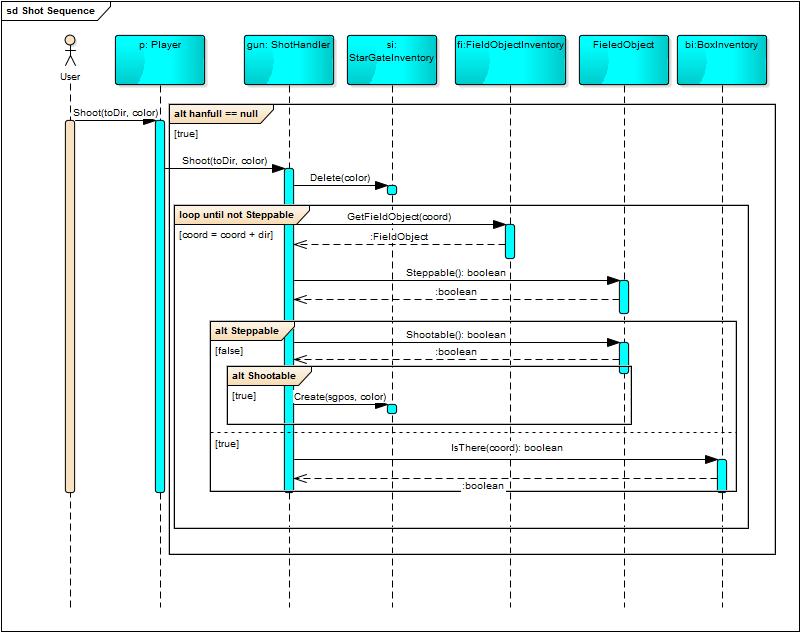
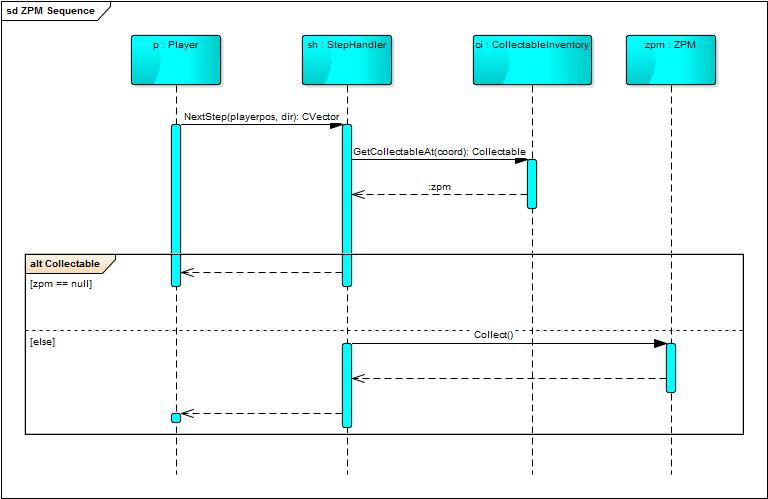
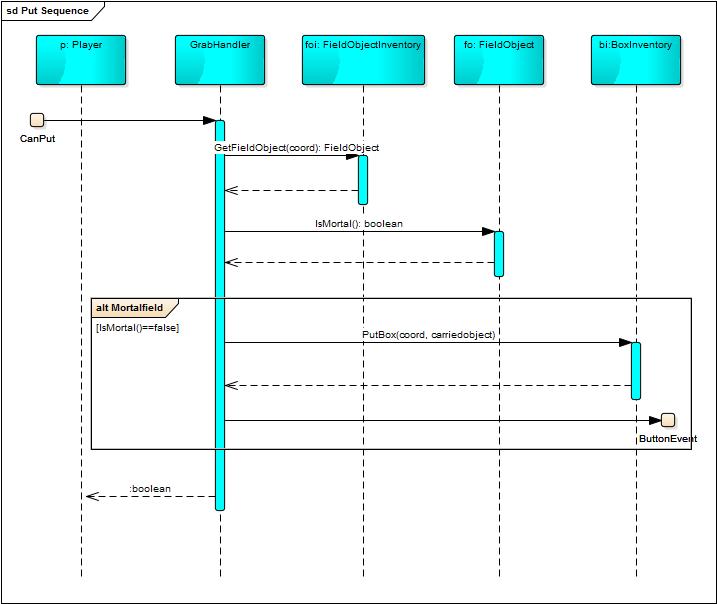
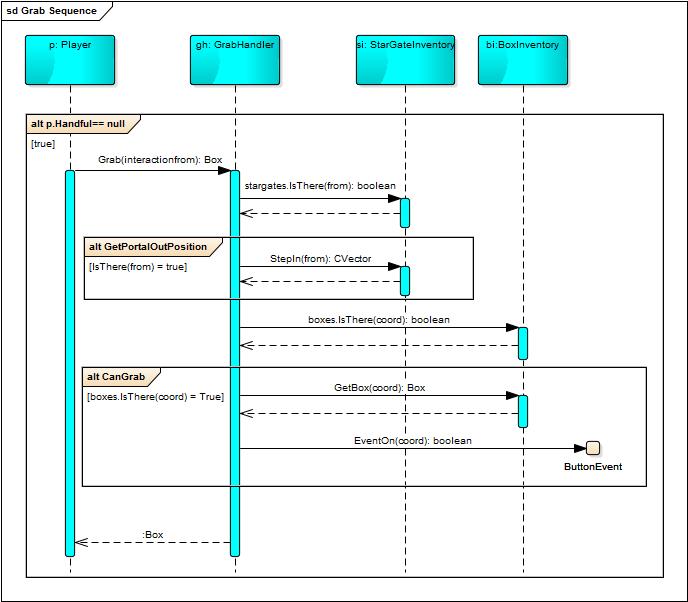
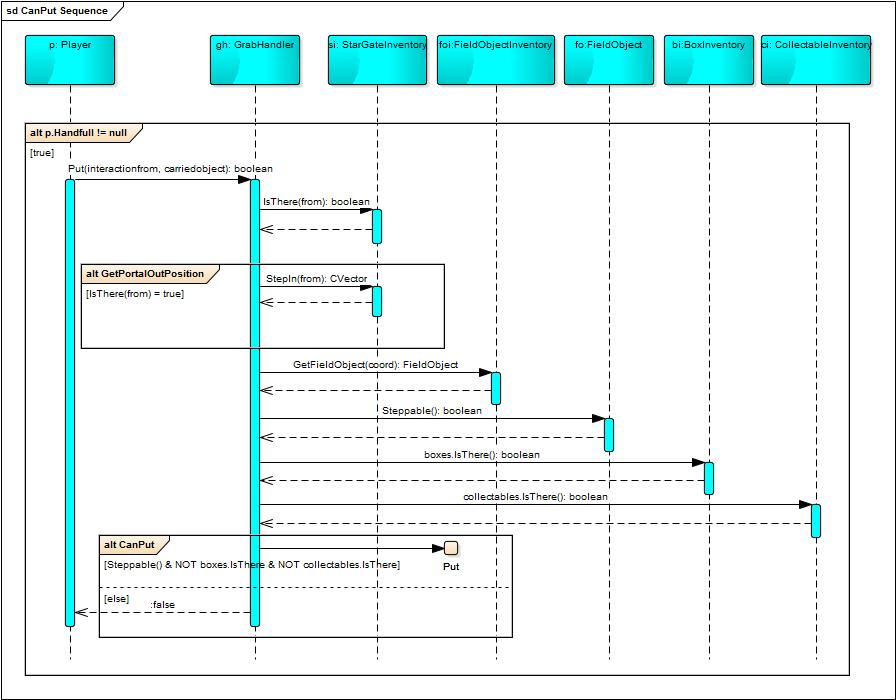
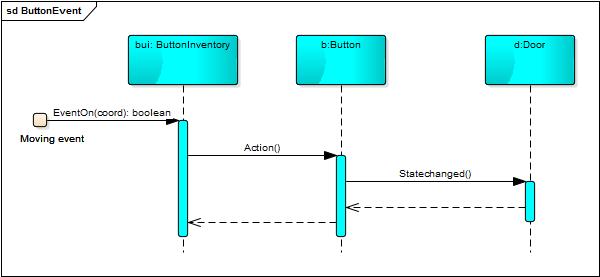
#### Interfészek

* Collectable

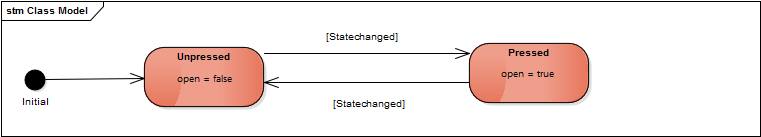
#### Attribútumok: Nincs

#### Metódusok: Nincs

## Szekvencia diagramok



## State-chartok



## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2016.03.12. 16:00 | 2,5 óra | Zöld, Borbély, Ritter, Papp | Értekezlet. Osztálydiagram módosítása. |
| 2016.03.12. 21:00 | 0,5 óra | Zöld | Osztálydiagram véglegesítése, ZPM szekvencia |
| 2016.03.12. 13:00 | 1 óra | Makki | Step szekvencia |
| 2016.03.13. 18:00 | 1 óra | Ritter | Shoot szekvencia |
| 2016.03.13. 19:00 | 1 óra | Papp | Box put szekvencia |
| 2016.03.13. 19:30 | 1 óra | Gáspár | Box grab szekvencia |
| 2016.03.13. 20:00 | 0,5 óra | Borbély | Objektumok, Osztályok |

# Szkeleton tervezése

18 – don

Konzulens:

Ferencz Endre

Csapattagok

|  |  |  |
| --- | --- | --- |
| Borbély László Benedek | V19RPS | [laci.dj@gmail.com](mailto:laci.dj@gmail.com) |
| Gáspár Vilmos | GF4UZL | [gaspar.vilmos24@gmail.com](mailto:gaspar.vilmos24@gmail.com) |
| Makki Viktor | I0UA10 | [deldemonho@gmail.com](mailto:deldemonho@gmail.com) |
| Ritter Alex | GS9ERJ | [ritt.alex@gmail.com](mailto:ritt.alex@gmail.com) |
| Papp Albert | F9YTAY | [albertxpapp@gmail.com](mailto:albertxpapp@gmail.com) |
| Zöld Máté | GQH7FW | [mate951006@gmail.com](mailto:mate951006@gmail.com) |
|  |  |  |

2016. március 21.

# Szkeleton tervezése

## A szkeleton modell valóságos use-case-ei

### Use-case diagram

### Skeleton Usecase

### Use-case leírások

|  |  |
| --- | --- |
| **Use-case neve** | Init |
| **Rövid leírás** | Betöltődik a pálya, a játékos felkerül a pályára. |
| **Aktorok** | User |
| **Forgatókönyv** | Feltölti a megfelelő konténereket a hozzá tartozó elemekkel. A játékost felkonfigurálja. |

|  |  |
| --- | --- |
| **Use-case neve** | Ütközés detektálás Dobozlerakáskor |
| **Rövid leírás** | Megvizsgálja, hogy az adott mezőre lehet-e dobozt helyezni. |
| **Aktorok** | User |
| **Forgatókönyv** | Le kell ellenőrizni, hogy van-e a játékos kezében doboz, majd ha igen akkor, hogy a szomszédos mező ahova rakni akarjuk a dobozt ott portál található-e, ha igen a vizsgált mezőnek nem a szomszéd mezőt, hanem a portál kimenetét kell választani.  Le kell ellenőrizni, hogy ráléphető-e a mező, van-e rajta doboz, vagy ZPM, ha ez igaz le lehet rakni az adott koordinátára a dobozt. |

|  |  |
| --- | --- |
| **Use-case neve** | Dobozlerakás |
| **Rövid leírás** | A játékos kezében lévő dobozt lerakja egy szomszédos mezőre |
| **Aktorok** | User |
| **Forgatókönyv** | Meg kell nézni, hogy a mező, amire rakjuk a dobozt elpusztítja-e, ha igen akkor a dobozt kikell törölni.  Ellenkező esetben lerakja a dobozt az adott pontra. |

|  |  |
| --- | --- |
| **Use-case neve** | Doboz felvétele |
| **Rövid leírás** | A játékos felveszi egy szomszédos mezőn lévő dobozt. |
| **Aktorok** | User |
| **Forgatókönyv** | Miután a játékos jelezte, hogy fel szeretne venni egy dobozt a pályáról, meg kell nézni, hogy a mező, amiről fel akarja venni, azon portál áll-e, ha igen, akkor a vizsgált mezőnek nem a szomszédos mezőnek, hanem a portál kimenetelének kell lennie. Majd meg kell nézni, hogy van-e a vizsgált mezőn doboz, ha van, akkor azt fel kell onnan venni, majd ha volt gomb alatta, annak az állapotát meg kell változtatni. |

|  |  |
| --- | --- |
| **Use-case neve** | Lövés |
| **Rövid leírás** | A játékos lő a fegyverével egy irányba, majd a lövedék, ami az adott irányban belecsapódik a legközelebbi akadályba. |
| **Aktorok** | User |
| **Forgatókönyv** | Miután a játékos jelezte, hogy az adott irányba lőni szeretne, a pozíciójától kezdve az adott irányba végig kell nézni a mezőket, amíg olyan mezőt nem talál, amin nem tud áthaladni a töltény. Tehát megnézzük, hogy az adott mező léphető, ha igen akkor meg kell nézni, hogy van-e rajta doboz; mivel a töltény nem tud nem léphető mezőn se dobozon áthaladni. Ha töltény nem léphető mezőbe akad meg akkor meg kell nézni, hogy létrehozható rajta portál, ha igen akkor ezt elhelyezzük a pályán. |

|  |  |
| --- | --- |
| **Use-case neve** | Gombnyomás |
| **Rövid leírás** | Egy gombon lévő súlymennyiség megváltozik, majd a gombhoz tartozó ajtó ehhez mérten állapotot változtat. |
| **Aktorok** | User |
| **Forgatókönyv** | Mikor a játékos egy szomszédos mezőre lép sikeresen, vagy dobozt helyez, levesz róla, meghívódik a gombokat tároló konténer, a mező koordinátáival, ami leellenőrzi, hogy van- e gomb azon a mezőn, ha van, akkor meghívja az adott gomb állapot változtató függvényét. A gomb felnyitja vagy bezárja az ajtót. |

|  |  |
| --- | --- |
| **Use-case neve** | Járható mezőre lépés |
| **Rövid leírás** | A játékos átlép egy szomszédos járható mezőre. |
| **Aktorok** | User |
| **Forgatókönyv** | Miután a játékos jelezte, hogy szeretne a szomszédos mezőre lépni, le kell ellenőrizni, hogy azon mezőn van-e portál, ha van, akkor a vizsgált mezőnek nem a szomszédos mezőnek, hanem a portál kimenetének kell lennie.  Le kell ellenőrizni, hogy a vizsgált mező léphető és hogy nem foglalja-e már doboz. Ha szabad a mező, akkor sikeres a lépés, ekkor, ha a mezőn, amin álltunk van gomb akkor annak az állapotát meg kell változtatni, illetve a mezőn, amire lépünk van-e ZPM, akkor azt felvesszük, illetve ha van rajta, lenyomjuk a gombot. |

## A szkeleton kezelői felületének terve, dialógusok

A szkeleton kezelői felületét szöveges utasításokkal lehet vezérelni egy kezdeti menün keresztül.

A menü kinézete:

1: Init

2: Box put/grab

3: Shoot

4: Step

q: Quit

A szekvenciadiagramoknak megfelelő, az alábbihoz hasonló kimenetek fognak keletkezni:

>>Player:: Shoot(toDir, color)

>>ShotHandler:: Delete(color)

>>ShotHandler:: GetfieldObject(coord)

<< ShotHandler:: GetfieldObject(coord)

>>ShotHandler:: Steppable()

<< ShotHandler:: Steppable()

>>ShotHandler:: Shootable()

<< ShotHandler:: Shootable()

>>Shothandler:: Create(sgpos, color)

A „>>” jelöli a függvény meghívását illetve „<<” a függvény visszatérését.

A futás során a következő példának megfelelő üzeneteket küld a program a standard outputra.

Init:

>>FieldObjectInventory::Create()

<<FieldObjectInventory::Create()

>>Wall::Wall()

<<Wall::Wall()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>Wall::Wall()

<<Wall::Wall()

>>SpecialWall::SpecialWall()

<<SpecialWall::SpecialWall()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>Abyss::Abyss()

<<Abyss::Abyss()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>Way::Way()

<<Way::Way()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>Door::Door()

<<Door::Door()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>Box::Box()

<<Box::Box()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>StarGateInventory::StarGateInventory()

<<StarGateInventory::StarGateInventory()

>>BoxInventory::BoxInventory()

<<BoxInventory::BoxInventory()

>>BoxInventory::addBox(Box newVal)

<<BoxInventory::addBox(Box newVal)

>>Button::Button(Door door)

<<Button::Button(Door door)

>>ButtonInventory::ButtonInventory()

<<ButtonInventory::ButtonInventory()

>>ButtonInventory::addButton(Button newVal)

<<ButtonInventory::addButton(Button newVal)

>>CollectableInventory::CollectableInventory()

<<CollectableInventory::CollectableInventory()

>>ZPM::ZPM()

<<ZPM::ZPM()

>>CollectableInventory::addCollectable(Collectable newVal)

<<CollectableInventory::addCollectable(Collectable newVal)

>>DataAccesspoint::Create(StarGateInventory stargates, FieldObjectInventory fields,BoxInventory boxes, ButtonInventory buttons, CollectableInventory collectables)

<<DataAccesspoint::Create(StarGateInventory stargates, FieldObjectInventory fields,BoxInventory boxes, ButtonInventory buttons, CollectableInventory collectables)

>>ShotHandler::Create(DataAccessPoint data)

<<ShotHandler::Create(DataAccessPoint data)

>>GrabHandler::Create(DataAccessPoint data)

<<GrabHandler::Create(DataAccessPoint data)

>>StepHandler::Create(DataAccessPoint data)

<<StepHandler::Create(DataAccessPoint data)

>>Player::Create(ShotHandler shothandler, GrabHandler grabhandler, StepHandler stephandler)

>>CVector::Create(int sx, int sy, Direction dir)

<<CVector::Create(int sx, int sy, Direction dir)

<<Player::Create(ShotHandler shothandler, GrabHandler grabhandler, StepHandler stephandler)

<<Game::Create()

Továbbá, egyes eldöntendő kérdéseket az alkalmazás futása közben a felhasználónak feltehet, hiszen a program jelenlegi állapotában nem tartalmaz elkészített pályamodellt, így az egyes funkciók tesztelése csakis felhasználói beavatkozással érhető el.

Step:

>>Player::Step(Direction toDir)

>>StepHandler::NextStep(CVectro playerpos, Direction dir)

>>StarGateInventoy::IsThere(CVectro from)

Is there a StarGate? (y/n)

n

<<StarGateInventoy::IsThere(CVectro from)

>>FieldObjectInventory::GetFieldObject(Coordinate coord)

<<FieldObjectInventory::GetFieldObject(Coordinate coord)

>>FieldObject::Steppable()

Is the FieldObject steppable? (y/n)

y

<<FieldObject::Steppable()

>>BoxInventory::IsThere(Coordinate coord)

Is there a box? (y/n)

n

<<BoxInventory::IsThere(Coordinate coord)

>>ButtonInventory::EventOn(Coordinate coord)

Is there a button item? (y/n)

n

<<ButtonInventory::EventOn(Coordinate coord)

>>FieldObjectInventory::GetFieldObject(Coordinate coord)

<<FieldObjectInventory::GetFieldObject(Coordinate coord)

>>FieldObject::IsMortal()

Is the FieldObject deadly? (y/n)

n

<<FieldObject::IsMortal()

>>CollectableInventory::IsThere(Coordinate coord)

Is there a collectable item? (y/n)

n

<<CollectableInventory::IsThere(Coordinate coord)

>>ButtonInventory::EventOn(Coordinate coord)

Is there a button item? (y/n)

n

<<ButtonInventory::EventOn(Coordinate coord)

<<StepHandler::NextStep(CVectro playerpos, Direction dir)

<<Player::Step(Direction toDir)

Waitin' for command!

4

>>Player::Step(Direction toDir)

>>StepHandler::NextStep(CVectro playerpos, Direction dir)

>>StarGateInventoy::IsThere(CVectro from)

Is there a StarGate? (y/n)

y

<<StarGateInventoy::IsThere(CVectro from)

>>StarGateInventory::StepIn(CVector from)

<<StarGateInventory::StepIn(CVector from)

>>FieldObjectInventory::GetFieldObject(Coordinate coord)

<<FieldObjectInventory::GetFieldObject(Coordinate coord)

>>FieldObject::Steppable()

Is the FieldObject steppable? (y/n)

y

<<FieldObject::Steppable()

>>BoxInventory::IsThere(Coordinate coord)

Is there a box? (y/n)

n

<<BoxInventory::IsThere(Coordinate coord)

>>ButtonInventory::EventOn(Coordinate coord)

Is there a button item? (y/n)

y

>>Button::Action()

>>Door::Statechanged()

<<Door::Statechanged()

<<Button::Action()

<<ButtonInventory::EventOn(Coordinate coord)

>>FieldObjectInventory::GetFieldObject(Coordinate coord)

<<FieldObjectInventory::GetFieldObject(Coordinate coord)

>>FieldObject::IsMortal()

Is the FieldObject deadly? (y/n)

n

<<FieldObject::IsMortal()

>>CollectableInventory::IsThere(Coordinate coord)

Is there a collectable item? (y/n)

y

<<CollectableInventory::IsThere(Coordinate coord)

>>ButtonInventory::EventOn(Coordinate coord)

Is there a button item? (y/n)

y

>>Button::Action()

>>Door::Statechanged()

<<Door::Statechanged()

<<Button::Action()

<<ButtonInventory::EventOn(Coordinate coord)

<<StepHandler::NextStep(CVectro playerpos, Direction dir)

<<Player::Step(Direction toDir)

Box put/grab

>>Player::Interact(Direction toDir)

Does the player have a box in hand? (y/n)

y

>>GrabHandler::Put(CVectro interactfrom, Box carrieobject)

>>StarGateInventoy::IsThere(CVectro from)

Is there a StarGate? (y/n)

y

<<StarGateInventoy::IsThere(CVectro from)

>>StarGateInventory::StepIn(CVector from)

<<StarGateInventory::StepIn(CVector from)

>>FieldObjectInventory::GetFieldObject(Coordinate coord)

<<FieldObjectInventory::GetFieldObject(Coordinate coord)

>>FieldObject::Steppable()

Is the FieldObject steppable? (y/n)

y

<<FieldObject::Steppable()

>>BoxInventory::IsThere(Coordinate coord)

Is there a box? (y/n)

n

<<BoxInventory::IsThere(Coordinate coord)

>>CollectableInventory::IsThere(Coordinate coord)

Is there a collectable item? (y/n)

n

<<CollectableInventory::IsThere(Coordinate coord)

>>BoxInventory::PutBox(Coordinate coord, Box carriedobject)

<<BoxInventory::PutBox(Coordinate coord, Box carriedobject)

>>ButtonInventory::EventOn(Coordinate coord)

Is there a button item? (y/n)

y

>>Button::Action()

>>Door::Statechanged()

<<Door::Statechanged()

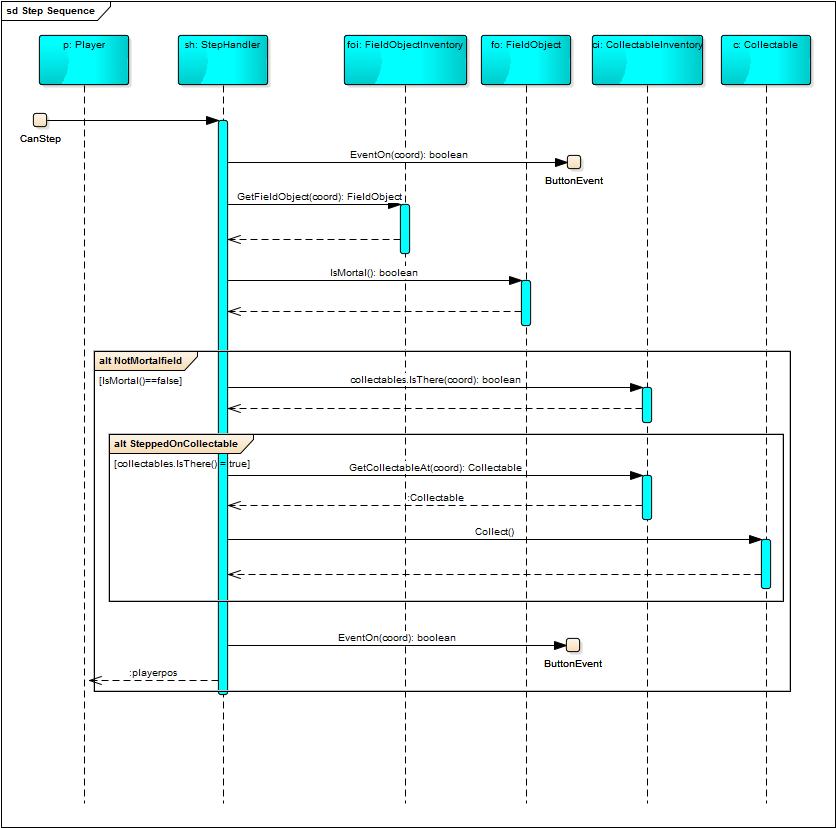
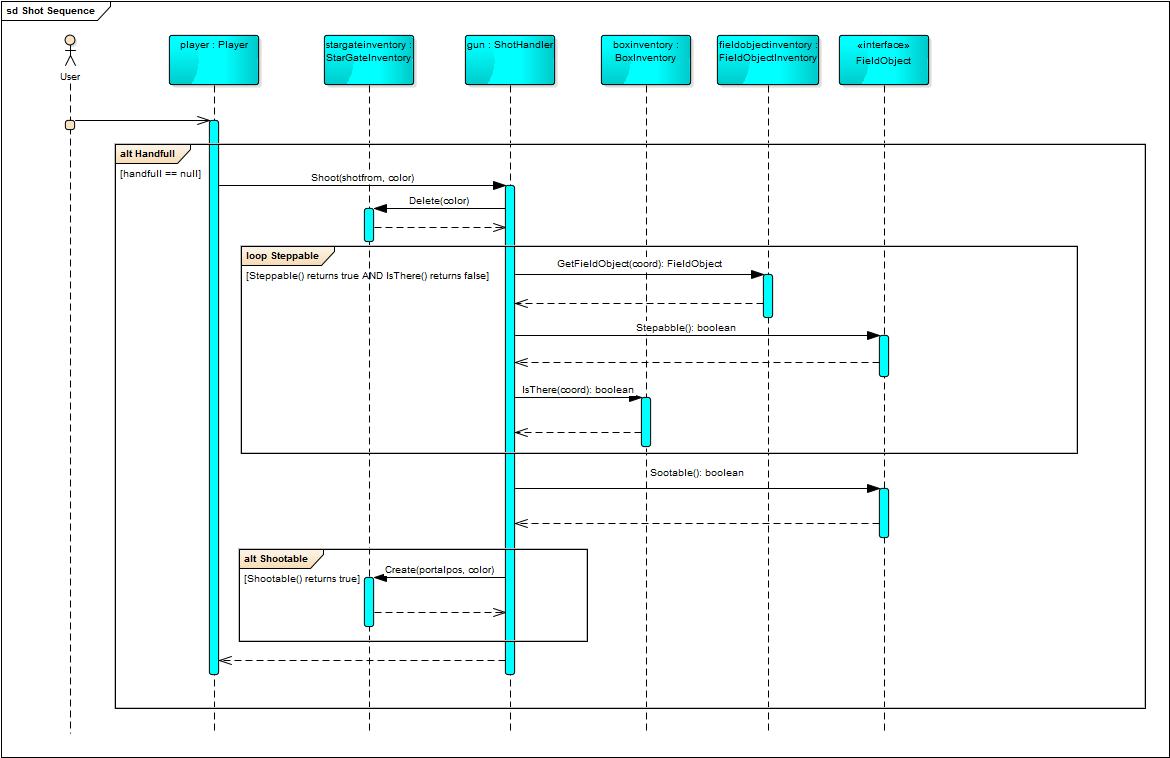
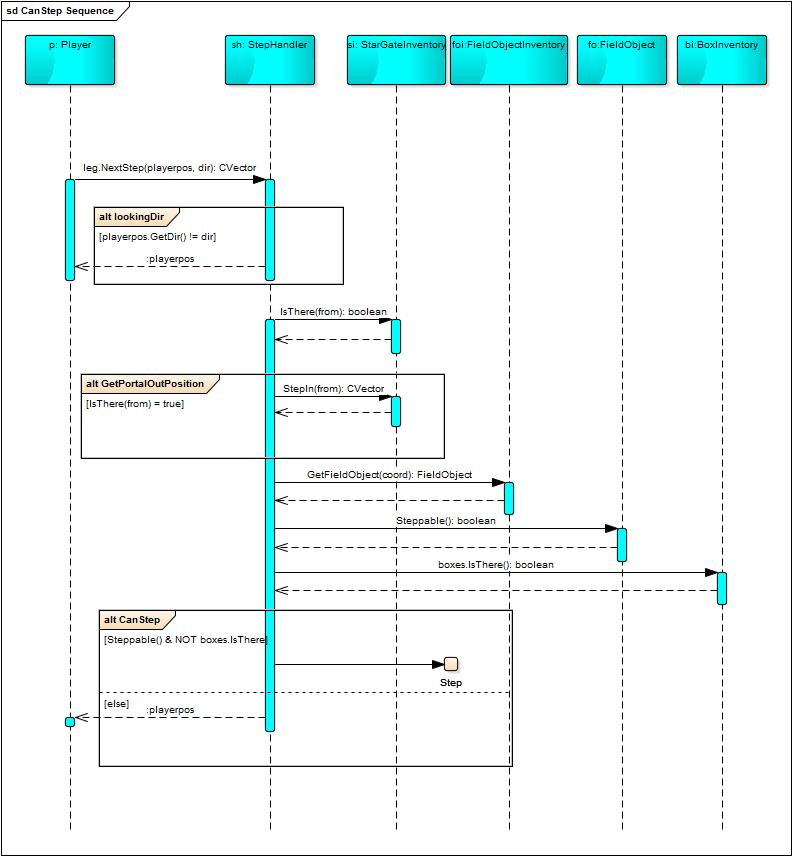
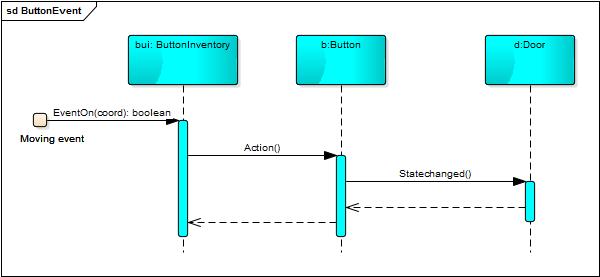
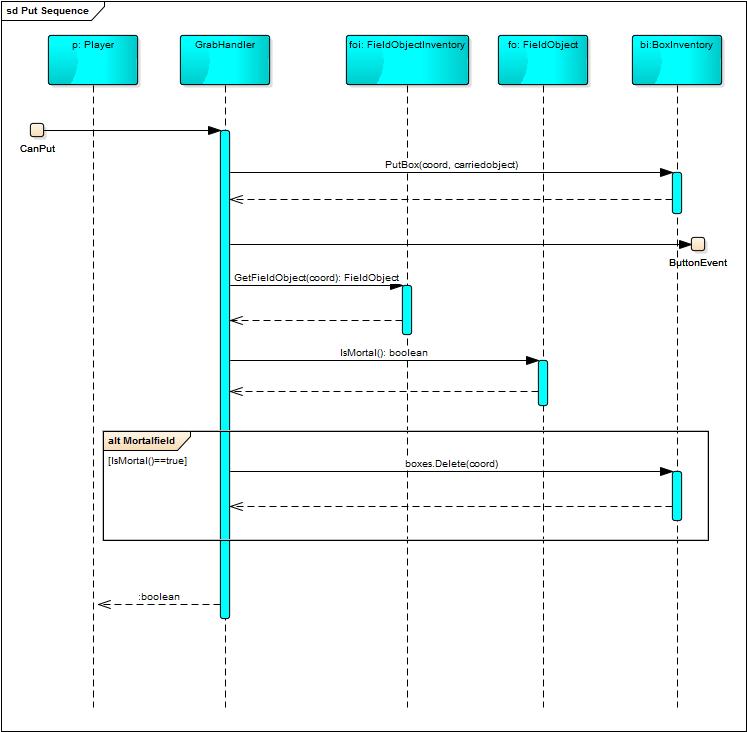
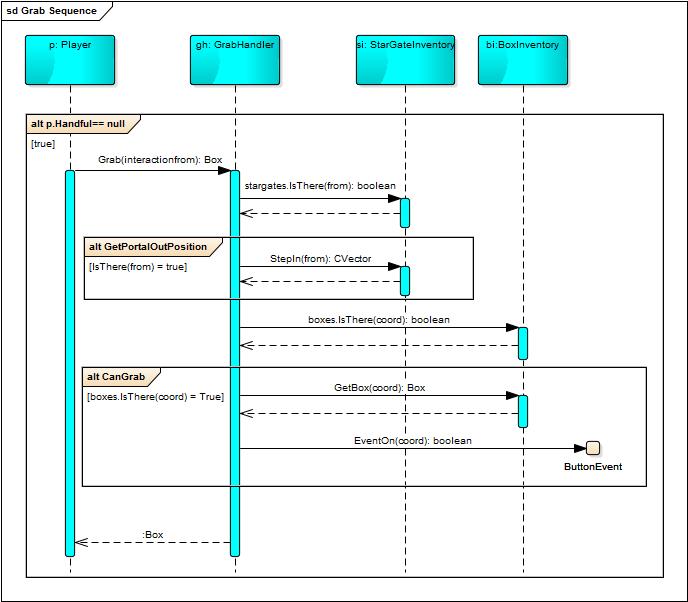
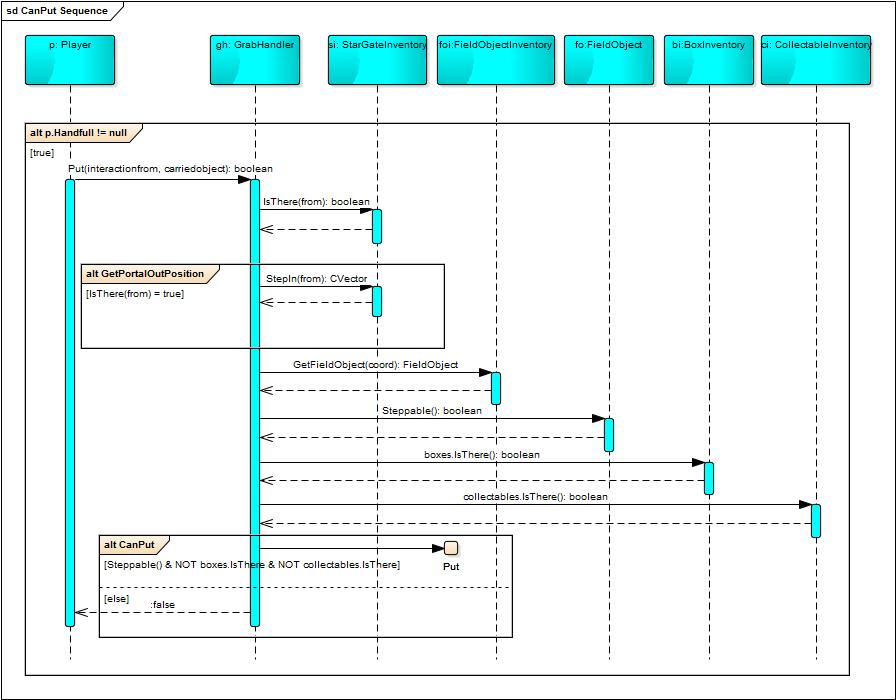
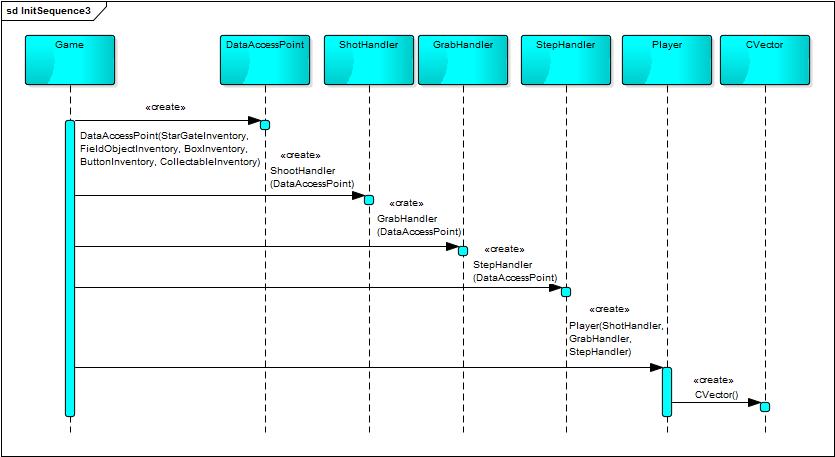
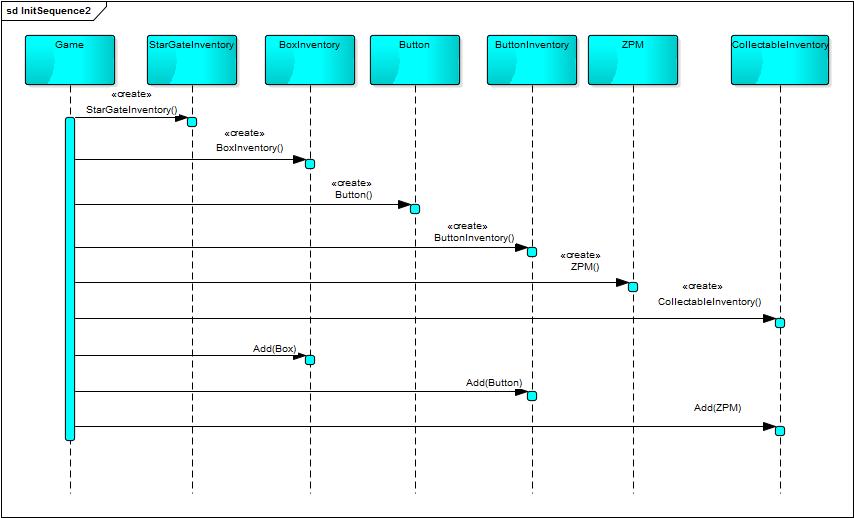
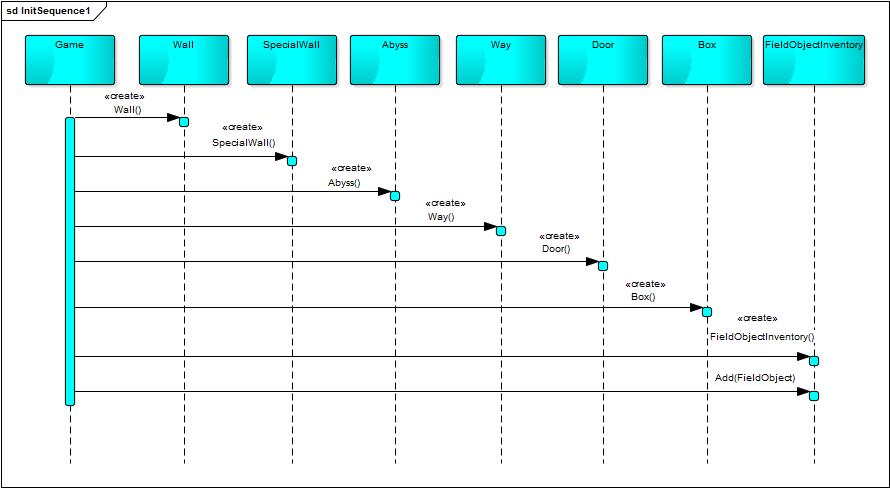
<<Button::Action()

<<ButtonInventory::EventOn(Coordinate coord)

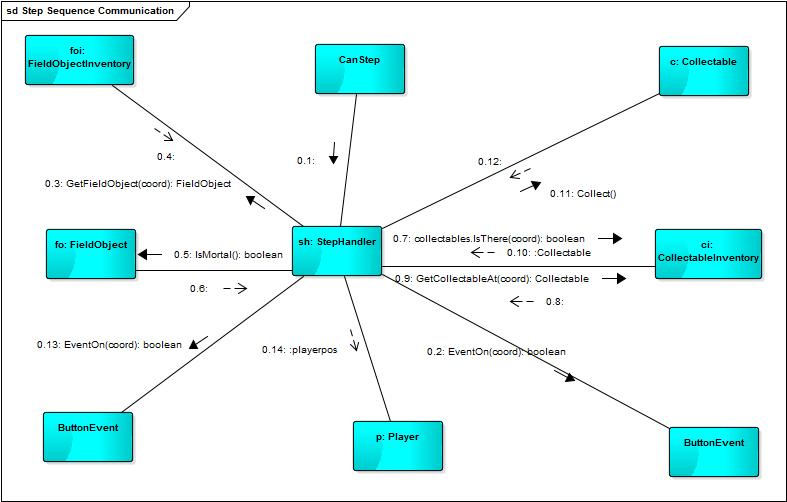
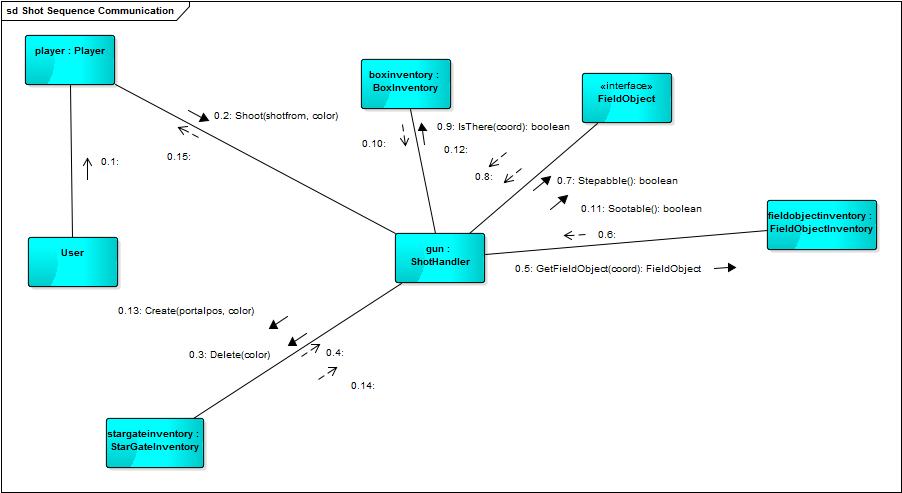
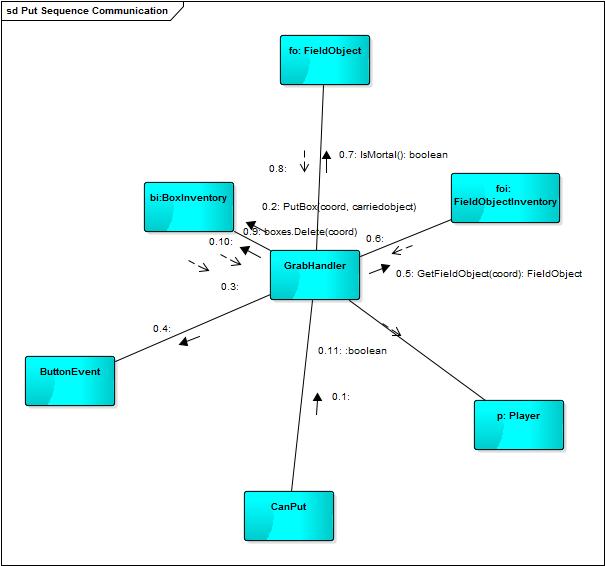
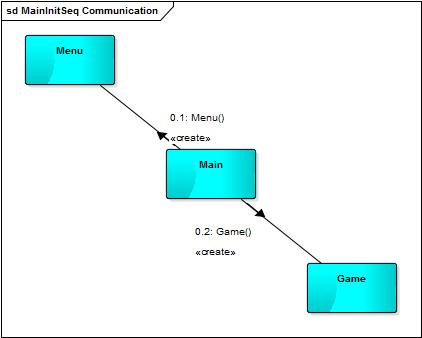
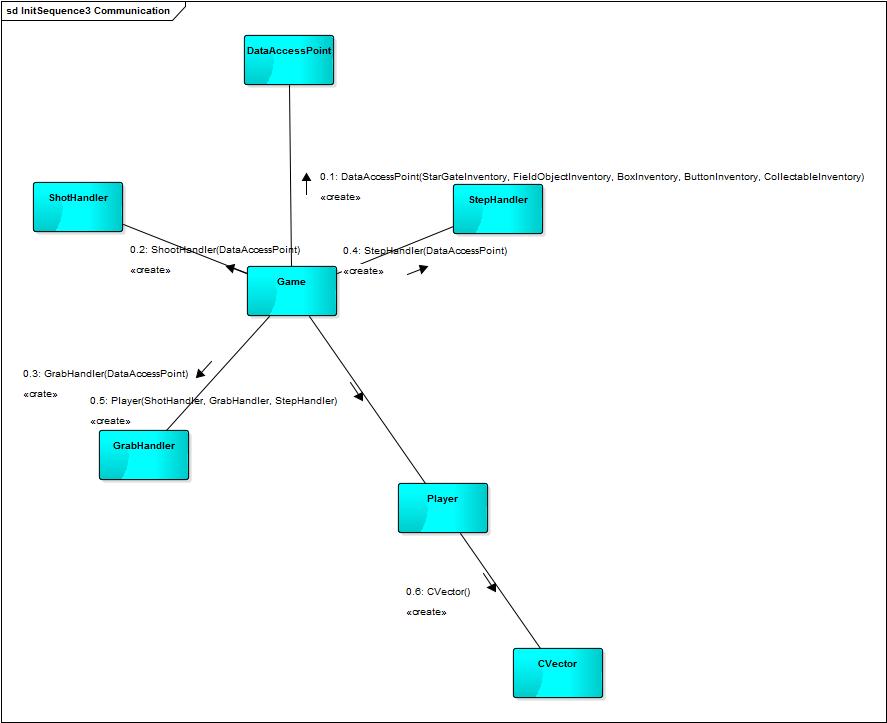
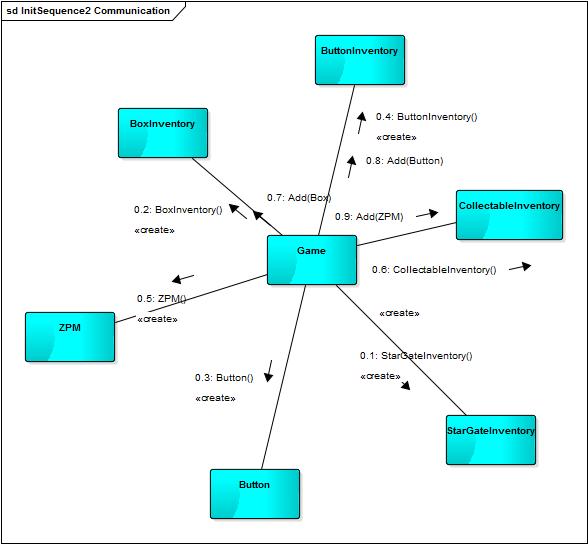
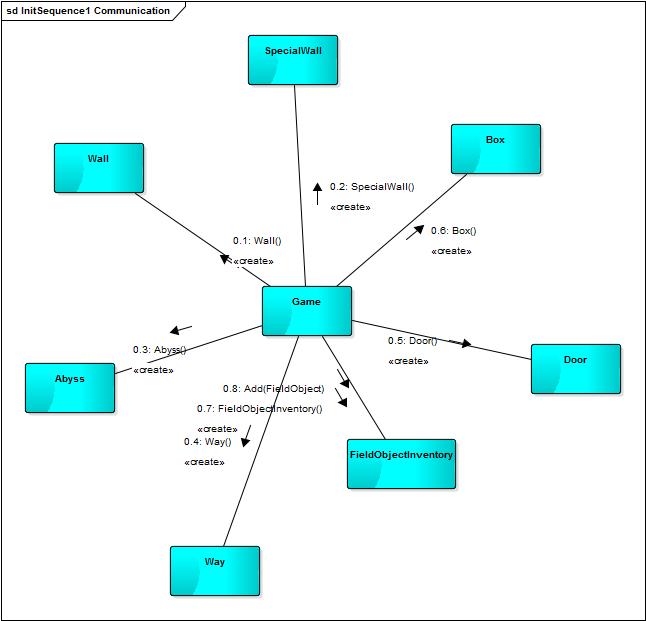
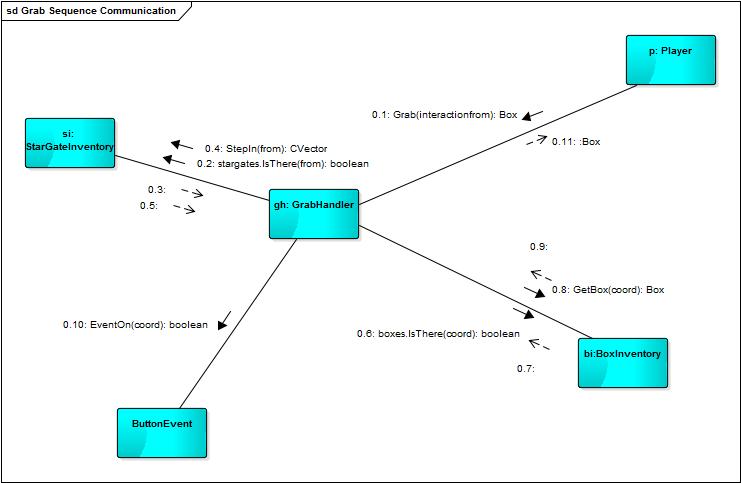
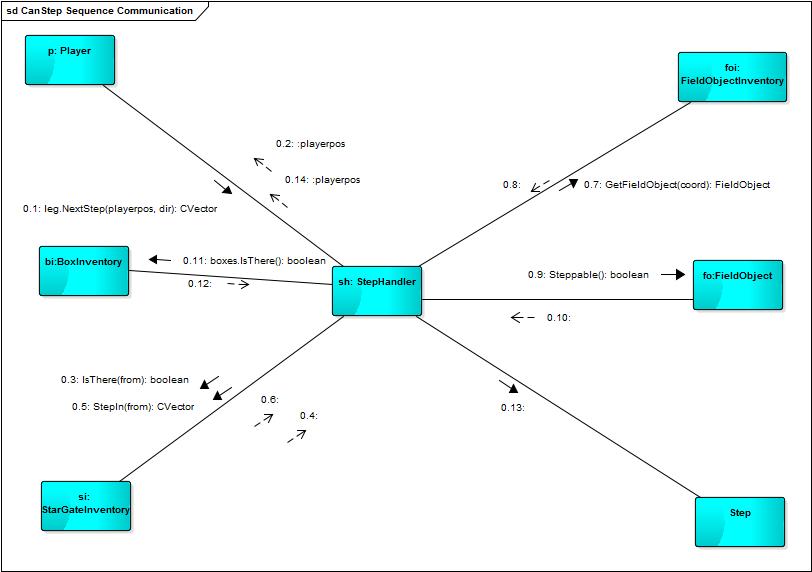
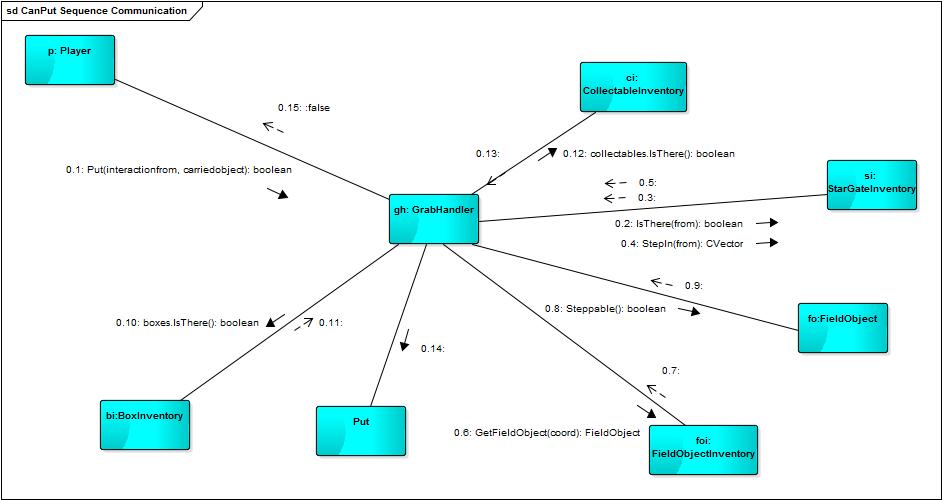
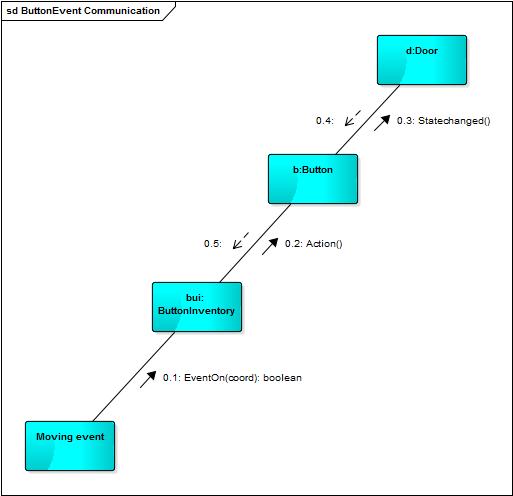
<<GrabHandler::Put(CVectro interactfrom, Box carrieobject)

<<Player::Interact(Direction toDir)

## Szekvencia diagramok a belső működésre



## Kommunikációs diagramok



## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2016.03.20. 18:00 | 1 óra | Zöld | Szekvencia diagramok újragondolása |
| 2016.03.20. 19:00 | 1,5 óra | Gáspár | Szekvencia diagramok szerkesztése |
| 2016.03.20. 20:00 | 1 óra | Makki | Init szekvenciák |
| 2016.03.20. 20:30 | 1 óra | Papp | Use-Case-ek |
| 2016.03.21. 00:00 | 1 óra | Ritter | Kezelői felület |
| 2016.03.21. 01:00 | 1,5 óra | Borbély | Kommunikációs diagramok |

# Szkeleton beadás

18 – don

Konzulens:

Ferencz Endre

Csapattagok

|  |  |  |
| --- | --- | --- |
| Borbély László Benedek | V19RPS | [laci.dj@gmail.com](mailto:laci.dj@gmail.com) |
| Gáspár Vilmos | GF4UZL | [gaspar.vilmos24@gmail.com](mailto:gaspar.vilmos24@gmail.com) |
| Makki Viktor | I0UA10 | [deldemonho@gmail.com](mailto:deldemonho@gmail.com) |
| Ritter Alex | GS9ERJ | [ritt.alex@gmail.com](mailto:ritt.alex@gmail.com) |
| Papp Albert | F9YTAY | [albertxpapp@gmail.com](mailto:albertxpapp@gmail.com) |
| Zöld Máté | GQH7FW | [mate951006@gmail.com](mailto:mate951006@gmail.com) |
|  |  |  |

2016. március 29.

# Szkeleton beadás

## Fordítási és futtatási útmutató

### Fájllista

|  |  |  |  |
| --- | --- | --- | --- |
| **Fájl neve** | **Méret** | **Keletkezés ideje** | **Tartalom** |
| Abyss.java | 1 KB | 2016.03.27 | Szakadék tulajdonságai |
| Box.java | 1 KB | 2016.03.27 | Doboz tulajdonságai |
| BoxInventory.java | 2 KB | 2016.03.27 | Dobozok tárolása |
| Button.java | 1 KB | 2016.03.27 | Nyomólapok tulajdonságai |
| ButtonInventory.java | 2 KB | 2016.03.27 | Nyomólapok tárolása |
| Collectable.java | 1 KB | 2016.03.27 | Felszedhető objektumok tulajdonságai |
| CollectableInventory.java | 2 KB | 2016.03.27 | Felszedhető objektumok tárolása |
| Coordinate.java | 1 KB | 2016.03.27 | Koordináták tárolása |
| Cvector.java | 1 KB | 2016.03.27 | Koordináták iránnyal tárolása |
| DataAcessPoint.java | 1 KB | 2016.03.27 | Adat összekötetést biztosít |
| Direction.java | 1 KB | 2016.03.27 | Irány enum |
| Door.java | 1 KB | 2016.03.27 | Ajtó tulajdonságai |
| FieldObject.java | 1 KB | 2016.03.27 | FieldObject tulajdonságai |
| FieldObjectInventory.java | 2 KB | 2016.03.27 | FieldObject tárolása |
| FieldObjectSample.java | 2 KB | 2016.03.27 | FieldObject általánosan |
| GameClass.java | 5 KB | 2016.03.28 | Menu (main) |
| GrabHandler.java | 3 KB | 2016.03.27 | Objektum felvétének ill. lerakásának esetei lekezelése |
| MANIFEST.MF | 1 KB | 2016.03.28 | Main file megadása |
| Player.java | 3 KB | 2016.03.27 | Játékos interakciói |
| QuestionAssistant.java | 1 KB | 2016.03.27 | Eldöntető kérdésre form |
| ShotHandler.java | 2 KB | 2016.03.27 | Lövés eseteinek lekezelése |
| SpecialWall.java | 1 KB | 2016.03.27 | Speciális fal tulajdonságai |
| Stargate.java | 1 KB | 2016.03.27 | Csillagkapu tulajdonságai |
| StargateColor.java | 1 KB | 2016.03.27 | Csillagkapuszíne enum |
| StargateInventory.java | 2 KB | 2016.03.27 | Csillagkapu tárolása |
| StepHandler.java | 2 KB | 2016.03.27 | Lépés eseteinek lekezelése |
| Wall.java | 1 KB | 2016.03.27 | Fal tulajdonságai |
| Way.java | 1 KB | 2016.03.27 | Út tulajdonságai |
| ZPM.java | 1 KB | 2016.03.27 | ZPM tulajdonságai |

### Fordítás

Az előre legenerált manifest fájl és a forrásfájlok segítségével parancssorból a következő parancsokat kell kiadni:

>javac \*.java

>jar cfm Game.jar MANIFEST.MF \*.class

Így a forrásfájlokból egyetlenegy futtatható jar fájl készül.

### Futtatás

|  |  |  |
| --- | --- | --- |
| **Azonosító** | **Leírás** | **Komment** |
| E1. | Java Runtime Environment 1.8 | Mivel a szoftvernek a HSZK gépein futtathatónak kell lenniük, ezért az említettnél frissebb és régebbi verzió használata nem ajánlott. |
| E2. | Billentyűzet | A szkeleton futása csak billentyűzetről vezérelhető. |

## A futtatáshoz a fordítási utasításokon túl még a következő parancsokat kell kiadni:

>java –jar Game.jar

## Értékelés

|  |  |
| --- | --- |
| **Tag neve** | **Munka százalékban** |
| Gáspár | 17% |
| Borbély | 16% |
| Makki | 16,75% |
| Zöld | 16,75% |
| Ritter | 16,75% |
| Papp | 16,75% |

## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2016.03.26. 18:00 | 3 óra | Zöld | Box put és függvényei |
| 2016.03.26. 20:00 | 2 óra | Makki | Init inplementálása |
| 2016.03.26. 21:00 | 1 óra | Papp | Step és Button implementálás |
| 2016.03.27. 14:00 | 1 óra | Ritter | Kezelői felület és Fájllista |
| 2016.03.27. 18:00 | 1 óra | Gáspár | Kódok felülvizsgálata |
| 2016.03.27. 23:00 | 3 óra | Borbély | Grab implementálás és kódvéglegesítés  Shott implementálása |
| 2016.03.28. 10:00 | 2 óra | Borbély | Dokumentum korrektúra |

## A szkeleton kezelői felületének terve, dialógusok

A szkeleton kezelői felületét szöveges utasításokkal lehet vezérelni egy kezdeti menün keresztül.

A menü kinézete:

1: Init

2: Box put/grab

3: Shoot

4: Step

q: Quit

A szekvenciadiagramoknak megfelelő, az alábbihoz hasonló kimenetek fognak keletkezni:

>>Player:: Shoot(toDir, color)

>>ShotHandler:: Delete(color)

>>ShotHandler:: GetfieldObject(coord)

<< ShotHandler:: GetfieldObject(coord)

>>ShotHandler:: Steppable()

<< ShotHandler:: Steppable()

>>ShotHandler:: Shootable()

<< ShotHandler:: Shootable()

>>Shothandler:: Create(sgpos, color)

A „>>” jelöli a függvény meghívását illetve „<<” a függvény visszatérését.

A futás során a következő példának megfelelő üzeneteket küld a program a standard outputra.

Init:

>>FieldObjectInventory::Create()

<<FieldObjectInventory::Create()

>>Wall::Wall()

<<Wall::Wall()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>Wall::Wall()

<<Wall::Wall()

>>SpecialWall::SpecialWall()

<<SpecialWall::SpecialWall()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>Abyss::Abyss()

<<Abyss::Abyss()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>Way::Way()

<<Way::Way()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>Door::Door()

<<Door::Door()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>Box::Box()

<<Box::Box()

>>FieldObjectInventory::addFieldObject(FieldObject newVal)

<<FieldObjectInventory::addFieldObject(FieldObject newVal)

>>StarGateInventory::StarGateInventory()

<<StarGateInventory::StarGateInventory()

>>BoxInventory::BoxInventory()

<<BoxInventory::BoxInventory()

>>BoxInventory::addBox(Box newVal)

<<BoxInventory::addBox(Box newVal)

>>Button::Button(Door door)

<<Button::Button(Door door)

>>ButtonInventory::ButtonInventory()

<<ButtonInventory::ButtonInventory()

>>ButtonInventory::addButton(Button newVal)

<<ButtonInventory::addButton(Button newVal)

>>CollectableInventory::CollectableInventory()

<<CollectableInventory::CollectableInventory()

>>ZPM::ZPM()

<<ZPM::ZPM()

>>CollectableInventory::addCollectable(Collectable newVal)

<<CollectableInventory::addCollectable(Collectable newVal)

>>DataAccesspoint::Create(StarGateInventory stargates, FieldObjectInventory fields,BoxInventory boxes, ButtonInventory buttons, CollectableInventory collectables)

<<DataAccesspoint::Create(StarGateInventory stargates, FieldObjectInventory fields,BoxInventory boxes, ButtonInventory buttons, CollectableInventory collectables)

>>ShotHandler::Create(DataAccessPoint data)

<<ShotHandler::Create(DataAccessPoint data)

>>GrabHandler::Create(DataAccessPoint data)

<<GrabHandler::Create(DataAccessPoint data)

>>StepHandler::Create(DataAccessPoint data)

<<StepHandler::Create(DataAccessPoint data)

>>Player::Create(ShotHandler shothandler, GrabHandler grabhandler, StepHandler stephandler)

>>CVector::Create(int sx, int sy, Direction dir)

<<CVector::Create(int sx, int sy, Direction dir)

<<Player::Create(ShotHandler shothandler, GrabHandler grabhandler, StepHandler stephandler)

<<Game::Create()

Továbbá, egyes eldöntendő kérdéseket az alkalmazás futása közben a felhasználónak feltehet, hiszen a program jelenlegi állapotában nem tartalmaz elkészített pályamodellt, így az egyes funkciók tesztelése csakis felhasználói beavatkozással érhető el.

Step:

>>Player::Step(Direction toDir)

>>StepHandler::NextStep(CVectro playerpos, Direction dir)

>>StarGateInventoy::IsThere(CVectro from)

Is there a StarGate? (y/n)

n

<<StarGateInventoy::IsThere(CVectro from)

>>FieldObjectInventory::GetFieldObject(Coordinate coord)

<<FieldObjectInventory::GetFieldObject(Coordinate coord)

>>FieldObject::Steppable()

Is the FieldObject steppable? (y/n)

y

<<FieldObject::Steppable()

>>BoxInventory::IsThere(Coordinate coord)

Is there a box? (y/n)

n

<<BoxInventory::IsThere(Coordinate coord)

>>ButtonInventory::EventOn(Coordinate coord)

Is there a button item? (y/n)

Is there a button item? (y/n)

n

<<ButtonInventory::EventOn(Coordinate coord)

>>FieldObjectInventory::GetFieldObject(Coordinate coord)

<<FieldObjectInventory::GetFieldObject(Coordinate coord)

>>FieldObject::IsMortal()

Is the FieldObject deadly? (y/n)

n

<<FieldObject::IsMortal()

>>CollectableInventory::IsThere(Coordinate coord)

Is there a collectable item? (y/n)

n

<<CollectableInventory::IsThere(Coordinate coord)

>>ButtonInventory::EventOn(Coordinate coord)

Is there a button item? (y/n)

n

<<ButtonInventory::EventOn(Coordinate coord)

<<StepHandler::NextStep(CVectro playerpos, Direction dir)

<<Player::Step(Direction toDir)

Waitin' for command!

4

>>Player::Step(Direction toDir)

>>StepHandler::NextStep(CVectro playerpos, Direction dir)

>>StarGateInventoy::IsThere(CVectro from)

Is there a StarGate? (y/n)

y

<<StarGateInventoy::IsThere(CVectro from)

>>StarGateInventory::StepIn(CVector from)

<<StarGateInventory::StepIn(CVector from)

>>FieldObjectInventory::GetFieldObject(Coordinate coord)

<<FieldObjectInventory::GetFieldObject(Coordinate coord)

>>FieldObject::Steppable()

Is the FieldObject steppable? (y/n)

y

<<FieldObject::Steppable()

>>BoxInventory::IsThere(Coordinate coord)

Is there a box? (y/n)

n

<<BoxInventory::IsThere(Coordinate coord)

>>ButtonInventory::EventOn(Coordinate coord)

Is there a button item? (y/n)

y

>>Button::Action()

>>Door::Statechanged()

<<Door::Statechanged()

<<Button::Action()

<<ButtonInventory::EventOn(Coordinate coord)

>>FieldObjectInventory::GetFieldObject(Coordinate coord)

<<FieldObjectInventory::GetFieldObject(Coordinate coord)

>>FieldObject::IsMortal()

Is the FieldObject deadly? (y/n)

n

<<FieldObject::IsMortal()

>>CollectableInventory::IsThere(Coordinate coord)

Is there a collectable item? (y/n)

y

<<CollectableInventory::IsThere(Coordinate coord)

>>ButtonInventory::EventOn(Coordinate coord)

Is there a button item? (y/n)

y

>>Button::Action()

>>Door::Statechanged()

<<Door::Statechanged()

<<Button::Action()

<<ButtonInventory::EventOn(Coordinate coord)

<<StepHandler::NextStep(CVectro playerpos, Direction dir)

<<Player::Step(Direction toDir)

Box put/grab

>>Player::Interact(Direction toDir)

Does the player have a box in hand? (y/n)

y

>>GrabHandler::Put(CVectro interactfrom, Box carrieobject)

>>StarGateInventoy::IsThere(CVectro from)

Is there a StarGate? (y/n)

y

<<StarGateInventoy::IsThere(CVectro from)

>>StarGateInventory::StepIn(CVector from)

<<StarGateInventory::StepIn(CVector from)

>>FieldObjectInventory::GetFieldObject(Coordinate coord)

<<FieldObjectInventory::GetFieldObject(Coordinate coord)

>>FieldObject::Steppable()

Is the FieldObject steppable? (y/n)

y

<<FieldObject::Steppable()

>>BoxInventory::IsThere(Coordinate coord)

Is there a box? (y/n)

n

<<BoxInventory::IsThere(Coordinate coord)

>>CollectableInventory::IsThere(Coordinate coord)

Is there a collectable item? (y/n)

n

<<CollectableInventory::IsThere(Coordinate coord)

>>BoxInventory::PutBox(Coordinate coord, Box carriedobject)

<<BoxInventory::PutBox(Coordinate coord, Box carriedobject)

>>ButtonInventory::EventOn(Coordinate coord)

Is there a button item? (y/n)

y

>>Button::Action()

>>Door::Statechanged()

<<Door::Statechanged()

<<Button::Action()

<<ButtonInventory::EventOn(Coordinate coord)

<<GrabHandler::Put(CVectro interactfrom, Box carrieobject)

# Prototípus koncepciója

18 – don

Konzulens:

Ferencz Endre

Csapattagok

|  |  |  |
| --- | --- | --- |
| Borbély László Benedek | V19RPS | [laci.dj@gmail.com](mailto:laci.dj@gmail.com) |
| Gáspár Vilmos | GF4UZL | [gaspar.vilmos24@gmail.com](mailto:gaspar.vilmos24@gmail.com) |
| Makki Viktor | I0UA10 | [deldemonho@gmail.com](mailto:deldemonho@gmail.com) |
| Ritter Alex | GS9ERJ | [ritt.alex@gmail.com](mailto:ritt.alex@gmail.com) |
| Papp Albert | F9YTAY | [albertxpapp@gmail.com](mailto:albertxpapp@gmail.com) |
| Zöld Máté | GQH7FW | [mate951006@gmail.com](mailto:mate951006@gmail.com) |
|  |  |  |

2016. április 4.

# Prototípus koncepciója

## Prototípus interface-definíciója

### Az interfész általános leírása

A prototípus interfésze szabványos bemenetről fogad parancsokat és a szabványos kimentre írja ki az ezekre adott válaszokat, így parancssorból futtatható. A bemenete átirányítható a tesztelést támogató segédprogram segítségével ki- és bemeneti fájlokra. A segédprogrammal előre megírt teszteseteket lehet futtatni, amik megkönnyítik a tesztelést, mert a bementeket nem kézzel kell megadni a programnak, hanem a bemeneti fájlok tartalmazzák azokat. A tesztesetek automatikusan lefutnak, a bemenetekre (parancsokra) adott válaszokat pedig a kimeneti fájl(ok)ba írja a program. Az a teszteset sikeres, amelyre a leírt elvárt kimenettel megegyező kimenetet kapunk.

### Bemeneti nyelv

A bemeneten megadható parancsokat a következő formátumban fogadja el a program:

**<aktor> <utasítás> <paraméter>**

**Colonel Step <irány>**

**Leírás:** Az ezredes irányítására szolgáló parancs. A megadott irányba próbál meg lépni az ezredes.

**Opciók:** East, North, West, South

**Colonel Shoot <szín>**

**Leírás:** Az ezredes által lőhető portálok irányítására szolgáló parancs. A megadott írányba próbál meg lőni az ezredes.

**Opciók:** Yellow, Blue

**Colonel Interact**

**Leírás:** Az ezredes által dobozok felvételére és lerakására szolgáló parancs, mely az ezredes pozíciójának megfelelően hajtja végre az utasítást.

**Jaffa Step <irány>**

**Leírás:** Jaffa irányítására szolgáló parancs. A megadott irányba próbál meg lépni Jaffa.

**Opciók:** East, North, West, South

**Jaffa Shoot <szín>**

**Leírás:** A Jaffa által lőhető portálok irányítására szolgáló parancs. A megadott írányba próbál meg lőni Jaffa.

**Opciók:** Red, Green

**Jaffa Interact**

**Leírás:** Jaffa által dobozok felvételére és lerakására szolgáló parancs, mely a pozíciójának megfelelően hajtja végre az utasítást.

**Replicator Step <irány>**

**Leírás:** Az replikátor irányítására szolgáló parancs. A megadott irányba próbál meg lépni a replikátor.

**Opciók:** East, North, West, South

**A pályát megadó fájlok nyelvtana:**

A beolvasó metódus úgy párosítja az ajtót a mérlehez, hogy a fájlban a mérleg közvetlenül a hozzá tartozó ajtó után helyezkedik el.

FieldObjectTípus ; y koordináta ; x koordináta

FieldObjectTípus2 ; y koordináta ; x koordináta

Ajtó ; y koordináta ; x koordináta

Mérleg ; y koordináta ; x koordináta ; Mérleg súlya

.

.

.

### Kimeneti nyelv

Minden végrehajtott művelet után a kimeneten, arra egy válasz érkezik a következő formában:

**Step:**

* <Aktor> new position: <x> <y>, [Game Over] [a door opened | a door closed] [way spawned] [ZPM collected] [ZPM spawned]

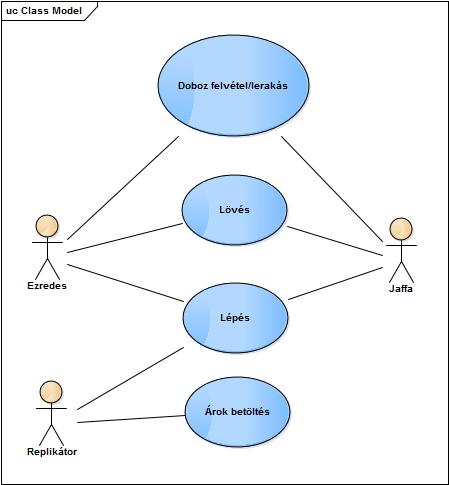
**Shoot:**

* <Aktor> <<PortalColor> portal at <x> <y> | failed to create portal | hit enemy at <x> <y>>

**Interact:**

* **<**Actor**> <**has grabbed | has put**>** an item at <x> <y> [ a door opened | a door closed | box destroyed]
* **<**Actor**>** can’tinteract at <x> <y>
* **<**Actor>’s hand is full, interact failed

## Összes részletes use-case



|  |  |
| --- | --- |
| **Use-case neve** | Doboz lerakás |
| **Rövid leírás** | Megvizsgálja, hogy az adott mezőre lehet-e dobozt helyezni. |
| **Aktorok** | Jaffa, Ezredes |
| **Forgatókönyv** | Miután a Jaffa, vagy az ezredes jelezte az igényt a doboz lerakásra; le kell ellenőrizni, hogy van-e a játékos kezében doboz, majd ha igen akkor, hogy a szomszédos mező ahova rakni akarjuk a dobozt ott portál található-e, ha igen a vizsgált mezőnek nem a szomszéd mezőt, hanem a portál kimenetét kell választani.  Le kell ellenőrizni, hogy ráléphető-e a mező, és nincs-e rajta doboz, ZPM, replikátor vagy egy másik játékos, ha ez igaz le lehet rakni az adott koordinátára a dobozt. |

|  |  |
| --- | --- |
| **Use-case neve** | Doboz felvétele |
| **Rövid leírás** | A játékos felveszi egy szomszédos mezőn lévő dobozt. |
| **Aktorok** | Jaffa, Ezredes |
| **Forgatókönyv** | Miután a Jaffa vagy az ezredes jelezte az igényt, hogy fel szeretne venni egy dobozt a pályáról. Le kell ellenőrizni, hogy van-e a játékos kezében hely, ezután meg kell nézni, hogy a mező, amiről fel akarja venni, azon portál áll-e, ha igen, akkor a vizsgált mezőnek nem a szomszédos mezőnek, hanem a portál kimenetelének kell lennie. Majd meg kell nézni, hogy van-e a vizsgált mezőn doboz, ha van, akkor azt fel kell onnan venni, majd ha volt gomb alatta, a rá nehezedő nyomást csökkenteni kell a doboz súlyával. |

|  |  |
| --- | --- |
| **Use-case neve** | Lövés |
| **Rövid leírás** | A játékos lő a fegyverével egy irányba, majd a lövedék, ami az adott irányban belecsapódik a legközelebbi akadályba. |
| **Aktorok** | Jaffa, Ezredes |
| **Forgatókönyv** | Miután a Jaffa, vagy az ezredes jelezte az igényt a portál kilövésére, a pozíciójától kezdve az adott irányba végig kell nézni a mezőket, amíg olyan mezőt nem talál, amin nem tud áthaladni a töltény. Tehát megnézzük, hogy az adott mező léphető, ha igen akkor meg kell nézni, hogy van-e rajta doboz vagy replikátor; mivel a töltény nem tud nem léphető mezőn se dobozon se replikátoron áthaladni; ha replikátorban akad meg a portál, akkor a replikátort meg kell semmisíteni. Ha töltény nem léphető mezőbe akad meg akkor meg kell nézni, hogy létrehozható rajta portál, ha igen akkor ezt elhelyezzük a pályán. |

|  |  |
| --- | --- |
| **Use-case neve** | Lépés |
| **Rövid leírás** | A játékos átlép egy szomszédos járható mezőre. |
| **Aktorok** | Jaffa, Ezredes, Repliátor |
| **Forgatókönyv** | Miután a Jaffa, vagy az ezredes jelezte igényét a lépésre, le kell ellenőrizni, hogy azon mezőn van-e portál, ha van, akkor a vizsgált mezőnek nem a szomszédos mezőnek, hanem a portál kimenetének kell lennie.  Le kell ellenőrizni, hogy a vizsgált mező léphető és hogy nem foglalja-e már doboz, replikátor vagy másik játékos. Ha szabad a mező, akkor sikeres a lépés, ekkor, ha a mezőn, amin álltunk van gomb akkor annak az állapotát meg kell változtatni, illetve a mezőn, amire lépünk van-e ZPM, akkor azt felvesszük, illetve ha van rajta, a gombra nehezedő nyomást megváltoztatjuk a játékos súlyával. |

|  |  |
| --- | --- |
| **Use-case neve** | Árok betöltés |
| **Rövid leírás** | A replikátor beleesik a szakadékba, majd út lesz az helyén |
| **Aktorok** | Replikátor |
| **Forgatókönyv** | Miután megsemmisült egy szakadékba lépés következtében a replikátor, az adott mező koordinátájához tartozó szakadék kicserélődik egy járható útra. |

## Tesztelési terv

|  |  |
| --- | --- |
| **Teszt-eset neve** | Doboz felvétel dobozzal rendelkező mezőrő |
| **Rövid leírás** | Doboz felvételének tesztelése |
| **Teszt célja** | Teszteli, hogy játékos fel tud venni egy dobozt, ha a megfelelő mezőn tartózkodik (közvetlenül a doboz előtt áll). |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Doboz felvétel doboz nélküli mezőről |
| **Rövid leírás** | Doboz felvételének tesztelése |
| **Teszt célja** | Teszteli, hogy játékos fel tud venni egy dobozt, ha a nem megfelelő mezőn tartózkodik (nincs közvetlenül előtte doboz). |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Doboz felvétel, a játékos kezében már van doboz |
| **Rövid leírás** | Doboz felvételének tesztelése |
| **Teszt célja** | Teszteli, hogy játékos fel tud venni egy dobozt, ha a már a kezében is van egy, ha a felvételhez megfelelő mezőn tartózkodik. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Doboz felvétel mérlegről – ajtózárás |
| **Rövid leírás** | Doboz felvételének mérlegről történő tesztelése, amelynek következtében bezárja az ajtót. |
| **Teszt célja** | Teszteli, hogy mérlegről lekerülő doboz miatt a rajta lévő súly a súlykorlát alá kerül, így a mérleg bezárja a hozzá tartozó ajtót. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Doboz felvétel mérlegről – ajtó nyitva marad |
| **Rövid leírás** | Doboz felvételének mérlegről történő tesztelése, amelynek következtében az ajtó nyitva marad. |
| **Teszt célja** | Teszteli, hogy a mérlegről lekerülő doboz miatt csökken a rajta lévő súly, de a súlykorlát fölött marad, így a mérleghez tartozó ajtó nyitva marad. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Doboz lerakás mérlegre – ajtó kinyílik |
| **Rövid leírás** | Doboz lerakásának mérlegre történő, amelynek következtében a mérleghez tartozó ajtó kinyílik. |
| **Teszt célja** | Teszteli, hogy a mérlegre kerülő súly növeli a rajta lévő összsúlyt, ami eléri a súlykorlátot, így a mérleghez tartozó ajtó kinyílik. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Doboz lerakás mérlegre – ajtó zárva marad |
| **Rövid leírás** | Doboz lerakásának mérlegre történő tesztelése, amelynek következtében mérleghez tartozó ajtó zárva marad. |
| **Teszt célja** | Teszteli, hogy a mérlegre kerülő súly növeli a rajta lévő összsúlyt, de nem éri el a súlykorlátot, így a mérleghez tartozó ajtó zárva marad. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Doboz felvétel féregjáraton keresztül |
| **Rövid leírás** | A játékos a féregjárat egyik kapuja előtt áll és a másik kapu előtt található dobozt felveszi. |
| **Teszt célja** | Teszteli, hogy a játékos féregjáraton keresztül fel tud venni egy dobozt. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Doboz lerakás féregjáraton keresztül |
| **Rövid leírás** | A játékos a féregjárat egyik kapuja előtt áll és leteszi a dobozt a féregjárat végén található koordinátára. |
| **Teszt célja** | Teszteli, hogy a játékos le tud tenni egy dobozt féregjáraton keresztül. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lövés falra / zárt ajtóra |
| **Rövid leírás** | A játékos lövése falat vagy zárt ajtót talál el. |
| **Teszt célja** | Ellenőrzi, hogy amikor a játékos lő és a lövedék falat vagy zárt ajtót talál el, azon nem hoz létre csillagkaput. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lövés speciális falra |
| **Rövid leírás** | A játékos lövése speciális falat talál el. |
| **Teszt célja** | Ellenőrzi, hogy amikor a játékos lő és a lövése speciális falt talál el, azon létrehoz egy csillagkaput. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lövés dobozra |
| **Rövid leírás** | A játékos lövése dobozt talál el. |
| **Teszt célja** | Ellenőrzi, hogy amikor a játékos lő és a lövése dobozt talál el, azon nem hoz létre csillagkaput. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lövés replikátorra |
| **Rövid leírás** | A játékos lövése és replikátort talál el. |
| **Teszt célja** | Ellenőrzi, hogy amikor a játékos lő és a lövése replikátort talál el, akkor a replikátor elpusztul. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lépés útra/nyitott ajtóba |
| **Rövid leírás** | A játékos olyan koordinátára lép ahol út vagy nyitott ajtó van. |
| **Teszt célja** | Teszteli, hogy amikor a játékos útra vagy nyitott ajtóba lép, akkor nem történik semmi a játékos koordinátájának megváltozásán kívül. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lépés falra/spec falra/zárt ajtóra/dobozra/másik játékosra |
| **Rövid leírás** | A játékos lépése során olyan koordinátára akar lépni ahol fal vagy speciális fal vagy zárt ajtó vagy doboz, vagy a másik játékos van. |
| **Teszt célja** | Teszteli, hogy játékos nem tud a kívánt koordinátára lépni, mert a fenti objektumok egyik ott van, és az meggátolja a lépési kísérletet. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lépés mérlegre – ajtó kinyílik |
| **Rövid leírás** | A játékos mérlegre lép, aminek következtében kinyílik a mérleghez tartozó ajtó |
| **Teszt célja** | Teszteli, hogy amikor a játékos rálép egy mérlegre, akkor megnövekszik a rajta lévő súly, átlépi a súlykorlátot, így hozzá tartozó ajtó kinyílik. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lépés mérlegre – ajtó zárva marad |
| **Rövid leírás** | A játékos mérlegre lép, de a mérleghez tartozó ajtó zárva marad. |
| **Teszt célja** | Teszteli, hogy amikor a játékos rálép egy mérlegre, akkor megnövekszik a rajta lévő súly, de nem lépi át a súlykorlátot, így hozzá tartozó ajtó tárva marad. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lépés mérlegről – ajtó bezárul |
| **Rövid leírás** | A játékos mérlegről lép egy másik koordinátára, aminek következtében bezárul a mérleghez tartozó ajtó |
| **Teszt célja** | Teszteli, hogy amikor a játékos lelép egy mérlegről, akkor a rajta lévő súly a súlykorlát alá csökken, így hozzá tartozó ajtó bezárul. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lépés mérlegről – ajtó bezárul |
| **Rövid leírás** | A játékos mérlegről lép egy másik koordinátára, aminek következtében bezárul a mérleghez tartozó ajtó |
| **Teszt célja** | Teszteli, hogy amikor a játékos lelép egy mérlegről, akkor a rajta lévő súly a súlykorlát alá csökken, így hozzá tartozó ajtó bezárul. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lépés féregjáratba |
| **Rövid leírás** | A játékos belép a féregjárat egyik oldalán lévő csillagkapujába és átkerül a másik csillagkapu elé. |
| **Teszt célja** | Teszteli, hogy amikor a játékos belép a féregjárat egyik kapuján, akkor átkerül a másik kapu elé. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Lépés árokba |
| **Rövid leírás** | A játékos olyan koordinátára lép, ahol árok van. |
| **Teszt célja** | tesztlei, hogy amikor a játékos árokba lép, akkor meghal, így a játék véget ér. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Replikátor lépése stepabble mezőre |
| **Rövid leírás** | A replikátor olyan koordinátára lép, ahova lehet. |
| **Teszt célja** | Teszteli, hogy a replikátor tud-e olyan koordinátára lépni, amire a játékos igen. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Replikátor lépése nem stepabble mezőre |
| **Rövid leírás** | A replikátror olyan koordinátára próbál lépni, amire nem lehet. |
| **Teszt célja** | Teszteli, hogy a replikátor nem tud olyan koordinátára lépni, amire a játékos sem. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Replikátor árokba lépése |
| **Rövid leírás** | A replikátror árokba lépve eltűnik és az árkot úttá alakítja. |
| **Teszt célja** | Teszteli, hogy amikor a replikátor mozgása során árokba lép, akkor eltűnik és az árok helyén létrejön egy sima út (az árok is eltűnik). |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Jaffa ZPM felvétel |
| **Rövid leírás** | Jaffa olyan koordinátára lép ahol ZPM van. |
| **Teszt célja** | Teszteli, hogy ha a Jaffa-t irányító játékos a karakterét olyan koordinátára lépteti, ahol ZPM található, akkor felveszi azt a pályáról, így azon eggyel kevesebb ZPM található. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | O’Neil páratlan sorszámú ZPM felvétele |
| **Rövid leírás** | O’Neil olyan koordinátára lép ahol ZPM van és ezzel együtt pártalan számú ZPM-et szedett össze. |
| **Teszt célja** | Teszteli, hogy ha az ezredest irányító játékos a karakterét olyan koordinátára lépteti, ahol ZPM található, akkor felveszi azt apályáról, ezzel a ZPM-mel együtt páratlan számú ZPM-t szedett fel, így nem keletkezik újabb ZPM a pályán. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | O’Neil páros sorszámú ZPM felvétele |
| **Rövid leírás** | O’Neil olyan koordinátára lép ahol ZPM van és ezzel együtt páros számú ZPM-et szedett össze. |
| **Teszt célja** | Teszteli, hogy ha az ezredest irányító játékos a karakterét olyan koordinátára lépteti, ahol ZPM található, akkor felveszi azt apályáról, ezzel a ZPM-mel együtt párosszámú ZPM-t szedett fel, így keletkezik újabb ZPM a pályán. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | Utolsó ZPM felvétele |
| **Rövid leírás** | Az egyik játékos felveszi az utolsó ZPM-et is a pályáról (nem keletkezik utána több). |
| **Teszt célja** | Teszteli, hogy ha az egyik játékos felveszi az utolsó ZPM-t akkor a játék véget ér. |

## Tesztelést támogató segéd- és fordítóprogramok specifikálása

A programunk működése determinisztikus, replikátor véletlenszerű mozgását kivéve, ezért a tesztelés a legtöbb esetben elvégezhető adott bemenetre előállt kimenet és az elvárt kimenet összehasonlításával. A replikátor mozgásánál a tesztelést adott bemenetre előállt kimenet és az elfogadható kimenetek listájának összehasonlításával végezzük.

Az összehasonlítások elvégzéséhez és a tesztesetek könnyű kezeléséhez az előbbiekben definiált ki- és bemeneti nyelvet használó felületet hozunk létre. A teszteset indításakor meg kell adnunk az elvárt kimenetet, vagy az elfogadható kimenetek listáját, melyet a teszteset végén összehasonlítunk. Egyezés vagy találatok esetén a tesztesetet hibamentesnek minősítjük.

A több elfogadható kimenettel rendelkező tesztesetek esetében, az elfogadható bemenetek számával arányosan több teszt futtatása szükséges. Ezt követően kiértékelést kapunk a tesztek hibás vagy hibamentes voltáról.

## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2016.03.30. 18:00 | 1 óra | Papp, Makki, Borbély, Gáspár, Zöld | Megbeszélés, feladatok kiosztása. |
| 2016.03.30. 19:50 | 20 perc | Zöld | Az interfész általános leírása |
| 2016.03.31. 12:15 | 1 óra | Makki | Tesztek |
| 2016.03.31. 16:15 | 30 perc | Gáspár | Tesztelést támogató segéd- és fordítóprogramok specifikálása |
| 2016.04.01. 16:00 | 2 óra | Borbély | Interfész leírás, dokumentáció véglegesítés. |
| 2016.04.03. 24:00 | 1 óra | Ritter | Dokumentáció javítása |

# 8. Részletes tervek

18 – don

Konzulens:

Ferencz Endre

Csapattagok

|  |  |  |
| --- | --- | --- |
| Borbély László Benedek | V19RPS | [laci.dj@gmail.com](mailto:laci.dj@gmail.com) |
| Gáspár Vilmos | GF4UZL | [gaspar.vilmos24@gmail.com](mailto:gaspar.vilmos24@gmail.com) |
| Makki Viktor | I0UA10 | [deldemonho@gmail.com](mailto:deldemonho@gmail.com) |
| Ritter Alex | GS9ERJ | [ritt.alex@gmail.com](mailto:ritt.alex@gmail.com) |
| Papp Albert | F9YTAY | [albertxpapp@gmail.com](mailto:albertxpapp@gmail.com) |
| Zöld Máté | GQH7FW | [mate951006@gmail.com](mailto:mate951006@gmail.com) |
|  |  |  |

2016. április 11.

# Részletes tervek

## Osztályok leírása

### Abyss

#### Felelősség

Szakadékot megvalósító osztály. Átjárható,rápakolható, ezáltal át lehet lőni felette. Nem jöhet létre rajta csillagkapu. Halálos.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok: Nincs

### Box

#### Felelősség

A dobozt megvalósító osztály. Mozgatható, rápakolható, nem átjárható, nem lehet átlőni felette, nem halálos és nem jöhet létre rajta csillagkapu, súlya van.

#### Ősosztályok: Nincs

#### Interfészek

* Movable

#### Attribútumok: Nincs

#### Metódusok:+ int GetWeight() : Vissza adja a doboz súlyát.

### BoxInventory

#### Felelősség

Tárolja a Box objektumokat.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-mov: HashMap<Coordinate, Box>**

#### Metódusok

* **+void Delete(Coordinate coord)**: Törli a paraméterként átvett koordinátán található dobozt.
* **+Box Get(Coordinate coord)**: Visszaadja a paraméterként átvett koordinátán található objektumot a HashMap-ből, ha nincs azon a koordinátán Box, akkor null-al tér vissza.
* **+void PutBox(Coordinate coord, Box carriedobject)**: A paraméterként átvett koordinátával elhelyezi a HashMap-ben a hordozott dobozt.
* **+Boolean IsThere(Coordinate coord)**: Visszaadja, hogy az adott helyen van-e doboz.

### Button

#### Felelősség

Egy mérleg. Feladata, hogy tudja, melyik ajtó tartozik hozzá, hogy annak meg tudja változtatni az állapotát, ha a rajta lévő súly átlép egy bizonyos limitet.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-wire: Door**
* **-data: DataAccessPoint**

#### Metódusok

* **+void Action()**: Megkérdi a rajta lévő dobozok illetve játékosok súlyát, és ha ez elér egy bizonyos értéket, akkor kinyitja a hozzá tartozó ajtót.

### ButtonInventory

#### Felelősség

Tárolja a pályán elhelyezkedő Button objektumokat.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-buttons: HashMap<Coordinate, Button>**

#### Metódusok

* **+void EventOn(CVector coord)**: Kezeli, ha erre a pozícióra objektum kerül.

### Charachter

#### Felelősség

Mozgásra képes objektumokat valósítja meg.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok: Nincs

#### Metódusok

* ***int GetWeight()****:* Visszaadja a karakter súlyát.
* ***void Step(toDir)****:* Elindítja a lépést.

### CollectableInventory

#### Felelősség

Tárolja a ZPM modulokat.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-col: HashMap<Coordinate, ZPM>**

#### Metódusok

* **+ZPM GetCollectableAt(Coordinate coord)**: Visszaadja a paraméteként átvett koordinátán található ZPM modult, minden második meghíváskor lerak egy újat a pályára véletlen szerűen, ha nincs ott semmi, akkor null-t ad vissza.
* **+Boolean IsThere(Coordinate coord)**: Visszaadja, hogy az adott helyen van-e ZPM.

### Coordinate

#### Felelősség

*Tárolja x, y koordinátákat.*

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-x**: int: x koordinátát tárolja.
* **-y**: int: y koordinátát tárolja.

#### Metódusok

* **+void Set(int sx, int sy)**: Beállítja erre a pozícióra koordinátákat.
* **+int GetX()**: Visszaadja x értékét.
* **+void GetY()**: Visszaadja y értékét.

### CVector

#### Felelősség

*Tárolja x, y koordinátákat, és az irányt ahova néz az adott objektum.*

#### Ősosztályok: Coordinate

#### Interfészek: Nincs

#### Attribútumok

* **-dir**: Direction: Az irányt tárolja.

#### Metódusok

* **+void SetDir(int sx, int sy, Direction sd)**: Beállítja erre a pozícióra koordinátákat.
* **+Direction GetDir()**: Visszaadja az irányt.

### DataAccessPoint

#### Felelősség

Hozzáférés biztosítása a StepHandler, a ShotHandler, és a GrabHandler számára a FieldObjectInventory-hoz, MovableInventory-hoz, ButtonInventory-hoz és a CollectableInventory-hoz.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **+field**: FieldObjectInventory: Referencia a FieldObject-ek tárolójára
* **+box**: BoxInventory: Referencia a mozgatható objektumok tárolójára.
* **+collectables**: CollectableInventory: Referencia gyűjthető objektumok tárolójára
* **+button**: ButtonInventory: Referencia a lenyomható gombok tárolójára.
* **+charachter**: Tartalmazza az entitásokat a pályán.

#### Metódusok: Nincs

### Door

#### Felelősség

Az ajtót megvalósító osztály. Felelőssége, hogy tudja, hogy milyen állapotban van. Zárt állapotban nem lehet átmenni rajta, így átlőni sem. Nyílt állapotban pedig át lehet menni rajta és így át is lehet lőni a nyitott ajtón. Semelyik állapotában nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok

* **-open: boolean**

#### Metódusok

* **+void** **Statechanged()**: Megváltoztatja az ajtó állapotát. Ha nyitva volt, akkor becsukja, ha zárva volt, akkor kinyitja.

### FieldObject

#### Felelősség

Interface, amely gondoskodik arról, hogy azokról az osztályokról, amik megvalósítják, el lehessen dönteni, hogy átjárhatók-e, át lehet-e lőni felettük, létre lehet-e hozni felette csillagkaput, illetve halálosak-e.

#### Ősosztályok: Nincs

#### Metódusok

* ***boolean IsMortal()****:* Visszaad egy logikai értéket, hogy az objektumra lépés halálos-e. Azaz ha a játékos rálép vagy doboz kerül rá, akkor az megsemmisül-e.
* ***boolean Shootable()****:* Visszaad egy logikai értéke, hogy az objektumon létrejöhet-e csillagkapu.
* ***boolean Steppable()****:* Visszaad egy logikai értéket, hogy az objektumra rá lehet-e lépni. Azokon amikre rá lehet lépni, át is lehet lőni.
* ***boolean Putable():*** Visszad egy logikai értéket, hogy az objektumra lehet-e rá rakni dobozt.

### FieldObjectInventory

#### Felelősség

FieldObject interface-et megvalósító objektumok tárolása.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-fieldobj: HashMap<Coordinate, FieldObject>**

#### Metódusok

* **+FieldObject GetFieldObject(Coordinate coord)**: A paraméter listán átvett koordinátán található FieldObject-et visszaadja a HashMap-ből, ha nincs a megodott koordinátán FieldObject, akkor null-t ad vissza.

### GrabHandler

#### Felelősség

A mozgatható objektumok mozgatásáról gondoskodik.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-data**: DataAcessPoint: Adat-összeköttetést biztosít.

#### Metódusok

* **-boolean CanPut(CVectro interactfrom):** Megnézi hogy lerakhass-e a dobozt az adott kordinátán.
* **+boolean Grab(CVector interactfrom)**: CVectorral adjuk meg, hogy honnan milyen irányból akarunk felvenni, true-t ad vissza, ha a felvétel sikerült.
* **+boolean Put(CVector interactfrom)**: CVectorral adjuk meg, hogy honnan milyen irányból akarunk letenni, true-t ad vissza, ha a letételsikerült.

### Player

#### Felelősség

Tárolja a Player (O’Neill ezredes) állapotát: pozícióját, irányát és hogy van-e valami a kezében. Ezeket változtatja a megfelelő függvényekkel.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-handfull**: Box: Van-e valami mozgatható a Player kezében.
* **-pos:** CVector: Megadja a Player pozícióját és irányát.
* **-leg**: StepHandler: Kezeli a Player lépéseit.
* **-gun**: ShootHandler: Kezeli a Player lövését.
* **-hand**: GrabHandler: Kezeli a mozgatható objektumok felszedését.

#### Metódusok

* **+void Interact(Direction toDir)**: A Player megpróbálja felvenni az előtte lévő objektumot.
* **+void Shoot(Direction toDir, StarGateColor color)**: A Player megpróbál egy adott színű csillagkaput lőni.
* **+void Step(Direction toDir)**: A Player megpróbál egy adott irányba lépni.

### Replicator

#### Felelősség

Járkál, ha árokba lép utat hoz létre az árok helyén.

#### Ősosztályok: Player

#### Interfészek: Nincs

#### Attribútumok

* **-data**: DataAcessPoint: Adat-összeköttetést biztosít.

#### Metódusok

* **+void Step(toDir)**: A lépést indítja el.

### ShootHandler

#### Felelősség

Lekezeli a játékos lövéseit.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-data**: DataAcessPoint: Adat-összeköttetést biztosít.

#### Metódusok

* **- boolen IsTransparent(C Vector shootfrom, Direction color):** a visszatérési értéke megadja hogy elakadt-e a lövés, ha replikárt talált el, akkor azt elpusztítja, ha képes az adott mezőn, akkor létrehozza a portál.
* **+void Shoot(CVector shootfrom, Direction color)**: CVectorral átadjuk a pozíciónkat, és hogy onnan milyen irányból akarunk lőni. Kiszámolja a becsapódás helyét, ha shootable, vezérléstovábbadással gondoskodik az esetleges csillagkapukról.

### SpecialWall

#### Felelősség

Speciális falat megvalósító osztály. Biztosítja, hogy nem átjárható, nem lehet átlőni felette. Nem halálos. Abban különbözik a Wall-tól, hogy létrejöhet rajta csillagkapu.

#### Ősosztályok: Wall

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok: Nincs

### StarGate

#### Felelősség

Maga a csillagkapu objektum, nyilvántartja a színét.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-color**: StarGateColor: Tárolja a csillagkapuk színét.
* **-pos**: CVector: Tárolja a csillagkapuk pozícióját.
  + - * **Metódusok**: Nincs

### StarGateInventory

#### Felelősség

Nyilvántartja a csillagkapukat.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok

* **-sgates**: StarGate: Tárolja a csillagkapukat.

#### Metódusok

* **+boolean IsThere(CVector from)**: Visszaadja, hogy az adott helyről érkezve van-e csillagkapu.
* **+CVector StepIn(CVector from)**: Adott helyről belép a csillagkapuba, visszaadja a következő pozíciót.
* **+void Create(CVector sgpos, StarGatColor color)**: Létrehoz adott pozíciójú, irányú és színű csillagkaput.
* **+void Delete(StarGatColor color)**: Törli az adott színű csillagkaput.

### StepHandler

#### Felelősség

Lekezeli a játékos lépéseit, információt szolgáltat a mezők járhatóságáról.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok:

* **-pushed**: Boolean: Nyilvántartj, hogy jelenleg Button-on áll-e a Player.
* **-data**: DataAcessPoint: Adat-összeköttetést biztosít.

#### Metódusok

* **+CVector Nextstep(CVector playerpos, Direction dir)**: Visszaadja a Player pozícióját, annak függvényében, hogy tudott-e lépni vagy sem, ha null pointert ad vissza akkor meghalt.

### Wall

#### Felelősség

A falat megvalósító osztály. Biztosítja, hogy nem lehet átmenni, átlőni rajta. Nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok: Nincs

### Way

#### Felelősség

Az utat megvalósító osztály. Átjárható, át lehet lőni felette, nem halálos és nem jöhet létre rajta csillagkapu.

#### Ősosztályok: Nincs

#### Interfészek

* FieldObject

#### Attribútumok: Nincs

#### Metódusok: Nincs

### ZPM

#### Felelősség

A játék során begyűjtendő ZPM-et megvalósító osztály.

#### Ősosztályok: Nincs

#### Interfészek: Nincs

#### Attribútumok: -count: int: A pályán található összes ZPM száma.

#### Metódusok

* **+void Collect()**:Begyűjtésekor hívódik meg.
* **+int HowMany()**: Megadja, hogy egy adott típusú Collectable-ből hány található a pályán.

## A tesztek részletes tervei, leírásuk a teszt nyelvén

A tesztesetek leírása során, amennyiben ezt külön nem említjük, a teszteset ugyanúgy működtethető lenne a *Jaffa* játékos objektummal, de nem hoztunk létre külön teszteseteket, mivel a *Colonel*-lel (O’Neil ezredes) ugyanazon osztály példányai, ez a teszteset sikerességét sok esetben nem befolyásolná.

A *Colonel* minden tesztesetben, amennyiben mást nem kötünk ki, kezdetben a (1;1) koordinátákon foglal helyet és északi (North) irányba néz.

### Doboz felvétel dobozzal rendelkező mezőről

* **Leírás**

Teszteli, hogy játékos fel tud venni egy dobozt, ha a megfelelő mezőn tartózkodik (közvetlenül a doboz előtt áll).

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a doboz felvétele megtörténik-e és a felvételi irányban már nem maradt doboz a pályán. Hibalehetőség például, hogy doboz kerül a játékos *handfull* tárolójába, de a felvétel helyén is marad egy doboz (duplikáció történik). Előfordulhat az is, hogy a dobozfelvétel nem történik meg, a játékos *handfull* tárolója üres marad.

* **Bemenet**

Colonel Step East

Colonel Interact

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel has grabbed an item at 2,1

### Doboz felvétel doboz nélküli mezőről

* **Leírás**

Teszteli, hogy játékos fel tud venni egy dobozt, ha a nem megfelelő mezőn tartózkodik (nincs közvetlenül előtte doboz).

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a doboz felvétele megtörténik-e. Hibalehetőség hogy megtörténik a felvétel, a játékos *handfull* tárolójába doboz kerül, de eközben nincs doboz az inteakciós irányban.

* **Bemenet**

Colonel Interact

* **Elvárt kimenet**

Colonel can’tinteract at: 1, 2

### Doboz lerakás, ha a mezőn van már doboz

* **Leírás**

Teszteli, hogy játékos le tud rakni egy dobozt, ha a már a mezőn is van egy, ha a lerakáshoz megfelelő mezőn tartózkodik. A tesztesetben a (2;2) koordinátán vana másik doboz.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a lerakás megfelelően végbement. Hibalehetőség, hogy nem lehetne dobozra dobozt rakni.

* **Bemenet**

Colonel Step East

Colonel Interact

Colonel Step East

Colonel Step North

Colonel Interact

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel has grabbed an item at 2,1

Colonel new position: 2,1

Colonel new position: 2,1

Colonel has put an item at 2,2

### Doboz felvétel mérlegről – ajtózárás

* **Leírás**

Teszteli, hogy mérlegről lekerülő doboz miatt a mérleghez tartozó ajtó bezárul-e. (A mérlegen lévő súly a súlykorlát alá kerül). A tesztesetben a (2;1) koordinátán van a mérleg, amin éppen annyi doboz van, amennyi még magasabb a súlykorlátnál.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a mérleghez tartozó ajtó bezárul-e. Hibalehetőség: az ajtó nyitva marad.

* **Bemenet**

Colonel Step East

Colonel Interact

* **Elvárt kimenet**

door opened Colonel new position: 1,1

Colonel has grabbed an item at 2,1 door closed

### Doboz felvétel mérlegről – ajtó nyitva marad

* **Leírás**

Teszteli, hogy mérlegről lekerülő doboz miatt a mérleghez tartozó ajtó bezárul-e. (A mérlegen lévő súly a súlykorlát felett marad). A tesztesetben a (2;1) koordinátán van a mérleg, amin eggyel több doboz van, mint amennyi szükséges a súlykorlát túllépéséhez.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a mérleghez tartozó ajtó bezárul-e. Hibalehetőség: az ajtó bezárul.

* **Bemenet**

Colonel Step East

Colonel Interact

* **Elvárt kimenet**

door opened Colonel new position: 1,1

Colonel has grabbed an item at 2,1

### Doboz felvétel féregjáraton keresztül

* **Leírás**

Teszteli, hogy a játékos féregjáraton keresztül fel tud venni egy dobozt. A tesztesetben egy (0;1) koordinátájú csillagkaput és egy (3;1) koordinátájú csillagkaput hozunk létre, az (2;1) koordinátán doboz található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a dobozfelvétel lehetséges-e csillagkapuk használatával, féregjáraton keresztül. Hibalehetőség például, hogy doboz kerül a játékos *handfull* tárolójába, de a felvétel helyén is marad egy doboz (duplikáció történik). Előfordulhat az is, hogy a dobozfelvétel nem történik meg, a játékos *handfull* tárolója üres marad.

* **Bemenet**

Colonel Step West

Colonel Shoot Blue

Colonel Step East

Colonel Interact

Colonel Step North

Colonel Interact

Colonel Step East

Colonel Shoot Yellow

Colonel Step North

Colonel Interact

Colonel Step West

Colonel Interact

Colonel Interact

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel Blue portal at 0,1

Colonel new position: 1,1

Colonel has grabbed an item at 2,1

Colonel new position: 1,1

Colonel has put an item at 1,2

Colonel new position: 1,1

Colonel Yellow portal at 3,1

Colonel new position: 1,1

Colonel has grabbed an item at 1,2

Colonel new position: 1,1

Colonel has put an item at 2,1

Colonel has grabbed an item at 2,1

### Doboz lerakás szabad mezőre

* **Leírás**

Teszteli, hogy játékos le tud-e rakni egy dobozt, ha egy szabad mező irányába néz.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a doboz lerakása megtörténik-e. Hibalehetőség például, hogy a doboz kikerül a játékos *handfull* tárolójából, de a lerakási helyen nem lesz doboz. Előfordulhat az is, hogy a lerakás nem történik meg, a játékos *handfull* tárolójában marad.

* **Bemenet**

Colonel Step East

Colonel Interact

Colonel Interact

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel has grabbed an item at 2,1

Colonel has put an item at 2,1

### Doboz lerakás nem szabad mezőre

* **Leírás**

Teszteli, hogy játékos le tud-e rakni egy dobozt, ha egy nem szabad mező irányába néz. A tesztesetben a (0;1) koordinátán ZPM modul található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a doboz lerakása megtörténik-e. Hibalehetőség hogy megtörténik a lerakása, a játékos *handfull* tárolója kiürül.

* **Bemenet**

Colonel Step East

Colonel Interact

Colonel Step North

Colonel Interact

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel has grabbed an item at 2,1

Colonel new position: 1,1

Colonel can't interact at 1,2

### Doboz lerakás, a játékos kezében nincs doboz

* **Leírás**

Teszteli, hogy játékos le tud-e tenni egy dobozt, ha a kezében nincs.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a doboz lerakás megtörtént-e. Hibalehetőség, hogy létrejön egy doboz az interakció irányában.

* **Bemenet**

Colonel Interact

* **Elvárt kimenet**

Colonel can’t interact at: 1,2

### Doboz lerakás mérlegre – ajtónyitás

* **Leírás**

Teszteli, hogy mérlegre kerülő doboz miatt a mérleghez tartozó ajtó kinyílik-e. (A mérlegen lévő súly a súlykorlátot eléri). A tesztesetben a (2;1) koordinátán van a mérleg, amin éppen annyi doboz van, amennyi még alacsonyabb a súlykorlátnál.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a mérleghez tartozó ajtó kinyílik-e. Hibalehetőség: az ajtó zárva marad.

* **Bemenet**

Colonel Interact

Colonel Step East

Colonel Interact

* **Elvárt kimenet**

Colonel has grabbed an item at 1,2

Colonel new position: 1,1

Colonel has put an item at 2,1 door opened

### Doboz lerakás mérlegre – ajtó zárva marad

* **Leírás**

Teszteli, hogy mérlegre kerülő doboz miatt a mérleghez tartozó ajtó kinyílik-e. (A mérlegen lévő súly a súlykorlát alatt marad). A tesztesetben a (2;1) koordinátán van a mérleg, amin kettővel kevesebb doboz van, mint amennyi szükséges a súlykorlát túllépéséhez.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a mérleghez tartozó ajtó kinyílik-e. Hibalehetőség: az ajtó kinyílik.

* **Bemenet**

Colonel Interact

Colonel Step East

Colonel Interact

* **Elvárt kimenet**

Colonel has grabbed an item at 1,2

Colonel new position: 1,1

Colonel has put an item at 2,1

### Doboz lerakás féregjáraton keresztül

* **Leírás**

Teszteli, hogy a játékos féregjáraton keresztül le tud-e tenni egy dobozt. A tesztesetben egy (0;1) koordinátájú csillagkapu és egy (3;1) koordinátájú csillagkapu lesz létrehozva, az (5;5) koordináta szabad (nincs rajt olyan elem, ami a dobozlerakást akadályozná).

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a dobozlerakás lehetséges-e csillagkapuk használatával, féregjáraton keresztül. Hibalehetőség például, hogy doboz kikerül a játékos *handfull* tárolójából, de a lerakás helyére nem kerül doboz. Előfordulhat az is, hogy a dobozlerakás nem történik meg, a játékos *handfull* tárolójában marad.

* **Bemenet**

Colonel Step West

Colonel Shoot Blue

Colonel Step East

Colonel Interact

Colonel Step North

Colonel Interact

Colonel Step East

Colonel Shoot Yellow

Colonel Step North

Colonel Interact

Colonel Step West

Colonel Interact

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel Blue portal at 0,1

Colonel new position: 1,1

Colonel has grabbed an item at 2,1

Colonel new position: 1,1

Colonel has put an item at 1,2

Colonel new position: 1,1

Colonel Yellow portal at 3,1

Colonel new position: 1,1

Colonel has grabbed an item at 1,2

Colonel new position: 1,1

Colonel has put an item at 2,1

### Lövés falra / zárt ajtóra

* **Leírás**

Teszteli, hogy amikor a játékos lő és a lövedék falat vagy zárt ajtót talál el, azon nem hoz létre csillagkaput. A tesztesetben a (0;1) pozícióban fal található, valamint (1;2) pozícióban egy ajtó található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy létrejött-e csillagkapu a találat helyén. Hibalehetőség, hogy a nem alkalmas felületen csillagkapu jön létre.

* **Bemenet**

Colonel Shoot Blue

Colonel Step West

Colonel Shoot Blue

* **Elvárt kimenet**

Colonel failed to create portal at 1,2

Colonel new position: 1,1

Colonel failed to create portal at 0,1

### Lövés speciális falra

* **Leírás**

Teszteli, hogy amikor a játékos lő és a lövése speciális falt talál el, létrejön-e csillagkapu. A lövéssel azonos színű csillagkapunak be kell záródnia. A tesztesetben a (0;1) és (1;2) pozícióban speciális fal található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy hogy létrejött-e csillagkapu a találat helyén, valamint hogy fentmaradt-e az előző azonos színű csillagkapu. Hibalehetőség, hogy az alkalmas felületen nem jön létre csillagkapu, vagy az előző pozícióban nem semmisül meg az adott színű.

* **Bemenet**

Colonel Shoot Blue

Colonel Step West

Colonel Shoot Blue

* **Elvárt kimenet**

Colonel Blue portal at 1,2

Colonel new position: 1,1

Colonel Blue portal at 0,1

### Lövés dobozra

* **Leírás**

Teszteli, hogy amikor a játékos lő és a lövése dobozt talál el, azon nem hoz létre csillagkaput. A tesztesetben a (2;1) pozícióban doboz található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy létrejött-e csillagkapu. Hibalehetőség: létrejön a nem alkalmas felületen.

* **Bemenet**

Colonel Step East

Colonel Shoot Blue

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel failed to create portal at 2,1

### Lövés replikátorra

* **Leírás**

Teszteli, hogy amikor a játékos lő és a lövése replikátort talál el, akkor a replikátor elpusztul-e. A tesztesetben a (1;3) pozícióban replikátor található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy az eltalált replikátor elpusztul-e. Hibalehetőség: A replikátor nem pusztul el, esetleg csillagkapu jön létre rajta.

* **Bemenet**

Colonel Shoot Blue

* **Elvárt kimenet**

Colonel Replicator has been shot

failed to create portal at 1,3

### Lépés útra/nyitott ajtóba

* **Leírás**

Teszteli, hogy amikor a játékos útra vagy nyitott ajtóba lép, akkor nem történik semmi a játékos koordinátájának megváltozásán kívül. Kezdetben a játékos az (1;1) koordinátán áll, majd a (1;2) koordinátán található ajtóra lép.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékos pozíciója megváltozott-e. Hibalehetőség: Az előbbi működés nem valósul meg.

* **Bemenet**

Colonel Step North

* **Elvárt kimenet**

door opened Colonel new position: 1,2

### Lépés falra

* **Leírás**

Teszteli, hogy játékos nem tud a kívánt koordinátára lépni, mert a fenti objektumok egyik ott van, és az meggátolja a lépési kísérletet. A játékos az (1;1) koordinátán áll, tőle közvetlenül délre (balra), a mozgást megakadályozó objektum.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékost megállítja-e az útjába került objektum.

* **Bemenet**

Colonel Step South

Colonel Step South

* **Elvárt kimenet**

door opened Colonel new position: 1,1

Colonel new position: 1,1

### Lépés spec falra

* **Leírás**

Teszteli, hogy játékos nem tud a kívánt koordinátára lépni, mert a fenti objektumok egyik ott van, és az meggátolja a lépési kísérletet. A játékos az (1;1) koordinátán áll, tőle közvetlenül délre (felfelé), a mozgást megakadályozó objektum.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékost megállítja-e az útjába került objektum.

* **Bemenet**

Colonel Step West

Colonel Step West

* **Elvárt kimenet**

door opened Colonel new position: 1,1

Colonel new position: 1,1

### Lépés zárt ajtóra

* **Leírás**

Teszteli, hogy játékos nem tud a kívánt koordinátára lépni, mert a fenti objektumok egyik ott van, és az meggátolja a lépési kísérletet. A játékos az (1;1) koordinátán áll, tőle közvetlenül északra (lefelé), a mozgást megakadályozó objektum.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékost megállítja-e az útjába került objektum.

* **Bemenet**

Colonel Step North

* **Elvárt kimenet**

Colonel new position: 1,1

### Lépés dobozra

* **Leírás**

Teszteli, hogy játékos nem tud a kívánt koordinátára lépni, mert a fenti objektumok egyik ott van, és az meggátolja a lépési kísérletet. A játékos az (1;1) koordinátán áll, tőle közvetlenül keletre (jobbra), a mozgást megakadályozó objektum.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékost megállítja-e az útjába került objektum.

* **Bemenet**

Colonel Step East

Colonel Step East

* **Elvárt kimenet**

door opened Colonel new position: 1,1

Colonel new position: 1,1

### Lépés másik játékosra

* **Leírás**

Teszteli, hogy játékos nem tud a kívánt koordinátára lépni, mert a fenti objektumok egyik ott van, és az meggátolja a lépési kísérletet. A játékos az (1;1) koordinátán áll, tőle közvetlenül keletre (jobbra), a mozgást megakadályozó objektum.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékost megállítja-e az útjába került objektum.

* **Bemenet**

Colonel Step East

Colonel Step East

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel new position: 1,1

### Lépés mérlegre – ajtó kinyílik

* **Leírás**

Teszteli, hogy amikor a játékos rálép egy mérlegre, akkor megnövekszik a rajta lévő súly, átlépi a súlykorlátot, így hozzá tartozó ajtó kinyílik. A játékos az (1;1) koordinátán áll, tőle közvetlenül keletre (jobbra) egy mérleg található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékos hatására az ajtó kinyílik-e. Hibalehetőség: Az ajtó zárva marad.

* **Bemenet**

ColonelStep East

ColonelStep East

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel new position: 2,1 door opened

### Lépés mérlegre – ajtó zárva marad

* **Leírás**

Teszteli, hogy amikor a játékos rálép egy mérlegre, akkor megnövekszik a rajta lévő súly, de nem lépi át a súlykorlátot, így hozzá tartozó ajtó zárva marad. A játékos az (1;1) koordinátán áll, tőle közvetlenül keletre (jobbra) egy mérleg található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékos hatására az ajtó zárva marad-e. Hibalehetőség: Az ajtó kinyílik.

* **Bemenet**

ColonelStep East

ColonelStep East

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel new position: 2,1

### Lépés mérlegről – ajtó bezárul

* **Leírás**

Teszteli, hogy amikor a játékos lelép egy mérlegről, akkor a rajta lévő súly a súlykorlát alá csökken, így hozzá tartozó ajtó bezárul. A játékos az (1;1) koordinátán áll, alatta egy mérleg található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékos hatására az ajtó bezárul-e. Hibalehetőség: Az ajtó nyitva marad.

* **Bemenet**

ColonelStep East

ColonelStep East

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel new position: 2,1 door opened

Colonel new position: 2,1

Colonel new position: 1,1 door closed

### Lépés féregjáratba

* **Leírás**

Teszteli, hogy amikor a játékos belép a féregjárat egyik kapuján, akkor átkerül a másik kapu elé. A játékos az (1;1) koordinátán áll. Tőle közvetlenül nyugatra (balra) egy csillagkaput lő, meg a (4;1) koordinátára is.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékos a lépés hatására átkerül-e a kívánt pozícióba. Hibalehetőség: A játékos a csillagkapun marad.

* **Bemenet**

Colonel Step West

Colonel Shoot Blue

Colonel Step East

Colonel Shoot Yellow

Colonel Step West

Colonel Step West

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel Blue portal at 0,1

Colonel new position: 1,1

Colonel Yellow portal at 4,1

Colonel new position: 1,1

Colonel new position: 3,1

### Lépés árokba

* **Leírás**

Teszteli, hogy amikor a játékos árokba lép, akkor meghal, így a játék véget ér. A játékos az (1;1) koordinátán áll, tőle közvetlenül keletre (jobbra) egy árok található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékos a lépés hatására átkerül-e a kívánt pozícióba. Hibalehetőség: A játékos a csillagkapun marad.

* **Bemenet**

ColonelStep East

ColonelStep East

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel new position: 2,1 Game Over

### Replikátor lépése stepabble mezőre

* **Leírás**

Teszteli, hogy a replikátor tud-e olyan koordinátára lépni, amire a játékos igen. A replikátor az (2;1) koordinátán áll, tőle közvetlenül keletre (jobbra) szabad út található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a replikátor át tud-e lépni a (3;1) koordinátájú pontra. Hibalehetőség: Nem tud átlépni.

* **Bemenet**

Replicator Step East

Replicator Step East

* **Elvárt kimenet**

Replicator new position: 2,1

Replicator new position: 3,1

### Replikátor lépése nem stepabble mezőre

* **Leírás**

Teszteli, hogy a replikátor nem tud olyan koordinátára lépni, amire a játékos sem. A replikátor az (2;1) koordinátán áll, tőle közvetlenül keletre (jobbra) fal található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a replikátor át tud-e lépni a (3;1) koordinátájú pontra. Hibalehetőség: Át tud.

* **Bemenet**

Replicator Step East

Replicator Step East

* **Elvárt kimenet**

Replicator new position: 2,1

Replicator new position: 2,1

### Replikátor árokba lépése

* **Leírás**

Teszteli, hogy amikor a replikátor mozgása során árokba lép, akkor eltűnik és az árok helyén létrejön egy sima út (az árok is eltűnik). A replikátor az (1;0) koordinátán áll, tőle közvetlenül keletre (jobbra) árok található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékos a lépés hatására átkerül-e a kívánt pozícióba és létrejön-e az út. Hibalehetőség: A replikátor nem hoz létre utat.

* **Bemenet**

Replicator Step East

Replicator Step East

* **Elvárt kimenet**

Replicator new position: 2,1

Replicator new position: 3,1 way spawned

### Jaffa ZPM felvétel

* **Leírás**

Teszteli, hogy ha a Jaffa-t irányító játékos a karakterét olyan koordinátára lépteti, ahol ZPM található, akkor felveszi azt a pályáról, így azon eggyel kevesebb ZPM található. Jaffa az (2;1) koordinátán áll, tőle közvetlenül keletre (jobbra) egy ZPM található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy Jaffa a lépés hatására átkerül-e a kívánt pozícióba és felveszi a ZPM-et. Hibalehetőség: Jaffa nem veszi fel a ZPM-et.

* **Bemenet**

JaffaStep East

JaffaStep East

* **Elvárt kimenet**

Jaffa new position: 2,1

Jaffa new position: 3,1 ZPM collected

### O’Neil páratlan sorszámú ZPM felvétele

* **Leírás**

Teszteli, hogy ha az ezredest irányító játékos a karakterét olyan koordinátára lépteti, ahol ZPM található, akkor felveszi azt apályáról, ezzel a ZPM-mel együtt páratlan számú ZPM-t szedett fel, így nem keletkezik újabb ZPM a pályán. Az ezredes az (1;1) koordinátán áll, tőle közvetlenül keletre (jobbra) egy ZPM található.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékos a lépés hatására átkerül-e a kívánt pozícióba és felveszi a ZPM-et. Hibalehetőség: A játékos nem veszi fel a ZPM-et.

* **Bemenet**

ColonelStep East

ColonelStep East

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel new position: 2,1 ZPM collected

### O’Neil páros sorszámú ZPM felvétele

* **Leírás**

Teszteli, hogy ha az ezredest irányító játékos a karakterét olyan koordinátára lépteti, ahol ZPM található, akkor felveszi azt a pályáról, ezzel a ZPM-mel együtt páros számú ZPM-t szedett fel, így keletkezik újabb ZPM a pályán.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a játékos a lépés hatására átkerül-e a kívánt pozícióba, felveszi a ZPM-et majd egy új kerül a pályára. Hibalehetőség: Nem generálódik új ZPM a pályán.

* **Bemenet**

ColonelStep East

Colonel Step East

Colonel Step East

* **Elvárt kimenet**

Colonel new position: 1,1

Colonel new position: 2,1 ZPM collected

Colonel new position: 3,1 ZPM collected ZPM spawned

## A tesztelést támogató programok tervei

A teszteléshez egy egyénileg elkészített alkalmazással fogjuk generálni a különböző, az előző dokumentumban definiált pályaleíró nyelvnek megfelelő bemeneti állományokat. Majd azokat a megfelelő sorrendben, a feljebb definiált bemeneti utasításokkal tesztelni.

A tesztesetek diverzifikálására az alkalmazás, azokat egymás után futtatja, majd összehasonlítja a kapott kimenetet az elvárt kimenettel és az egyezést (vagy az elvárttól való esetleges eltérést) a tesztelést végző személy felé jelzi. A tesztelést végző személynek lehetőséget biztosítunk az egyes tesztesetek különálló futtatására is.

A tesztprogram használata egyébként nem létfontosságú, hiszen a bemeneti állományok jegyzettömbbel is szerkeszthetők és értelmezhetők, az alkalmazás csak ezt a folyamatot hivatott meggyorsítani.

## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2016.04.07. 12:00 | 2,5 óra | Zöld | Tesztesetek kibővítése (múlt heti) |
| 2016.04.09. 17:00 | 3 óra | Gáspár | Lövéssel és interakcióval kapcsolatos tesztesetek kifejtése |
| 2016. 04. 10. 10:00 | 2 óra | Borbély | Tesztesetek kifejtése, visszamenőleges pontosítás végzése, tesztprogram terve. |
| 2016. 04. 10. 18:00 | 2,5 óra | Makki, Papp | Osztályleírások |

# 10. Prototípus beadása

18 – don

Konzulens:

Ferencz Endre

Csapattagok

|  |  |  |
| --- | --- | --- |
| Borbély László Benedek | V19RPS | [laci.dj@gmail.com](mailto:laci.dj@gmail.com) |
| Gáspár Vilmos | GF4UZL | [gaspar.vilmos24@gmail.com](mailto:gaspar.vilmos24@gmail.com) |
| Makki Viktor | I0UA10 | [deldemonho@gmail.com](mailto:deldemonho@gmail.com) |
| Ritter Alex | GS9ERJ | [ritt.alex@gmail.com](mailto:ritt.alex@gmail.com) |
| Papp Albert | F9YTAY | [albertxpapp@gmail.com](mailto:albertxpapp@gmail.com) |
| Zöld Máté | GQH7FW | [mate951006@gmail.com](mailto:mate951006@gmail.com) |
|  |  |  |

2016. április. 25.

# Prototípus beadása

## Fordítási és futtatási útmutató

### Fájllista

|  |  |  |  |
| --- | --- | --- | --- |
| **Fájl neve** | **Méret** | **Keletkezés ideje** | **Tartalom** |
| Abyss.java | 1 KB | 2016.04.20 | Szakadék tulajdonságai |
| Box.java | 1 KB | 2016.04.20 | Doboz tulajdonságai |
| BoxInventory.java | 4 KB | 2016.04.20 | Dobozok tárolása |
| Button.java | 1 KB | 2016.04.20 | Nyomólapok tulajdonságai |
| ButtonInventory.java | 2 KB | 2016.04.20 | Nyomólapok tárolása |
| Collectable.java | 1 KB | 2016.04.20 | Felszedhető objektumok tulajdonságai |
| CollectableInventory.java | 3 KB | 2016.04.20 | Felszedhető objektumok tárolása |
| Colonal.java | 2 KB | 2016.04.20 | Az ezredes |
| Coordinate.java | 2 KB | 2016.04.20 | Koordináták tárolása |
| CustomHashMap.java | 2 KB | 2016.04.20 | Saját HashMap megvalósítása |
| Cvector.java | 2 KB | 2016.04.20 | Koordináták iránnyal tárolása |
| DataAcessPoint.java | 2 KB | 2016.04.20 | Adat összekötetést biztosít |
| Direction.java | 1 KB | 2016.04.20 | Irány enum |
| Door.java | 1 KB | 2016.04.20 | Ajtó tulajdonságai |
| FieldObject.java | 1 KB | 2016.04.20 | FieldObject tulajdonságai |
| FieldObjectInventory.java | 2 KB | 2016.04.20 | FieldObject tárolása |
| GameClass.java | 11 KB | 2016.04.20 | Menu (main) |
| GrabHandler.java | 3 KB | 2016.04.20 | Objektum felvétének ill. lerakásának esetei lekezelése |
| Jaffa.java | 2 KB | 2016.04.20 | A második játékos |
| MANIFEST.MF | 1 KB | 2016.04.20 | Main file megadása |
| Player.java | 3 KB | 2016.04.20 | Játékos interakciói |
| Replicator.java | 3 KB | 2016.04.20 | Eldöntető kérdésre form |
| ShotHandler.java | 2 KB | 2016.04.20 | Lövés eseteinek lekezelése |
| SpecialWall.java | 1 KB | 2016.04.20 | Speciális fal tulajdonságai |
| Stargate.java | 1 KB | 2016.04.20 | Csillagkapu tulajdonságai |
| StargateColor.java | 1 KB | 2016.04.20 | Csillagkapuszíne enum |
| StargateInventory.java | 4 KB | 2016.04.20 | Csillagkapu tárolása |
| StepHandler.java | 3 KB | 2016.04.20 | Lépés eseteinek lekezelése |
| test1in.txt | 1 KB | 2016.04.24 | 1 Test bemeneti parancsai |
| test2in.txt | 1 KB | 2016.04.24 | 2 Test bemeneti parancsai |
| test3in.txt | 1 KB | 2016.04.24 | 3 Test bemeneti parancsai |
| test4in.txt | 1 KB | 2016.04.24 | 4 Test bemeneti parancsai |
| test5in.txt | 1 KB | 2016.04.24 | 5 Test bemeneti parancsai |
| test6in.txt | 1 KB | 2016.04.24 | 6 Test bemeneti parancsai |
| test7in.txt | 1 KB | 2016.04.24 | 7 Test bemeneti parancsai |
| test8in.txt | 1 KB | 2016.04.24 | 8 Test bemeneti parancsai |
| test9in.txt | 1 KB | 2016.04.24 | 9 Test bemeneti parancsai |
| test10in.txt | 1 KB | 2016.04.24 | 10 Test bemeneti parancsai |
| test11in.txt | 1 KB | 2016.04.24 | 11 Test bemeneti parancsai |
| test12in.txt | 1 KB | 2016.04.24 | 12 Test bemeneti parancsai |
| test13in.txt | 1 KB | 2016.04.24 | 13 Test bemeneti parancsai |
| test14in.txt | 1 KB | 2016.04.24 | 14 Test bemeneti parancsai |
| test15in.txt | 1 KB | 2016.04.24 | 15 Test bemeneti parancsai |
| test16in.txt | 1 KB | 2016.04.24 | 16 Test bemeneti parancsai |
| test17in.txt | 1 KB | 2016.04.24 | 17 Test bemeneti parancsai |
| test18in.txt | 1 KB | 2016.04.24 | 18 Test bemeneti parancsai |
| test19in.txt | 1 KB | 2016.04.24 | 19 Test bemeneti parancsai |
| test20in.txt | 1 KB | 2016.04.24 | 20 Test bemeneti parancsai |
| test21in.txt | 1 KB | 2016.04.24 | 21 Test bemeneti parancsai |
| test22in.txt | 1 KB | 2016.04.24 | 22 Test bemeneti parancsai |
| test23in.txt | 1 KB | 2016.04.24 | 23 Test bemeneti parancsai |
| test24in.txt | 1 KB | 2016.04.24 | 24 Test bemeneti parancsai |
| test25in.txt | 1 KB | 2016.04.24 | 25 Test bemeneti parancsai |
| test26in.txt | 1 KB | 2016.04.24 | 26 Test bemeneti parancsai |
| test27in.txt | 1 KB | 2016.04.24 | 27 Test bemeneti parancsai |
| test28in.txt | 1 KB | 2016.04.24 | 28 Test bemeneti parancsai |
| test29in.txt | 1 KB | 2016.04.24 | 29 Test bemeneti parancsai |
| test30in.txt | 1 KB | 2016.04.24 | 30 Test bemeneti parancsai |
| test31in.txt | 1 KB | 2016.04.24 | 31 Test bemeneti parancsai |
| test32in.txt | 1 KB | 2016.04.24 | 32 Test bemeneti parancsai |
| test33in.txt | 1 KB | 2016.04.24 | 33 Test bemeneti parancsai |
| TestHandler.java | 8 KB | 2016.04.20 | Tesztelés |
| testmap1.txt | 1 KB | 2016.04.24 | 1 Test map |
| testmap2.txt | 1 KB | 2016.04.24 | 2 Test map |
| testmap3.txt | 1 KB | 2016.04.24 | 3 Test map |
| testmap4.txt | 1 KB | 2016.04.24 | 4 Test map |
| testmap5.txt | 1 KB | 2016.04.24 | 5 Test map |
| testmap6.txt | 1 KB | 2016.04.24 | 6 Test map |
| testmap7.txt | 1 KB | 2016.04.24 | 7 Test map |
| testmap8.txt | 1 KB | 2016.04.24 | 8 Test map |
| testmap9.txt | 1 KB | 2016.04.24 | 9 Test map |
| testmap10.txt | 1 KB | 2016.04.24 | 10 Test map |
| testmap11.txt | 1 KB | 2016.04.24 | 11 Test map |
| testmap12.txt | 1 KB | 2016.04.24 | 12 Test map |
| testmap13.txt | 1 KB | 2016.04.24 | 13 Test map |
| testmap14.txt | 1 KB | 2016.04.24 | 14 Test map |
| testmap15.txt | 1 KB | 2016.04.24 | 15 Test map |
| testmap16.txt | 1 KB | 2016.04.24 | 16 Test map |
| testmap17.txt | 1 KB | 2016.04.24 | 17 Test map |
| testmap18.txt | 1 KB | 2016.04.24 | 18 Test map |
| testmap19.txt | 1 KB | 2016.04.24 | 19 Test map |
| testmap20.txt | 1 KB | 2016.04.24 | 20 Test map |
| testmap21.txt | 1 KB | 2016.04.24 | 21 Test map |
| testmap22.txt | 1 KB | 2016.04.24 | 22 Test map |
| testmap23.txt | 1 KB | 2016.04.24 | 23 Test map |
| testmap24.txt | 1 KB | 2016.04.24 | 24 Test map |
| testmap25.txt | 1 KB | 2016.04.24 | 25 Test map |
| testmap26.txt | 1 KB | 2016.04.24 | 26 Test map |
| testmap27.txt | 1 KB | 2016.04.24 | 27 Test map |
| testmap28.txt | 1 KB | 2016.04.24 | 28 Test map |
| testmap29.txt | 1 KB | 2016.04.24 | 29 Test map |
| testmap30.txt | 1 KB | 2016.04.24 | 30 Test map |
| testmap31.txt | 1 KB | 2016.04.24 | 31 Test map |
| testmap32.txt | 1 KB | 2016.04.24 | 32 Test map |
| testmap33.txt | 1 KB | 2016.04.24 | 33 Testmap map |
| Wall.java | 1 KB | 2016.04.20 | Fal tulajdonságai |
| Way.java | 1 KB | 2016.04.20 | Út tulajdonságai |
| ZPM.java | 1 KB | 2016.04.20 | ZPM tulajdonságai |

### Fordítás

Az előre legenerált manifest fájl és a forrásfájlok segítségével parancssorból a következő parancsokat kell kiadni:

> javac GameObjects/\*.java Handlers/\*.java Inventories/\*.java Players/\*.java Tools/\*.java

> jar cfm Game.jar MANIFEST.MF GameObjects/\*.class Handlers/\*.class Inventories/\*.class Players/\*.class Tools/\*.class

Így a forrásfájlokból egyetlenegy futtatható jar fájl készül.

### Futtatás

|  |  |  |
| --- | --- | --- |
| **Azonosító** | **Leírás** | **Komment** |
| E1. | Java Runtime Environment 1.8 | Mivel a szoftvernek a HSZK gépein futtathatónak kell lenniük, ezért az említettnél frissebb és régebbi verzió használata nem ajánlott. |
| E2. | Billentyűzet | A prototípus futása csak billentyűzetről vezérelhető. |

## A futtatáshoz a fordítási utasításokon túl még a következő parancsokat kell kiadni:

>java –jar Game.jar

Ezután a bemeneten egy szám megadásával [1-33] futtathatók a tesztesetek.

## Tesztek jegyzőkönyvei

### Doboz felvétel dobozzal rendelkező mezőről

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz felvétel doboz nélküli mezőről

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz lerakás, ha a mezőn van már doboz

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz felvétel mérlegről – ajtózárás

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz felvétel mérlegről – ajtó nyitva marad

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz felvétel féregjáraton keresztül

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz lerakás szabad mezőre

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz lerakás nem szabad mezőre

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz lerakás, a játékos kezében nincs doboz

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz lerakás mérlegre – ajtónyitás

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz lerakás mérlegre – ajtó zárva marad

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Doboz lerakás féregjáraton keresztül

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lövés falra / zárt ajtóra

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lövés speciális falra

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lövés dobozra

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lövés replikátorra

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lépés útra/nyitott ajtóba

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lépés falra

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |
| **Teszt eredménye** | tudtunk a falra lépni |
| **Lehetséges hibaok** | nem megfelelően működik a Wall osztály |
| **Változtatások** | a Steppable visszatérését false módosítottam |

### Lépés spec falra

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lépés zárt ajtóra

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lépés dobozra

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lépés másik játékosra

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |
| **Teszt eredménye** | két játékos lehetett egy mezőn |
| **Lehetséges hibaok** | kimaradt feltételvizsgálat |
| **Változtatások** | a stephandlerben vizsgáljuk ezt a feltételt |

### Lépés mérlegre – ajtó kinyílik

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lépés mérlegre – ajtó zárva marad

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lépés mérlegről – ajtó bezárul

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lépés féregjáratba

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Lépés árokba

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Replikátor lépése stepabble mezőre

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Replikátor lépése nem stepabble mezőre

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Replikátor árokba lépése

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### Jaffa ZPM felvétel

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### O’Neil páratlan sorszámú ZPM felvétele

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

### O’Neil páros sorszámú ZPM felvétele

|  |  |
| --- | --- |
| **Tesztelő neve** | Ritter |
| **Teszt időpontja** | 2016.04.24 |

## Értékelés

|  |  |
| --- | --- |
| **Tag neve** | **Munka százalékban** |
| Borbély | 20 % |
| Gáspár | 17 % |
| Makki | 15 % |
| Papp | 15 % |
| Ritter | 17 % |
| Zöld | 16 % |

## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2016.04.15. 15:30 | 3,5 óra | Zöld | Step megvalósítása |
| 2016.04.18. 10:00 | 8 óra | Borbély | Alapvető működés |
| 2016.04.19. 22:00 | 2 óra | Papp | Replikátor működés |
| 2016.04.20. 14:00 | 2 óra | Makki | Pályaszerkesztő |
| 2016.04.22. 14:00 | 3,5 óra | Gáspár | Tesztkörnyezet |
| 2016.04.24. 15:00 | 5 óra | Ritter | Tesztelés |

# 11. Grafikus felület specifikációja

18 – don

Konzulens:

Ferencz Endre

Csapattagok

|  |  |  |
| --- | --- | --- |
| Borbély László Benedek | V19RPS | [laci.dj@gmail.com](mailto:laci.dj@gmail.com) |
| Gáspár Vilmos | GF4UZL | [gaspar.vilmos24@gmail.com](mailto:gaspar.vilmos24@gmail.com) |
| Makki Viktor | I0UA10 | [deldemonho@gmail.com](mailto:deldemonho@gmail.com) |
| Ritter Alex | GS9ERJ | [ritt.alex@gmail.com](mailto:ritt.alex@gmail.com) |
| Papp Albert | F9YTAY | [albertxpapp@gmail.com](mailto:albertxpapp@gmail.com) |
| Zöld Máté | GQH7FW | [mate951006@gmail.com](mailto:mate951006@gmail.com) |
|  |  |  |

2016. május. 2.

# Grafikus felület specifikációja

## A grafikus interfész

A játék kezdő menüje, egy opcióval:

* Levels: pálya kiválasztása. (L gomb)

**A játékban szereplő objektumok:**

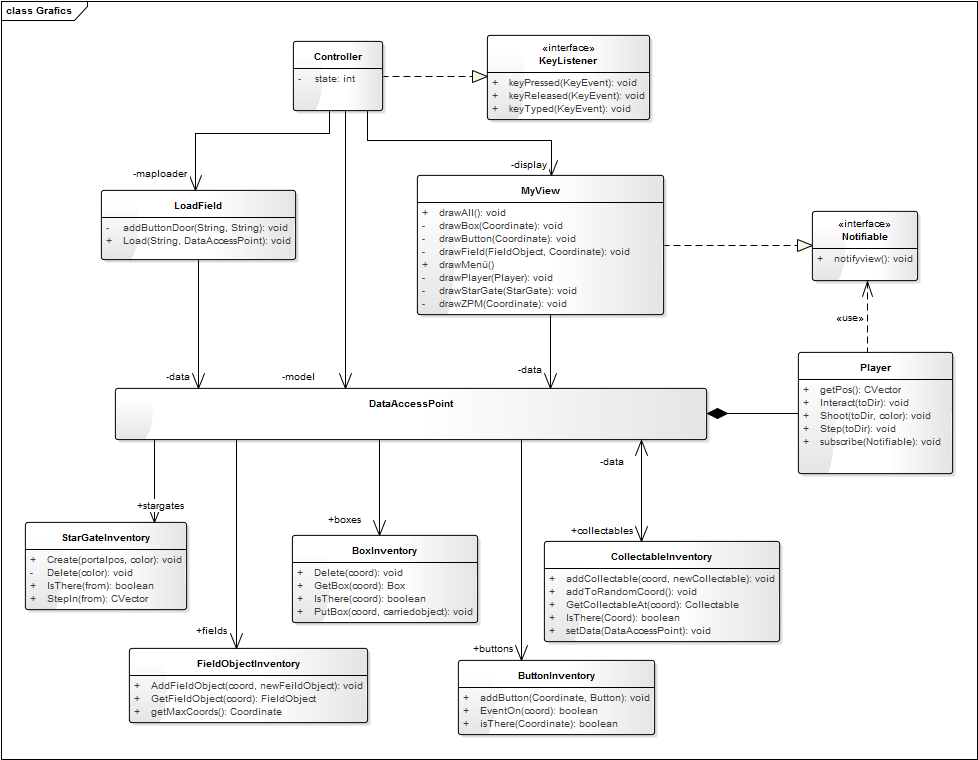
|  |  |  |  |
| --- | --- | --- | --- |
| **Fal** | **Speciális fal** | **Csillagkapu** | **Út** |
| **wall** | **SpecWall** | **rigthBlueGate** | **way** |
| **Gomb** | **Doboz** | **Ezredes** | **Jaffa** |
| **ButtonClosed** | **box** | **OneillUpStand** | **Jaffa** |
| **Replicator** | **Abyss** |  |  |
| **Replicator** |  |  |  |

## A grafikus rendszer architektúrája

### A felület működési elve

A grafikus osztályok működése kevert push-pull architektúrát követ. Azaz a kirajzolás csak a játék állapotának megváltozásakor zajlik le. Ugyanakkor a kirajzolandó objektumokat már nem a Model adja át a kirajzolást végző osztálynak, hanem az egy központi adattároló egységről kerül lekérdezésre közvetlenül a megjelenítés előtt.

### A felület osztály-struktúrája



## A grafikus objektumok felsorolása

### MyView

#### Felelősség

A pályán található objektumok grafikus megjelenítése.

#### Ősosztályok

Nincs

#### Interfészek

Notifiable

#### Attribútumok

* **- data: DataAccessPoint**: A pályán található objektumok elérésére szolgál.

#### Metódusok

* **+ void drawAll()**: A pályán található összes objektumot kirajzolja.
* **- void drawBox(Coordinate coord)**: A paraméterként átvett koordinátára kirajzol egy dobozt
* -  **void drawButton(Coordinate coord):** A paraméterként átvett koordinátára kirajzol egy nyomógombot.
* **- void drawFieldObject(FieldObject fieldObject, Coordinate coord)**: A paraméterként átvett FieldObjectet kirajzolja az átvett koordinátára. Az alapján dönti, el hogy az adott koordinátán milyen FieildObject van, hogy a Shootable, IsMortal, Steppable metódusai mivel térnek vissza.
* **- void drawPlayer(Player player)**: Kirajzolja a paraméterként átvett játékos avatarját.
* -  **void drawStarGate(StarGate stargate):** Kirajzolja a paraméterként átvett csillagkaput.
* **- void drawZPM(Coordinate coord)**: Kirajzol egy ZPM-et a paraméterként átvett koordinátára.
* **+ void notifyview()**: Értesíti a MyView-t, hogy változás történt a játék állapotában.

### Notifiable

#### Felelősség

Interfész, amely lehetővé teszi a különböző nézetek egységes értesítését, a játékban történt változásról.

#### Ősosztályok

Nincs

#### Metódusok

* **+ void notifyview()**: Értesíti az interfészt megvalósító osztályt, hogy a játékban változás történt.

### Player

#### Metódusok

* **+ void subscribe(Notifiable notifiable)**: A játékos ezzel a metódussal fel tud iratkozni egy értesíthető nézetre.

### Controller

#### Felelősség

Annak a kezelése, hogy az egyes billentyűk lenyomására megfelelő dolog történyen.

#### Ősosztályok

Nincs

#### Interfészek

KeyListener

#### Attribútumok

* **- state: int**: A játék állapotának tárolása
* **- maploader: LoadField**: Pályabetőltő osztály
* **- model: Player**: A mozgatható játékosok akcióinak hívása.
* **- display: MyView**: Pályabetőltő osztály

#### Metódusok

Nincs

### LoadField

#### Felelősség

A pálya betöltésére szolgál.

#### Ősosztályok

Nincs

#### Interfészek

Nincs

#### Attribútumok

* **- data: DataAccessPoint**: A pályán található objektumok elérésére szolgál.

#### Metódusok

* **+ void Load(String name, DateAcessPoint data)**: A megfelelő pálya betöltése.
* **- void addButtonDoor(String d, String b)**: A gombhoz ajtó rendelése.

### KeyListener

#### Felelősség

A billentyűzet állapotváltozásának érzékelése.

#### Ősosztályok

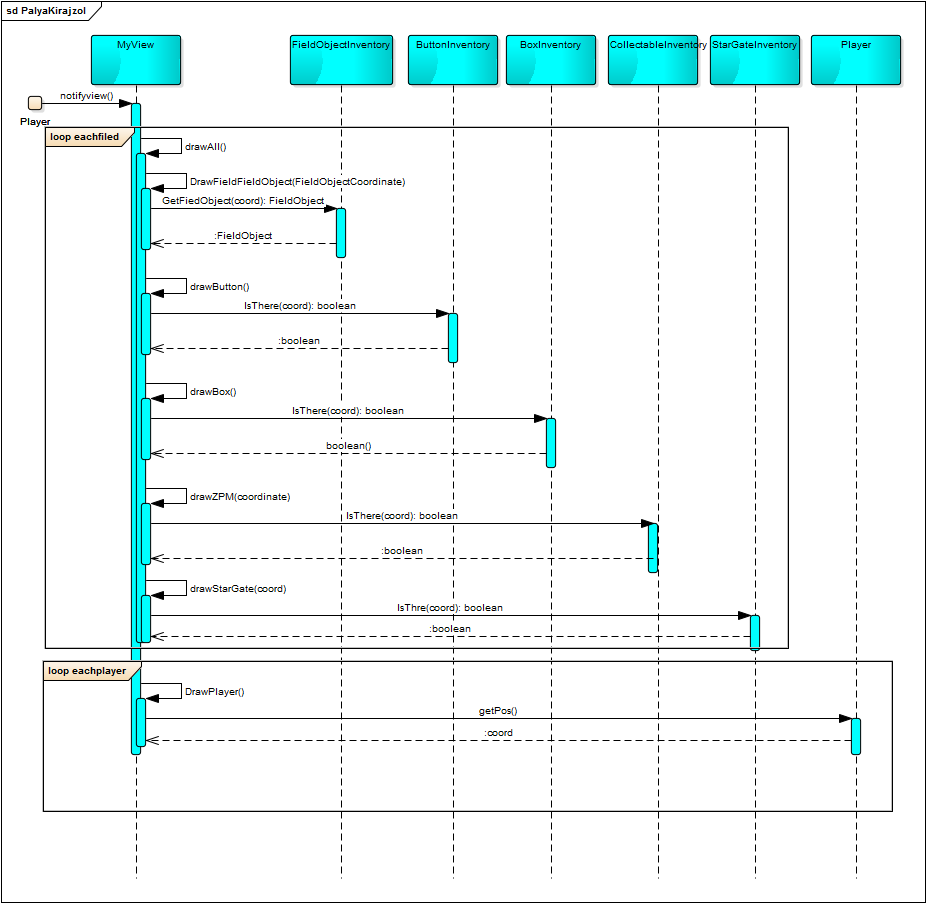
Nincs

#### Metódusok

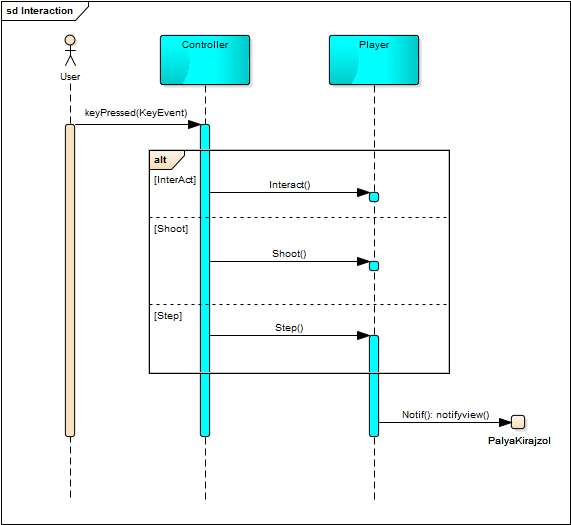
* **+ void keyPressed(KeyEvent e)**: A billentyű lenyomás érzékelése.
* **+ void keyReleased(KeyEvent e)**: A billentyű felengedés érzékelése.
* **+ void keyTyped(KeyEvent e)**: A billentyű nyomás érzékelése.

## Kapcsolat az alkalmazói rendszerrel

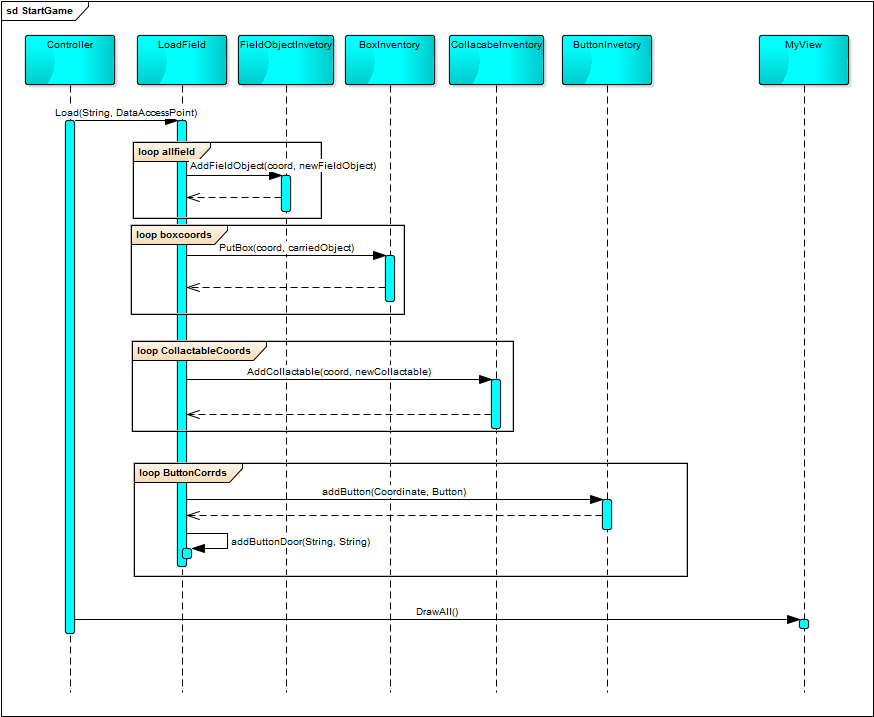
**Pályakirajzolása:**

****

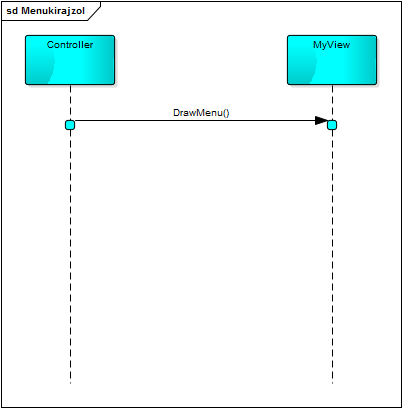
**Játékos interakció-ja:**

****

**Játék betöltése:**

****

**Menü kirajzolása:**



## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2016.04.29. 18:00 | 2 óra | Borbély, Gáspár | Grafikus rendszer architektúrája |
| 2016.04.29 18:00 | 2 óra | Makki | Grafikus interfész |
| 2016.05.01. 10:00 | fél óra | Zöld | MyView, Notifiable osztály leírások és Player változása |
| 2016.05.01 17:00 | 3 óra | Papp | Szekvencia diatrammok, Controller osztály létrehozása |
| 2016.05.01 19:00 | fél óra | Ritter | Controller, LoadField, KeyListener osztály leírások |
| 2016.05.02 10:00 | fél óra | Gáspár | Korrektúra |

# Grafikus változat beadása

18 – don

Konzulens:

Ferencz Endre

Csapattagok

|  |  |  |
| --- | --- | --- |
| Borbély László Benedek | V19RPS | [laci.dj@gmail.com](mailto:laci.dj@gmail.com) |
| Gáspár Vilmos | GF4UZL | [gaspar.vilmos24@gmail.com](mailto:gaspar.vilmos24@gmail.com) |
| Makki Viktor | I0UA10 | [deldemonho@gmail.com](mailto:deldemonho@gmail.com) |
| Ritter Alex | GS9ERJ | [ritt.alex@gmail.com](mailto:ritt.alex@gmail.com) |
| Papp Albert | F9YTAY | [albertxpapp@gmail.com](mailto:albertxpapp@gmail.com) |
| Zöld Máté | GQH7FW | [mate951006@gmail.com](mailto:mate951006@gmail.com) |
|  |  |  |

2016. május 17.

# Grafikus változat beadása

## Fordítási és futtatási útmutató

### Fájllista

|  |  |  |  |
| --- | --- | --- | --- |
| **Fájl neve** | **Méret** | **Keletkezés ideje** | **Tartalom** |
| Abyss.java | 1 KB | 2016.05.16 | Szakadék tulajdonságai |
| Basic2DView.java | 30 KB | 2016.05.17 | A megjelenítést végző osztály |
| Box.java | 1 KB | 2016.05.16 | Doboz tulajdonságai |
| BoxInventory.java | 4 KB | 2016.05.16 | Dobozok tárolása |
| Button.java | 2 KB | 2016.05.17 | Nyomólapok tulajdonságai |
| ButtonInventory.java | 3 KB | 2016.05.17 | Nyomólapok tárolása |
| Collectible.java | 1 KB | 2016.05.16 | Felszedhető objektumok tulajdonságai |
| CollectableInventory.java | 4 KB | 2016.05.16 | Felszedhető objektumok tárolása |
| Colonel.java | 2 KB | 2016.05.16 | Az ezredes |
| Controller.java | 5 KB | 2016.05.16 | Vezérlő osztály |
| Coordinate.java | 2 KB | 2016.05.16 | Koordináták tárolása |
| CustomHashMap.java | 2 KB | 2016.05.16 | Saját HashMap megvalósítása |
| Cvector.java | 3 KB | 2016.05.17 | Koordináták iránnyal tárolása |
| DataAcessPoint.java | 2 KB | 2016.05.16 | Adat összekötetést biztosít |
| Direction.java | 1 KB | 2016.05.16 | Irány enum |
| Door.java | 2 KB | 2016.05.16 | Ajtó tulajdonságai |
| FieldObject.java | 1 KB | 2016.05.16 | FieldObject tulajdonságai |
| FieldObjectInventory.java | 2 KB | 2016.05.16 | FieldObject tárolása |
| GameHandler.java | 6 KB | 2016.05.16 | Menu (main) |
| GrabHandler.java | 4 KB | 2016.05.17 | Objektum felvétének ill. lerakásának esetei lekezelése |
| Jaffa.java | 2 KB | 2016.05.16 | A második játékos |
| LoadField.java | 5 KB | 2016.05.16 | Pályát betöltő osztály |
| Notifiable.java | 1 KB | 2016.05.16 | Értesíthető interfész |
| Player.java | 5 KB | 2016.05.17 | Játékos interakciói |
| Replicator.java | 2 KB | 2016.05.16 | Eldöntető kérdésre form |
| ShotHandler.java | 2 KB | 2016.05.16 | Lövés eseteinek lekezelése |
| SpecialWall.java | 1 KB | 2016.05.16 | Speciális fal tulajdonságai |
| Stargate.java | 1 KB | 2016.05.16 | Csillagkapu tulajdonságai |
| StargateColor.java | 1 KB | 2016.05.16 | Csillagkapuszíne enum |
| StargateInventory.java | 6 KB | 2016.05.16 | Csillagkapu tárolása |
| StepHandler.java | 4 KB | 2016.05.17 | Lépés eseteinek lekezelése |
| Wall.java | 1 KB | 2016.05.16 | Fal tulajdonságai |
| Way.java | 1 KB | 2016.05.16 | Út tulajdonságai |
| ZPM.java | 1 KB | 2016.05.16 | ZPM tulajdonságai |
| downJaffaStep1.png | 115 KB | 2016.05.16 | Jaffa kép |
| downOneilUpstep1.png | 105 KB | 2016.05.16 | Colonel kép |
| downReplicator.png | 21 KB | 2016.05.16 | Replikátor kép |
| upJaffaStep1.png | 123 KB | 2016.05.16 | Jaffa kép |
| upOneilUpstep1.png | 112 KB | 2016.05.16 | Colonel kép |
| upReplicator.png | 28 KB | 2016.05.16 | Replikátor kép |
| leftJaffaStep1.png | 115 KB | 2016.05.16 | Jaffa kép |
| leftOneilUpstep1.png | 111 KB | 2016.05.16 | Colonel kép |
| leftReplicator.png | 29 KB | 2016.05.16 | Replikátor kép |
| rightJaffaStep1.png | 115 KB | 2016.05.16 | Jaffa kép |
| rightOneilUpstep1.png | 111 KB | 2016.05.16 | Colonel kép |
| rightReplicator.png | 28 KB | 2016.05.16 | Jaffa kép |
| abyss.png | 42 KB | 2016.05.16 | Szakadék kép |
| box.png | 27 KB | 2016.05.16 | Doboz kép |
| ButtonClosed.png | 120 KB | 2016.05.16 | Zárt nyomólap kép |
| ButtonOpened.png | 111 KB | 2016.05.16 | Nyitott nyomólap kép |
| DoorClosed.png | 110 KB | 2016.05.16 | Zárt ajtó kép |
| DoorOpened.png | 155 KB | 2016.05.16 | Nyitott ajtó kép |
| downBlueGate.png | 115 KB | 2016.05.17 | Kék portál |
| downGreenGate.png | 109 KB | 2016.05.17 | Zöld portál |
| downRedGate.png | 115 KB | 2016.05.17 | Piros portál |
| downYellowGate.png | 110 KB | 2016.05.17 | Sárga portál |
| leftBlueGate.png | 120 KB | 2016.05.17 | Kék portál |
| leftGreenGate.png | 128 KB | 2016.05.17 | Zöld portál |
| leftRedGate.png | 133 KB | 2016.05.17 | Piros portál |
| leftYellowGate.png | 143 KB | 2016.05.17 | Sárga portál |
| rightBlueGate.png | 133 KB | 2016.05.17 | Kék portál |
| rightGreenGate.png | 126 KB | 2016.05.17 | Zöld portál |
| rightRedGate.png | 134 KB | 2016.05.17 | Piros portál |
| rightYellowGate.png | 145 KB | 2016.05.17 | Sárga portál |
| SpecWall.png | 227 KB | 2016.05.16 | Speciális fal kép |
| upBlueGate.png | 126 KB | 2016.05.17 | Kék portál |
| upGreenGate.png | 119 KB | 2016.05.17 | Zöld portál |
| upRedGate.png | 129 KB | 2016.05.17 | Piros portál |
| upYellowGate.png | 122 KB | 2016.05.17 | Sárga portál |
| wall.png | 116 KB | 2016.05.16 | Fal kép |
| way.png | 574 KB | 2016.05.16 | Út kép |
| ZPM.png | 35 KB | 2016.05.16 | ZPM kép |
| config.ini | 1 KB | 2016.05.16 | Konfigurációs fájl |
| SingleLevel1.txt | 1 KB | 2016.05.16 | Pálya fájl |
| SingleLevel2.txt | 1 KB | 2016.05.17 | Pálya fájl |
| SingleLevel3.txt | 1 KB | 2016.05.17 | Pálya fájl |
| SingleLevel4.txt | 1 KB | 2016.05.17 | Pálya fájl |
| SingleLevel5.txt | 1 KB | 2016.05.17 | Pálya fájl |
| SingleLevel6.txt | 1 KB | 2016.05.16 | Pálya fájl |
| SingleLevel7.txt | 4 KB | 2016.05.16 | Pálya fájl |
| SingleLevel8.txt | 1 KB | 2016.05.17 | Pálya fájl |
| SingleLevel9.txt | 1 KB | 2016.05.17 | Pálya fájl |
| SingleLevel10.txt | 1 KB | 2016.05.17 | Pálya fájl |

### Fordítás és telepítés

A forrásfájlok segítségével parancssorból a következő parancsot kell kiadni:

> javac –d build GameObjects/\*.java Handlers/\*.java Inventories/\*.java Players/\*.java Tools/\*.java Graphics/\*.java

### Futtatás

A build mappában a következő parancs segítségével futtatható a játék:

> java Tools.Controller

## Értékelés

|  |  |
| --- | --- |
| **Tag neve** | **Munka százalékban** |
| Borbély | 18% |
| Gáspár | 18% |
| Makki | 16% |
| Ritter | 16% |
| Papp | 16% |
| Zöld | 16% |

## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2016.05.15. 18:00 | 3 óra | Gáspár | Menü |
| 2016.05.16. 10:00 | 4 óra | Borbély | Grafika |
| 2016.05.16. 14:00 | 1,3 óra | Zöld | Pályaszerkesztés, grafika |
| 2016.05.16. 18:00 | 2,5 óra | Papp | Képszerkesztés, dokumentáció, grafika |
| 2016.05.16. 18:30 | 1 óra | Ritter | Pályaszerkesztés |
| 2016.05.16. 19:00 | 1 óra | Makki | Pályaszerkesztés |