# 

# 

The Deserts of Drukmakor

# 

# 

# Konzulens:

# Dobos-Kovács Mihály

# 

# 

| Sütheö István | XOBJYX | istvan.sutheo@gmail.com (kapcsolattartó) |
| --- | --- | --- |
| Vörös László | SLLV0S | lacko1001@gmail.com |
| Gerencsér Máté | YVSPMN | gerencsermate02@gmail.com |
| Bognár Zsombor | BNO3XE | bzsombi99@gmail.com |
| Duy Anh Chu | G4H99J | chuduyanh1@gmail.com |

# 

# 2023.03.13.

# (A egyes fejezetek sorszámát az határozza meg, hogy hányadik héten került beadásra a feladat)

# 2. Követelmény, projekt, funkcionalitás

## 2.1 Bevezetés

### 2.1.1 Cél

*A projekt bemutatása és ismertetése mind követelményi mind funkcionalitás szempontjából. A fejlesztett programnál ezeket a funkciókat kell szem előtt tartani és megvalósítani.*

### 2.1.2 Szakterület

*A szoftver egy játékprogram, ami a szórakoztatásra készül. Ez egy többszemélyes játék, ahol szerelőket és szabotőröket kell irányítani. A két csapat egymás ellen játszik, a szerelőknek az a célja hogy minél több víz összegyűjtése, a szabotőröknek minél több víz elfolyatása. Az a csapat nyer, aki több pontot szerez.*

### 2.1.3 Definíciók, rövidítések

* *Java - Objektumorientált programnyelv, a projekt kötelező nyelve.*
* *Google Drive - Felhő Szolgáltatás: közös tárhely és fájl szerkesztéseknek a helye.*
* *Git - verziókezelő rendszer*
* *Intellij IDEA - Java fejlesztői környezet (IDE)*
* *IDE – fejlesztői környezet*
* *JDK - Java Development Kit: java fejlesztőknek szánt környezete, magában foglalja a JRE-t*
* *JRE - Java Runtime Environment: Java-t futtató környezet*
* *Google Docs – Online Szövegszerkesztő program*
* *Docs– lásd: Google Docs*

### 2.1.4 Hivatkozások

*https://www.iit.bme.hu/targyak/BMEVIIIAB02*

### 2.1.5 Összefoglalás

*A dokumentum további részében:*

* *Áttekintjük a főbb tulajdonságokat.*
* *A betartandó követelmények felsorolása.*
* *Use Casek felsorolása.*
* *Egy közös szótár létrehozása, amit egységesen fogunk használni és részletezi az esetleges ismeretlen kifejezéseket.*
* *Egy közös projekt terv, amiben leírjuk*
* *Egy közösen vezetett projektnapló.*

## 2.2 Áttekintés

### 2.2.1 Általános áttekintés

*A játék offline játszható, többfelhasználós, és több billentyűzet inputtal lehet irányítani a játékosokat. A felhasználók egy közös grafikus felhasználói felületen keresztül kommunikálnak a rendszerrel. Az egyes funkciók és lépések itt lesznek láthatóak és könnyen értelmezhetőek a felhaszlnálók számára.*

*A játék indulásakor egy forrás és egy ciszterna lesz elérhető a pályán. A szerelők a forráson, a szabotőrök a ciszternán kezdenek. Ezután a szerelők tudnak új csöveket és pumpákat létrehozni a pályán.*

*A játék közben egy másik alrendszer fog gondoskodni arról, hogy minden a követelmények alapján működjön. A feladatai többek között a játékosok helyes mozgatása, a csőrendszerben működtetése, pontok számítása, új ciszternák készítése.*

*Legfontosabb alrendszerek:*

* *grafikus felhasználói felület*
* *játékmotor*

### 2.2.2 Funkciók

A játék neve a Sivatagi vízhálózat. A drukmákori sivatagban játszunk, ahol egy vízvezeték rendszert akarunk telepíteni csövek és pumpák, hegyi források segítségével a városok ciszternáiba. Vannak szerelők, akik ezt a csőrendszert építik és tartják karban és vannak a gonosz nomád szabotőrök, akik a csőrendszert tudják elrontani és ezáltal a vízfolyást akadályozni.

A játékot a két csapat legalább kettő-kettő játékossal játssza. A játék céjla szabotőröknek, hogy minél több víz folyjon el a sivatagba, hogy a megkeserítse a városokban élő emberek életét. A szerelők pedig azon dolgoznak, hogy minél több víz jusson a ciszternákba. Az a csapat nyer, amelyik a játék végére több vizet szerez.

A víz a sivatagon keresztül a víz a forrásoktól a ciszternákig a csőrendszeren keresztül szállítódik, amit a szerelők a játék közben építenek fel. Egy cső két játékbeli elemet köt össze. Ez az elem lehet egy forrás, pumpa vagy nyitott végű cső. A játékosok csak játékbeli elemeket közlekedhetnek, a sivatagban nem. Egy csövön egyszerre csak egy játékos lehet, nem férnek el egymás mellett. A csöveknek van kapacitása, vagyis egyszerre csak ennyi vízet tud elvinni a következő játékelemig. A csövek elágazás nélküliek. Ha egy cső kilyukad és folyik bele víz akkor az a sivatagban elfolyik, illetve a szabad végű csövekből is a sivatagba folyik a betáplált víz.

A csőrendszerbe elágazásokat a pumpákkal lehet létrehozni, az maximális vízmennyiség elérése érdekében. Egy pumpába bármennyi cső csatlakozhat, de egyszerre csak egy be- és kimenete van. Ezeket a játékosok tudják állítani. Minden pumpának van egy tartálya, amiben tárolja a beérkező vizet, és használja átmeneti tárolóként a víz továbbítása közben. Hogyha nem érkezik be több víz, a benne lévő vizet még továbbadja. A pumpák néha el tudnak romlani, amit a szerelők tudnak megjavítani. Ha egy pumpa elromlik megszűnik benne a vízáramlás. Pumpákat a szerelők tudnak magukhoz venni a ciszternáktól. Egyszerre egy lehet egy szerelőnél, amit egy cső közben tud lehelyezni, annak a szétvágásával. Ha a pumpa bemeneti csöve ki van lyukadva, akkor a pumpához már nem jut el víz.

A játékba a víz a forrásból jön be, ami a pálya legelején van. Végtelen mennyiségű vizet lehet kinyerni belőle. Bármennyi csövet lehet kivezetni belőle a csőrendszerbe a vízáramlás biztosításához.

A városba a víz a ciszternán keresztül jut el, ami a pálya legvégén van. Végtelen mennyiségű vizet tud elnyelni. Ebbe nem tudnak a szerelők csöveket csatlakoztatni, viszont a játék közben bizonyos időközönként új szabad csővég lesz a ciszternán, amihez már lehet új bemeneteket csatlakoztatni.

A csőhálózatot a szerelők építik fel és tartják karban. Csak a játékelemeken közlekedhetnek. Egy csövön nem fér el egymás mellett két játékos, de a többi játékelemen (pumpa, forrás, ciszterna) igen, így ezeken tudják egymást kikerülni. A szerelők tudják megjavítani a pumpákat és befoltozni az kilyukadt csöveket. Ezen kívül össze tudnak játékelemeket kötni csövekkel, és új pumpákat a játékba vinni, valamint a pumpák ki és bemenetét állítani. A szerelők célja, hogy minél több víz jusson el a városba a ciszternákon keresztül, és ehhez a csőrendszer kiépítése.

A sivatagban élnek még nomád szabotőrök is, akik a szerelők ellen vannak és a céljuk a csőrendszer szabotálása. Ők is tudják a pumpák ki és bemenetét állítani, illetve a csöveken kilyukasztani és ezzel elfolyatni a vizet a sivatagban. Nekik a céljuk, hogy minél több víz folyjon el és ezzel megakadályozzák a szerelőket a víz városba juttatásában.

### 2.2.3 Felhasználók

*A felhasználók csak akkor tudnak játszani a programmal, ha mindkettő csapat legalább rendelkezik 2-2 játékossal. A felhasználóknak azt tanácsoljunk, hogy legalább 12+ évesek legyenek. A programot azoknak ajánljuk akik szeretnek csoportban stratégiai játékokkal játszani.*

### 2.2.4 Korlátozások

*A szoftvert JAVA nyelven kell, hogy írnunk.  
Minden hét hétfőjén le kell adni egy rész feladatot.   
A szoftvernek a 14. oktatási hét péntek 13 óráig (2023.06.02.) kész kell lennie működő állapotban.*

### 2.2.5 Feltételezések, kapcsolatok

*docs.google.com - itt szerkeztjük a dokumentációkat.  
https://www.iit.bme.hu/file/11582/feladat*

## 2.3 Követelmények

### 2.3.1 Funkcionális követelmények

| **Azono­sí­tó** | **Leírás** | **Ellenőrzés** | **Prio­ri­tás** | **Forrás** | **Use-case** |
| --- | --- | --- | --- | --- | --- |
| 01 | A játékosok a szabotőröket és a szerelőket irányítják. | Bemutatás | Alapvető | Megrendelő | Move |
| 02 | A pálya 5 fajta mezőből épül fel: sivatag, forrás, ciszterna, cső, pumpa. | Bemutatás | Alapvető | Megrendelő | Generate Field |
| 03 | Egy mező maximum 4 másik mezővel lehet szomszédos | Bemutatás | Alapvető | Csapat | Generate Field |
| 04 | A játékosok csak a cső és pumpa típusú mezőkön tudnak mozogni. | Bemutatás | Alapvető | Megrendelő | Move |
| 05 | Egy cső mezőn egyszerre csak egy játékos tartózkodhat. | Bemutatás | Alapvető | Megrendelő | Move |
| 06 | A játékosok a pumpa típusú mezőkön tudják kikerülni egymást. | Bemutatás | Alapvető | Megrendelő | Move |
| 07 | A vízhálózatot a forrás mezők látják el vízzel. | Bemutatás | Alapvető | Megrendelő | Control Water |
| 08 | A szabotőrök ki tudják lyukasztani a csöveket. | Bemutatás | Alapvető | Megrendelő | Damage Pipe |
| 09 | A lyukas csövekből a sivatagba ömlik a víz. | Kiértékelés | Alapvető | Megrendelő | Control Water |
| 10 | A pumpák időnként elromlanak. | Kiértékelés | Alapvető | Megrendelő | Control Pump |
| 11 | Az elromlott pumpákon nem áramlik keresztül víz. | Kiértékelés | Fontos | Megrendelő | Control Pump |
| 12 | A szerelők meg tudják javítani a meghibásodott pumpákat. | Bemutatás | Fontos | Megrendelő | Repair Pump |
| 13 | A szerelők a csövek közepére új pumpát tudnak telepíteni. | Bemutatás | Alapvető | Megrendelő | Assemble Pump |
| 14 | Egy pumpába csak egy csövön folyhat be és egy csövön folyhat ki víz. | Kiértékelés | Alapvető | Megrendelő | Control Pump |
| 15 | A játékosok beállíthatják melyik két cső között szállítson vizet a pumpa. | Bemutatás | Fontos | Megrendelő | Control Pump |
| 16 | A ciszternáknál a játék során új csövek keletkeznek. | Bemutatás | Fontos | Megrendelő | Generate Field |
| 17 | A szabad végű csövekből a csőbe betáplált víz a sivatagba elfolyik. | Kiértékelés | Fontos | Megrendelő | Control Water |
| 18 | A pumpák egy adott kapacitású tartállyal rendelkeznek, amiben víz tárolódik el. | Kiértékelés | Fontos | Megrendelő | Control Pump |
| 19 | A csövek egy adott kapacitással rendelkeznek. | Kiértékelés | Opcionális | Csapat | Control Water |
| 20 | Több ciszterna mező keletkezik a játék elején. | Bemutatás | Fontos | Csapat | Generate Field |
| 21 | Több forrás mező keletkezik a játék elején. | Bemutatás | Fontos | Csapat | Generate Field |
| 22 | A csövek egyik végének csatlakoznia kell egy pumpához. | Bemutatás | Fontos | Megrendelő | Generate Field |
| 23 | Ha a víz a sivatagba folyik el, akkor a szabotőrök kapnak pontot. | Kiértékelés | Fontos | Megrendelő | Control Water |
| 24 | Ha a víz a ciszternába folyik el, akkor a szerelők kapnak pontot. | Kiértékelés | Fontos | Megrendelő | Control Water |
| 25 | Egy pumpához maximum 4 cső csatlakozhat. | Bemutatás | Fontos | Csapat | Generate Field |
| 26 | Két pumpát legalább 1 cső mező kell, hogy összekapcsoljon. | Bemutatás | Fontos | Csapat | Generate Field |

### 2.3.2 Erőforrásokkal kapcsolatos követelmények

| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| --- | --- | --- | --- | --- | --- |
| 01 | IntelliJ IDEA | nincs | Opcionális | csapat | Java IDE |
| 02 | Eclipse | nincs | Opcionális | csapat | Java fejlesztőkörnyezet |
| 03 | Google Drive | nincs | fontos | csapat | Dokumentumok szerkesztése és megosztása |
| 04 | Discord | nincs | fontos | csapat | Online hivások, kommunikáció, kapcsolattartás |
| 05 | Messenger | nincs | fontos | csapat | Gyors kommunikáció, kapcsolattartás, fontos kérdések |
| 06 | Git | nincs | fontos | csapat | Verziókezelés |
| 07 | Email | nincs | alapvető | megrendelő | Kapcsolattartás a megrendelővel. |
| 08 | Microsoft Teams | nincs | alapvető | megrendelő | Kapcsolattartás a megrendelővel. |
| 09 | Hercules feladatbeadó rendszer | bemutatás | alapvető | megrendelő | Kapcsolattartás a megrendelővel. |
| 10 | IIT.bme.hu weboldala | nincs | alapvető | megrendelő | A programról, feladatokról információk megszerzése. |
| 11 | Laptop, számítógép | kiértékelés | alapvető | csapat | A program fejlesztéséhez használt eszközök. |
| 12 | Telefon | nincs | alapvető | csapat | A kommunikáció leggyorsabb módszere. |
| 13 | Perifériák | kiértékelés | alapvető | csapat | Egér, monitor, billentyűzet, hangszóró, mikrofon |

### 2.3.3 Átadással kapcsolatos követelmények

| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| --- | --- | --- | --- | --- | --- |
| 1 | Szkeleton átadás | Bemutatás | Alapvető | Megrendelő | április 17. |
| 2 | Prototípus átadás | Bemutatás | Alapvető | Megrendelő | május 8. |
| 3 | Kész program átadás | Bemutatás | Alapvető | Megrendelő | június 2. |

### 2.3.4 Egyéb nem funkcionális követelmények

Nincs egyéb nem funkcionális követelmény.

## 2.4 Lényeges use-case-ek

### 2.4.1 Use-case leírások

| **Use-case neve** | Move |
| --- | --- |
| **Rövid leírás** | A játékosok csöveken és pumpákon mozoghatnak. |
| **Aktorok** | Player |
| **Forgatókönyv** | Egy pumpáról játékos egy üres csőre lép. |
| **Alternatív forgatókönyv** | Egy játékos egy csőről egy üres pumpára lép. |
| **Alternatív forgatókönyv** | Egy játékos egy csőről olyan pumpára lép, amelyiken egy másik játékos tartózkodik. |
| **Alternatív forgatókönyv** | A játékos egy pumpán áll és egy olyan csőre próbál lépni, amin már van egy játékos, ekkor a lépni akaró játékos a pumpán marad. |

| **Use-case neve** | Set |
| --- | --- |
| **Rövid leírás** | A játékosok beállíthatják hogy, a pumpához csatlakoztatott csövek közül melyik kettő között folyjon víz. |
| **Aktorok** | Player |
| **Forgatókönyv** | A játékos kiválaszt egy csövet amelyikből a pumpa beszívja a vizet és egyet, amibe belepumpálja. |

| **Use-case neve** | Assemble Pump |
| --- | --- |
| **Rövid leírás** | A szerelők egy új pumpát telepítenek. |
| **Aktorok** | Repairmen |
| **Forgatókönyv** | A szerelő egy csövet kettévág és pumpával összeköti a két szétvágott csővéget. |

| **Use-case neve** | Repair pipe |
| --- | --- |
| **Rövid leírás** | A szerelők megfoltozzák a lyukas csöveket. |
| **Aktorok** | Repairmen |
| **Forgatókönyv** | A szerelő befoltozza a lyukat, ekkor a csőbe áramló víz újra a hálózatban áramlik tovább, ahelyett, hogy a sivatagba folyna. |

### 

| **Use-case neve** | Repair Pump |
| --- | --- |
| **Rövid leírás** | A szerelők megjavítják a meghibásodott pumpát. |
| **Aktorok** | Repairmen |
| **Forgatókönyv** | A szerelő megjavítja a hibás pumpát, ekkor a pumpa újra elkezd a két kiválasztott cső között vizet szállítani. |

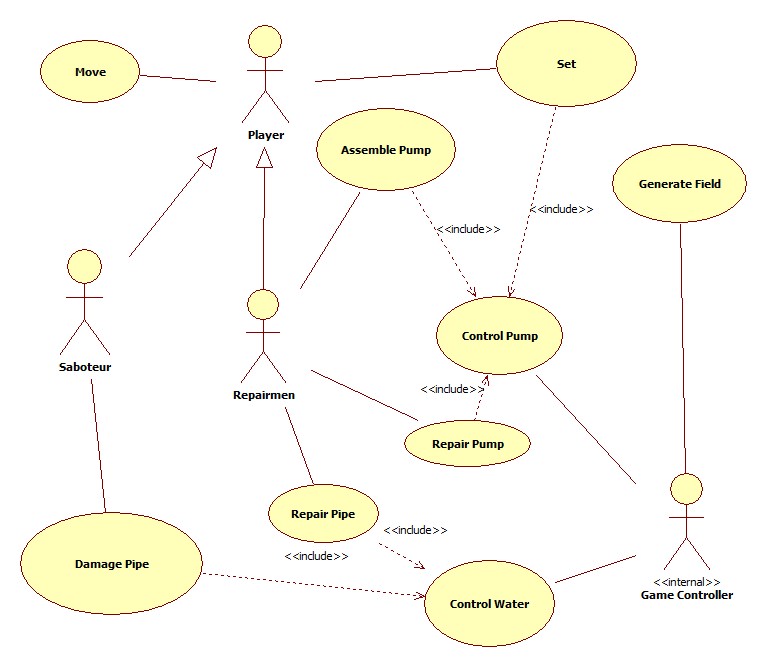
| **Use-case neve** | Control Pump |
| --- | --- |
| **Rövid leírás** | A pumpa működésének vezérlése. |
| **Aktorok** | Game Controller |
| **Forgatókönyv** | A pumpa a kiválasztott két cső között pumpálja a vizet. |

| **Use-case neve** | Control Water |
| --- | --- |
| **Rövid leírás** | Az áramlását szabályozza a hálózatban. |
| **Aktorok** | Game Controller |
| **Forgatókönyv** | Egy cső egy pumpához csatlakozik, a pumpa vizet szív ki a csőből. |
| **Alternatív forgatókönyv** | Egy cső egy pumpához csatlakozik, a pumpa vizet pumpál a csőbe. |
| **Alternatív forgatókönyv** | A cső lyukas, ekkor belőle a víz a sivatagba folyik. |
| **Alternatív forgatókönyv** | A cső egyik vége forráshoz csatlakozik, ekkor a forrásból víz áramlik a csőbe. |
| **Alternatív forgatókönyv** | A cső egyik vége ciszternához csatlakozik, ekkor a csőből víz áramlik a ciszternába. |

| **Use-case neve** | Generate Field |
| --- | --- |
| **Rövid leírás** | A pálya létrehozása. |
| **Aktorok** | Game Controller |
| **Forgatókönyv** | A játék elején egy véletlenszerűen generált hálózat jön létre. |

### 

### 2.4.2 Use-case diagram



## 2.5 Szótár

Befoltoz - Egy lyukas cső megjavítása, ekkor már nem folyat a cső.

Ciszterna - A pályának egy olyan eleme, ahova a víz eljuthat.

Cső - A pályának egy olyan eleme, amelyen víz folyik keresztül.

Elromlik - A pumpának egy olyan állapota, amikor nem tud vizet továbbítani.

Kapacitás - Az a vízmennyiség, amennyi egy adott mező típusba belefér.

Kilyukad - Ekkor az adott csőbe befolyó víz elfolyik a sivatagba.

Sivatag - A pályának egy olyan eleme, ahova a víz eljuthat.

Szabotőr - Az egyik játékos típus, célja, hogy minél több víz folyjon el a sivatagba

Szerelő - A másik játékos típus, célja, hogy minél több víz jusson a ciszternákba.

Pumpa - A pályának egy olyan eleme, amely a vizet egy kiválasztott csőből egy másik kiválasztott csőbe továbbítja.

vízvezeték/csőrendszer - pálya része, ahol a játékosok tudnak közlekedni

forrás - az a játékelem, ahonnan a víz jön.

karbantartás - A szerelőknek kell megjavítaniuk minden elromlott elemét a csőrendszernek.

vízfolyás - A víz a csőrendszerrel való interakciója.

játékos - az a személy, aki egy játékbeli karaktert (szabotőr/szerelő) kezel.

játékelem - A játékban szereplő és interaktálható dolgok. Pumpa, szabad csővég, forrás.

elágazás - Egy cső több irányba is tudjon menni. Ezt a pumpákkal lehet megoldani a játékban.

szabad végű cső - Olyan cső, aminek az egyik vége nem csatlakozik egy játékelemhez sem.

vízmennyiség - A csőrendszer átbocsátóképessége.

csatlakozik - A cső hozzá van kötve egy játékelemhez.

bemenet - Egy játékelembe innen érkezik a víz.

kimenet - Egy jákételem ezen keresztül adja tovább a vizet.

tartály - A pumpák ebben tudnak tárolni vizet átmeneti tárolóként.

megjavít - a szerelő egyik tevékenysége. Az elromlott csöveket/pumpákat tudja újra működésbe hozni.

szétvág - Egy csövet ketté lehet szedni, egy pumpa beillesztéséhez.

közlekedik - a játékosok a csőrendszeren mozognak.

elfolyik

## 2.6 Projekt terv

### 2.6.1 Hivatalos ütemterv

| **hét** | **feladat** | **pontszám** | **laboralkalom** | **határidő** |
| --- | --- | --- | --- | --- |
| 1 | *csapatalakítás* |  | közös előadás | márc. 3. 14:00 |
| 2 | Követelmény, projekt, funkcionalitás | 10 | konzultáció | március. 13. 14:15 |
| 3 | Analízis modell (I. változat) | 20 | konzultáció | márc. 20. 14:15 |
| 4 | Analízis modell (II. változat) | 30 | konzultáció | márc. 27. 14:15 |
| 5 | Szkeleton tervezése | 20 | konzultáció | ápr. 3. 14:15 |
| 6 | Szkeleton elkészítése | 20 | konzultáció | ápr. 17. 14:15 |
| 7 | - tavaszi szünet - |
| 8 | Prototípus koncepciója | 20 | Szkeleton bemutatása | ápr. 24. 14:15 |
| 9 | Részletes tervek | 45 | konzultáció | máj. 3. labor |
| 10 | Prototípus elkészítése | 35 | konzultáció | máj. 8. 14:15 |
| 11 | Grafikus változat tervei | 30 | Prototípus bemutatása | máj. 15. 14:15 |
| 12 | Grafikus változat elkészítése | 40 | konzultáció | máj 31. labor |
| 13 | konzultáció |
| 14 | Egyesített dokumentáció | 30 | Grafikus verzió bemutatása  egyesített doksi csak elektronikusan | jún. 2. 14.00 |

### 2.6.2 Csoportmunka lebonyolítása

Alapvetően hetente kétszer tervezünk konzultációt tartani a hivatalos laboralkalmon kívül a csapattal. Az egyik a szerdai nap folyamán, ahol a konzultáción megválaszolt kérdéseket átbeszéljük egymással. A második konzultációt szombatonként fogjuk tartani, ahol a megtekintjük a csapat egyes tagjainak munkáját, hibakeresést végzünk, majd véglegesítjük a beadandó anyagot. Szükség esetén vasárnapra marad a fájlok teendőinek befejezése.

Alapvetően a szerdai megbeszélést élőben tartjuk, a többit online felületen (Discordon) összeülve tartjuk. A feladatok során egymástól emailben, valamint Messengeren tudunk kérdezni.

A dokumentumok megosztására alapvetően Git-et, Google Drive-ot, Discord-ot, valamint Google Docs-ot fogunk használni.

Ha a dokumentum beadásra kész, akkor átbeszéljük a következő heti teendőket és határidőket. A felmerülő kérdéseket összegyűjtjük, amiket a következő labor órán a konzulenssel megtárgyalunk.

### 2.6.3 Csoport tagok felelőssége

Alapvetően a projekt készítése mindenki egyenrangúnak számít, mindenki kiveszi a részét az egyes részfeladatokból.

## 2.7 Napló

| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| --- | --- | --- | --- |
| 2023.03.08. 16:30 | 2 óra | Sütheö  Vörös  Gerencsér  Bognár  Chu | Értekezlet.  Az alapvető döntések, kommunikációhoz szükséges csoportok megalakítása.  Az ötletek átbeszélése. |
| 2023.03.09 17:00 | 1 óra | Bognár | Bevezetés megírása. |
| 2023.03.09 18:00 | 2 óra | Chu | Áttekintés megírása. |
| 2023.03.10 17:00 | 1 óra | Bognár  Chu | Értekezlet  A bevezetés és áttekintés áttekintése és javítása. |
| 2023.03.11. 11:00 | 1,5 óra | Sütheö  Vörös  Gerencsér  Bognár  Chu | Értekezlet Discordon.  A feladatok kiosztása. |
| 2023.03.11 16:15 | 0,5 óra | Sütheö  Vörös  Gerencsér | Értekezlet.  Részfeladatok kiosztása. |
| 2023.03.11 16:45 | 2 óra | Sütheö | Use-case diagram, projekt terv elkészítése. |
| 2023.03.11 16:45 | 2 óra | Gerencsér | Use-case leírások megfogalmazása. |
| 2023.03.11 17:00 | 2 óra | Vörös | Követelmények megírása. Áttekintés-felhasználók megírása. |
| 2023.03.11 21:30 | 1 óra | Vörös | Követelmények pontosítása, véglegesítése. |
| 2023.03.12. 13:00 | 1,5 óra | Sütheö | Szótár megírása. |
| 2023.03.12. 18:15 | 1 óra | Gerencsér | Dokumentum átvizsgálása, kisebb hibák javítása. |

# 3. Analízis modell kidolgozása

## 3.1 Objektum katalógus

### 3.1.1 Játékprogram

A játék elindításáért felelős és annak bizonyos idő után történő leállításáért. Számontartja a játékosok számát az egyes csapatokban, valamint ő adja hozzá a megfelelő pontokat a játék során a csapatok pontszámához. Ezen keresztül lehet játékosokat hozzáadni a játékhoz.

### 3.1.3 Szerelő

A célja, hogy minél több vizet vigyen a ciszternákhoz. A csővégeket tudja egyik játékelemből a másikba állítani, valamint a pumpák irányát tudja változtatni, ezen felül az elromlott pumpákat tudja megjavítani. A szerelők építik játék közben a csőrendszert. Új pumpákat tudnak a játékban lehelyezni csövek közé. Új pumpákat a ciszternáktól tudnak venni. Egyszerre egy pumpa lehet egy szerelőnél. A játékelemeken tudnak közlekedni. A csöveken nem férnek el egymás, illetve a szabotőrök mellett csak a többi játékelemen. A pumpa kimenetének átállítása után várni kell egy kis időt, hogy újra át lehessen állítani. Egy elromlott pumpa vagy kilyukadt cső megjavítása időt vesz igénybe. Bemenetet nem lehet állítani, mindig belefolyik a beérkező cső vize.

### 3.1.4 Szabotőr

A célja, hogy minél több vizet jutasson a sivatagba. A csöveket tudja kilyukasztani. Egy cső kilyukasztása időt vesz igénybe és utána várni is kell még valamennyi időt, hogy újra lyukasztani lehessen. A pumpákat át tudja állítani, hogy víz irányát megváltoztassa. A szabotőrök a játék elemeken tudnak mozogni.

**3.1.5 Cső**

A vizet szállítja. Egy csövön, egy bizonyos mennyiségű víz folyik el egyszerre, ez a cső kapacitása. A szabadvégű csövek végén a víz elfolyik a homokba. Ha a csövön egy lyuk keletkezik, akkor a cső kimenetére már nem jut el a víz.

### 3.1.6 Pumpa

A pumpa a csövek elágaztatására szolgál a játék során. A csövekben lehet elágazásokat létrehozni vele. Csövek kapcsolódnak hozzá, de egyszerre csak egy be- és kimenete lehet. Van megadott méretű tartálya, amiben a bemeneti csőből érkező vizet először eltárolja, mielőtt továbbadja a kimeneti cső felé. A ki- és bemenetet a játékosok tudják állítani. Bizonyos időközönként el tud romlani, ilyenkor nem ad ki a kimenetén vizet.

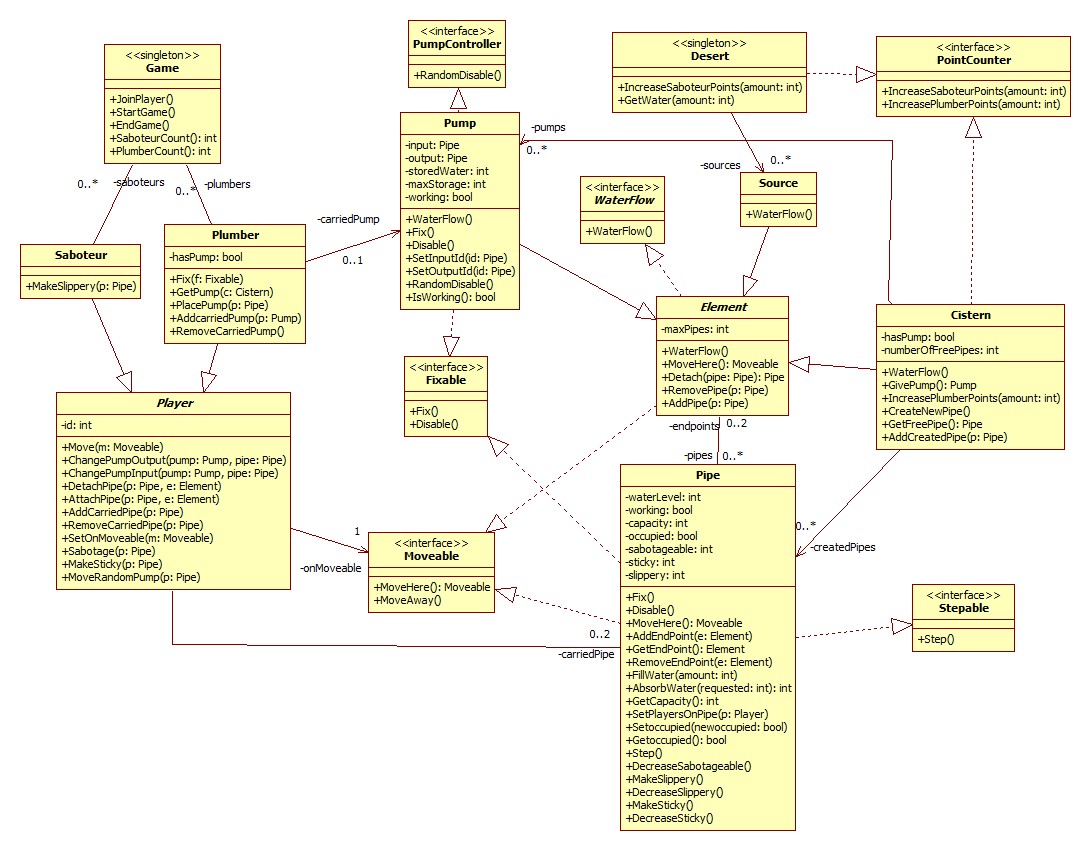
### 3.1.7 Ciszterna

A szerelők végső célpontja, ide kell a vizet eljuttatniuk, hogy pontot szerezzenek. Bármennyi vizet képes elnyelni. Innen tudják a szerelők felvenni a pumpákat. Csak bemenete van.

### 3.1.8 Forrás

A forrásból jön a végtelen mennyiségű víz, amiből a szerelők el tudják vezetni a vizet a ciszternákba. Ennek nincsen bemenetje, csak kimentje van. A forrásnak több kimenetje is lehet.

## 3.2 Statikus struktúra diagram



## 3.3 Osztályok leírása

### 3.3.1 Cistern

#### · Felelősség

A pálya végén helyezkedik el. Csak bemenetei lehetnek, amikből a játék során folyamatosan készülnek újak, szabad végű csövek. Az ide bemenő összes vizet elnyeli és ez alapján szereznek pontokat a szerelők. Innen tudnak a szerelők új pumpákat felvenni a csőrendszer bővítéséhez.

#### · Ősosztályok

Element

#### · Interfészek

PointCounter

#### · Attribútumok

* **bool hasPump:** Megadja, hogy a ciszternában van a felvehető pumpa.
* **int numberOfFreePipes:** Hány szabad végű cső van még a ciszternában.

#### · Metódusok

* **void WaterFlow()**: Elnyeli az összes bele menő vizet és ez alapján növeli a szerelők pontjait.
* **void Pump GivePump():** A szerelők a segítségével egy pumpát tudnak magukhoz venni. Ennek azonosítója eltárolódik a Szerelő objektumban.
* **void IncreasePlumberPoints(amount: int)**: Megnöveli a szerelők pontszámát a paraméterben megadott egész számmal.
* **void CreateNewPipe():** Egy új cső jön létre, ami az egyik vége a ciszternához van csatlakozva.
* **Pipe GetFreePipe():** Szabadon lévő cső véget adja vissza

### 3.3.2 Desert

#### · Felelősség

A pályát reprezentálja. A forrásokat tárolja, amiből az egész pályát fel lehet építeni. Ide folyik el a víz a kilyukadt csövekből. Az elfolyt víz alapján ad pontokat a szabotőröknek.

#### · Ősosztályok

-

#### · Interfészek

PointCounter

#### · Attribútumok

* **-**

#### · Metódusok

* **GetWater(amount: int):** Megkapja az elfolyt vizet a csövekből.
* **void IncreaseSaboteurPoints(amount: int)**: Megnöveli a szabotőrök pontszámát a paraméterben megadott egész számmal.

### 3.3.3 Element

#### · Felelősség

A játékban szereplő és azokkal interaktálható elemek. Ezekből épül fel a csőrendszer. Ezeken tudnak a játékosok mozogni és tudnak interaktálni velük. Ezenfelül a játék két fő csomópontja a ciszterna, valamint a forrás is ebből származik le. A víz szállítására, valamint a víz forrásaként/céljaként szolgálnak a játék során.

#### · Ősosztályok

**-**

#### · Interfészek

WaterFlow, Moveable

#### · Attribútumok

* **int maxPipes:** Az adott elemhez csatlakozó csövek számát adja vissza.

#### · Metódusok

* ***void WaterFlow()****:* Absztrakt függvény, amelyet a leszármazott játékelemek definiálnak. Az idő múlását szimuláljuk ezzel a függvénnyel. Az egyes elemek esetében más-más a hatása.
* **bool CanMoveHere():** Megnézi, hogy lehet-e ide mozogni.
* **bool CanDetach():** Megnézi, hogy lecsatolhato-e a cső.
* **RemovePipe(p: Pipe):** Elválasztja a csövet a játékelemről a paraméterben megadott csövet.
* **AddPipe(p: Pipe):** Hozzá adja a paraméterben megadott csövet a játékelemre.

### 3.3.4 Fixable

#### · Felelősség

Egy interface. Megvalósítja az elromolható/javítható funkciókat.

#### · Ősosztályok

**-**

#### · Interfészek

**-**

#### · Attribútumok

* **-**

#### · Metódusok

* ***Fix():*** megjavítja az elromlott elemet .
* **Disable():** Elrontja az elemet.
* **bool IsWorking():** Megmondja, hogy működik-e az elem.

### 3.3.5 Game

#### · Felelősség

Játékosokat ad hozzá a játékhoz és tárol. Elindítja a játékot. Minimum 4 ember kell a játék elindításához. Leállítja a játékot egy bizonyos idő után. Számon tartja a csapatok pontszámait, eldönti ki nyert a játék végén.Méri a játékidőt.

#### · Ősosztályok

-

#### · Interfészek

-

#### · Attribútumok

* **-**

#### · Metódusok

* **void JoinPlayer()**: Játékost ad hozzá az egyik csapathoz.
* **void StartGame()**: Elindítja a játékot. (csak akkor , ha elegendő játékos van)
* **void EngGame():** Leállítja a játékot. Kiírja melyik csapat nyert.
* **int SaboteurCount():** Visszaadja, hogy mennyi szabotőr van a játékban.
* **int PlumberCount():** Visszaadja, hogy mennyi szerelő van a játékban.

### 3.3.6 Moveable

#### · Felelősség

Interface. Azok a játékelemek, amikre a játékosok rá tudnak lépni.

#### · Ősosztályok

-

#### · Interfészek

-

#### · Attribútumok

* **-**

#### · Metódusok

* **bool CanMoveHere():** megkérdezi, hogy tudunk-e mozogni a játékelemre.

### 3.3.7 Pipe

#### · Felelősség

A vizet szállítja. Egy adott csövön, egy bizonyos mennyiségű víz folyik el egyszerre, ez a cső kapacitása. A szabadvégű csövek végén a víz elfolyik a homokba. Ha a csövön egy lyuk keletkezik, akkor a cső kimenetére már nem jut el a víz, a betáplált víz elfolyik a sivatagba.

#### · Ősosztályok

**-**

#### · Interfészek

Moveable, Fixable

#### · Attribútumok

* **int waterLevel:** Az aktuális vízmennyiséget mutatja a csőben
* **int capacity:** Az adott cső kapacitása. (maximum vízmennyiség, amit a cső vinni tud)
* **bool working:** Megadja, hogy a cső működik-e vagy nem.
* **int sticky**: Mennyi ideig ragad még a cső. Addig érvényes, amíg nagyobb mint 0.
* **int slippery**: Mennyi ideig csúszós még a cső. Addig érvényes amíg nagyobb mint 0.
* **int sabotageable**: Mennyi ideig nem lehet még kilyukasztani a csövet. Addig érvényes, amíg nagyobb mint 0.

#### · Metódusok

* **void Fix:** Az adott cső megjavítása. (working változó igazra állítása)
* **void Disable():** Az adott cső kilyukasztása. (working változó hamisra állítása)
* **bool CanMoveHere():** Megkérdezi, hogy el tud-e mozogni a játékelemre ahova akar menni.
* **void AddEndPoint(Element e):** A cső végére rácsatlakoztatja a paraméterben megadott játékelemet.
* **Element GetEndPoint():** Visszaadja a cső végében csatlakoztatott játékelemet.
* **void RemoveEndPoint(Element e):** A cső végén lecsatlakoztatja a paraméterben megadott játékelemet.
* **void FillWater(amount: int):** Megtölti a csövet az adott vízmennyiséggel
* **int AbsorbWater(requested: int):** Kiüríti a csövet az adot vízmennyiséggel
* **int GetCapacity():** Visszaadja a capacity változó értékét
* **bool IsWorking():** Visszaadja a working változó értékét.
* **Step()**: Minden a játékban eltelt idő után csökkenti a ragadás, csúszás és lyukaszhatóságot.
* **MakeSlippery()**: a slippery változó értékét állítja egy 0-nál nagyobb értékre
* **DecreaseSlippery()**: a slippery változó értékét csökkenti 1-el(ha > 0)
* **MakeSticky()**: a sticky változó értékét állítja egy 0-nál nagyobb értékre
* **DecreaseSticky()**: a sticky változó értékét csökkenti 1-el(ha > 0)
* **MakeSlippery()**: a slippery változó értékét állítja egy 0-nál nagyobb értékre
* **DecreaseSabotageable()**: a sabotageable változó értékét csökkenti 1-el(ha > 0)

**3.3.8** **Player**

#### · Felelősség

A játékosok tevékenységeit gyűjti össze általánosságban. Belőle származnak le a két csapat tagjai (Saboteur, Plumber). Külön azonosítóval rendelkezik minden játékos a megkülönböztetésük érdekében. Egy adott játékosnál egyszerre maximum 1 csővég lehet.

#### · Ősosztályok

**-**

#### · Interfészek

**-**

#### · Attribútumok

* **int id:** A játékos egyedi azonosítója.

#### · Metódusok

* **void Move(Moveable m)**: Arra szolgál, hogy a játékos egy másik Moveable játékelemre tudjon mozogni.
* **void ChangePumpOutput(p: Pump, pipe: Pipe):** Átállítja a p pumpa kimeneti csövét, egy másik csőre az id int változó segítségével.
* **void ChangePumpInput(p: Pump, pipe: Pipe):** Átállítja a p pumpa bemeneti csövét, egy másik csőre az id int változó segítségével.
* **void DetachPipe(p: Pipe, e: Element):** Leválasztja a kiválasztott csövet a paraméterben megadott elemnél.
* **void AttachPipe(p: Pipe, e: Element):** Leteszi a kiválasztott csövet a paraméterben megadott elemnél.
* **void AddCarriedPipe(p: Pipe):** Felveszi a paraméterben megadott csövet a kezébe.
* **void RemoveCarriedPipe(p: Pipe):** Leteszi a paraméterben megadott csövet a kezéből

**3.3.9** **Plumber**

#### · Felelősség

Csöveket/pumpákat tudja megjavítani. Meglévő játékelemeket tudnak a ciszternáknál készülő új csövekkel összekötni. Új pumpákat tudnak a játékban lehelyezni egy adott csőszakaszra, valamint ezek ki- és bemenetét állítani (egyszerre csak egy ki- és bemenete lehet). Új pumpákat a ciszternáktól tudnak venni, Egyszerre egy pumpa lehet egy szerelőnél. A játékelemeken tudnak közlekedni. A csöveken nem férnek el egymás, illetve a szabotőrök mellett csak a többi játékelemen. A pumpa bemenetének/kimenetének átállítása után várni kell egy kis időt, hogy újra át lehessen állítani. Egy elromlott pumpa vagy kilyukadt cső megjavítása időt vesz igénybe.

#### · Ősosztályok

Player

#### · Interfészek

**-**

#### · Attribútumok

* **bool hasPump:** Van-e az adott szerelőnél pumpa.

#### · Metódusok

* **void Fix(f: fixable ):** Egy javítható játékelem megjavítása (befoltozása).
* **void GetPump(c: Cistern):** Pumpa felvétele a paraméterben adott ciszternából, abban az esetben ha még nem rendelkezik pumpával.
* **void PlacePump(p: Pipe):** Pumpa lerakása a paraméterben megadott csőszakaszra, abban az esetben ha rendelkezik egy pumpával a szerelő.
* **void** **canMove()**: bool: tud-e mozogni a játékos.
* **Sabotage(p: Pipe)**: A szabotőrtől került át ez a metódus. Kilyukasztja a megadott csövet.
* **MoveRandomPump(p:Pipe)**: a paraméterként kapott cső véletlenszerű végpontjára lép

**3.3.10** **PointerCounter**

#### · Felelősség

Egy interface. A játék pontszerzési metódusait valósítja meg.

#### · Ősosztályok

**-**

#### · Interfészek

**-**

#### · Attribútumok

* **-**

#### · Metódusok

* **IncreaseSaboteurPoints(amount: int):** A paraméterben pontott hozzáadja szabotőröknek.
* **IncreasePlumberPoints(amount: int):** A paraméterben pontott hozzáadja szerelőknek.

### 3.3.11 Pump

#### · Felelősség

A csövekben lehet elágazásokat létrehozni vele. Sok cső kapcsolódhat hozzá, de egyszerre csak egy be- és kimenete lehet. Van tartálya, amiben a bemeneti csőből érkező vizet először eltárolja, mielőtt továbbadja a kimeneti cső felé. A ki- és bemenetet a játékosok tudják állítani. Bizonyos időközönként el tud romlani, ilyenkor nem ad ki a kimenetén vizet.

#### · Ősosztályok

Element

#### · Interfészek

**-**

#### · Attribútumok

* **Pipe outputId:** A cső amely felé továbbítja a vizet.
* **Pipe inputId:** A cső amely felől kapja a vizet.
* **bool working:**  Megmondja, hogy tud-e vizet továbbítani a pumpa.
* **int storedWater:** A tárolójában lévő víz mennyisége.
* **int maxStorage:** A maximum vízmennyiség, amit a pumpa el tud tárolni.

#### · Metódusok

* **void WaterFlow()**: Vizet pumpál a kimeneti csőbe, ha nincs elromolva (annyit amennyit tud).
* **void Fix():** Megjavítja a hibás pumpát.
* **void Disable():** Elrontja a pumpát.
* **SetOutputId(int id)**: Beállítja a pumpa kimenetét, a paraméterben megadott ID-vel rendelkező cső felé.
* **SetInputId(int id)**: Beállítja a pumpa bemenetét, a paraméterben megadott ID-vel rendelkező cső felé.
* **void RandomDisable():** Változó időközönként elrontja a pumpát.
* **bool IsWorking():** Visszaadja a working változó értékét.

### 3.3.12 PumpController

#### · Felelősség

Interface.Véletlenszerűen elront egy véletlenszerű pumpát.

#### · Ősosztályok

**-**

#### · Interfészek

**-**

#### · Attribútumok

* **-**

#### · Metódusok

* **void RandomDisable():** Elront egy véletlenszerű pumpát, a pumpa függvény working változóját állítja hamisra, ekkor a pumpa nem továbbít vizet.

### 3.3.13 Saboteur

#### · Felelősség

A csöveket tudja kilyukasztani. Egy cső kilyukasztása időt vesz igénybe.. A pumpákat át tudja állítani, hogy víz irányát megváltoztassa. A szabotőrök a játék elemeken tudnak mozogni. Viszont a csövön nem tudja megkerülni a másik játékost.

#### · Ősosztályok

Player

#### · Interfészek

**-**

#### · Attribútumok

* **-**

#### · Metódusok

* **void Sabtogage():** kilyukasztja a paraméterben megadott csövet.
* **MakeSlippery(p: Pipe)**: A megadott csövet bizonyos ideig csúszóssá teszi.

### 3.3.14 Source

#### · Felelősség

A forrásból jön a végtelen mennyiségű víz, amiből a szerelők el tudják vezetni a vizet a ciszternákba. Ennek nincsen bemenetje, csak kimentje van. A forrásnak több kimenetje is lehet a játék folyamán.

#### · Ősosztályok

Element

#### · Interfészek

#### · Attribútumok

* **-**

#### · Metódusok

* **void WaterFlow():** Vizet visz az összes kimenti csövekre, amennyit az adott kapacitás enged.

### 3.3.15. WaterFlow

#### · Felelősség

Interface. A víz folyását valósítja meg .

#### · Ősosztályok

-

#### · Interfészek

#### · Attribútumok

**-**

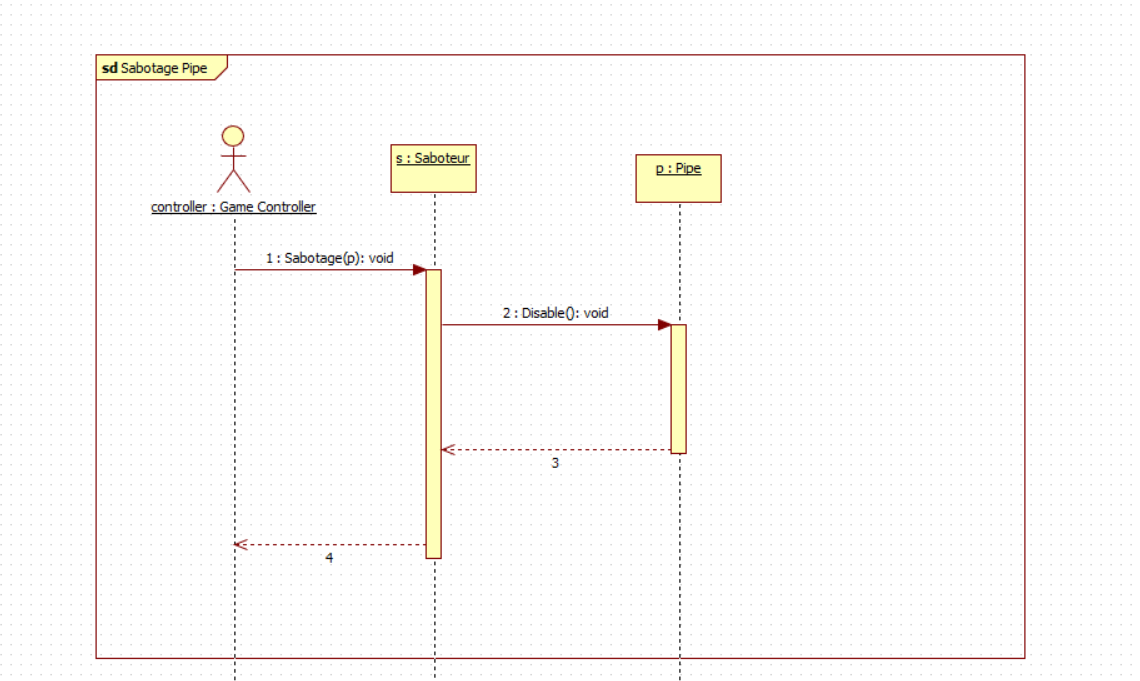
#### · Metódusok

* **void WaterFlow():** Víz folyását kezeli.

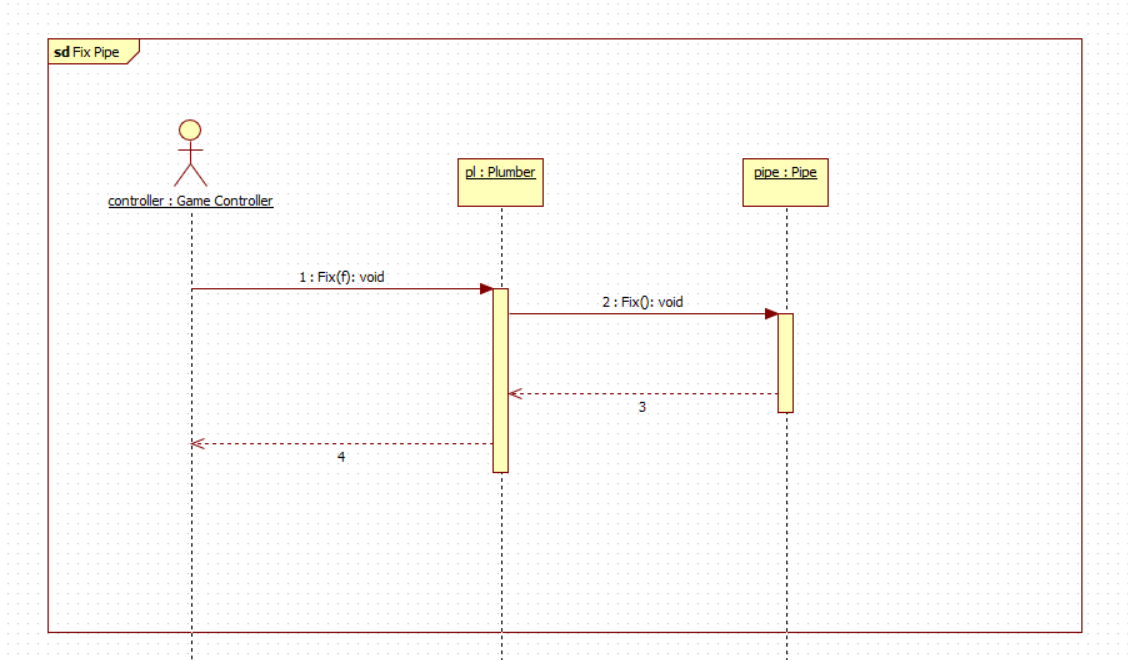
.

## 3.4 Szekvencia diagramok

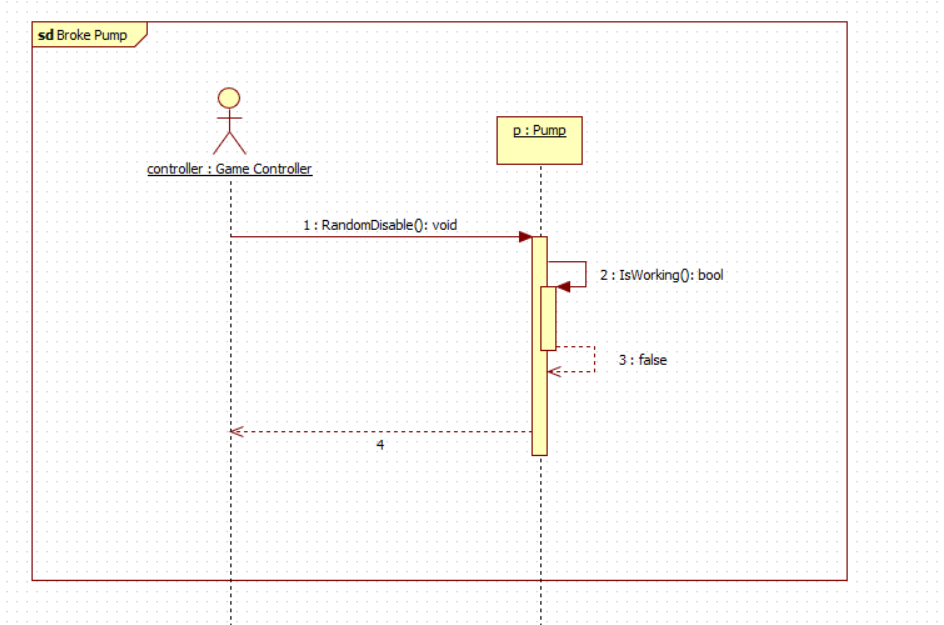
**3.4.1 Sabotage Pipe**



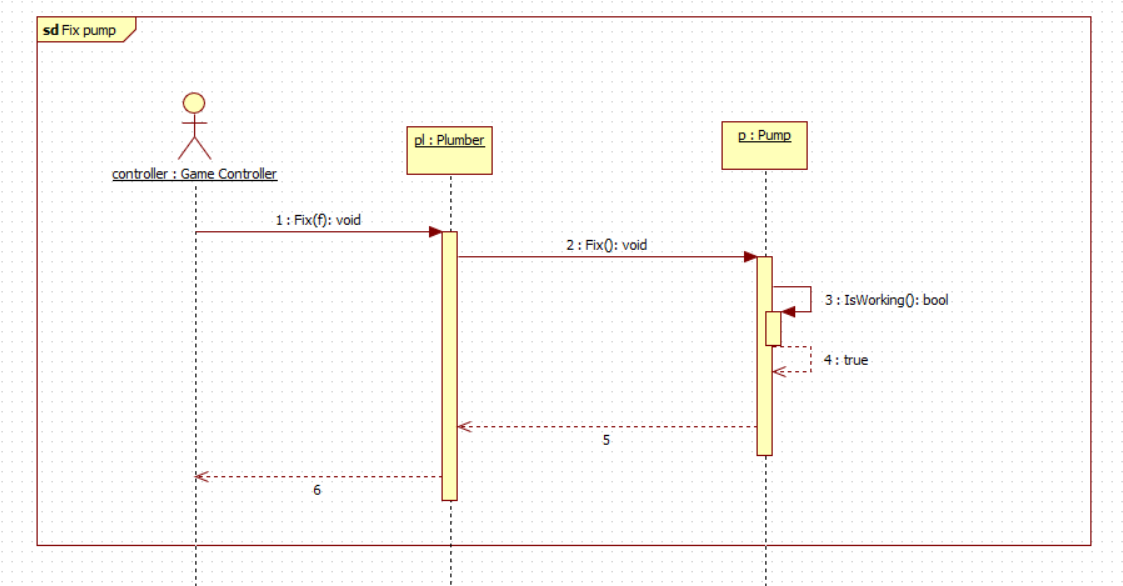
**3.4.2. Fix Pipe**



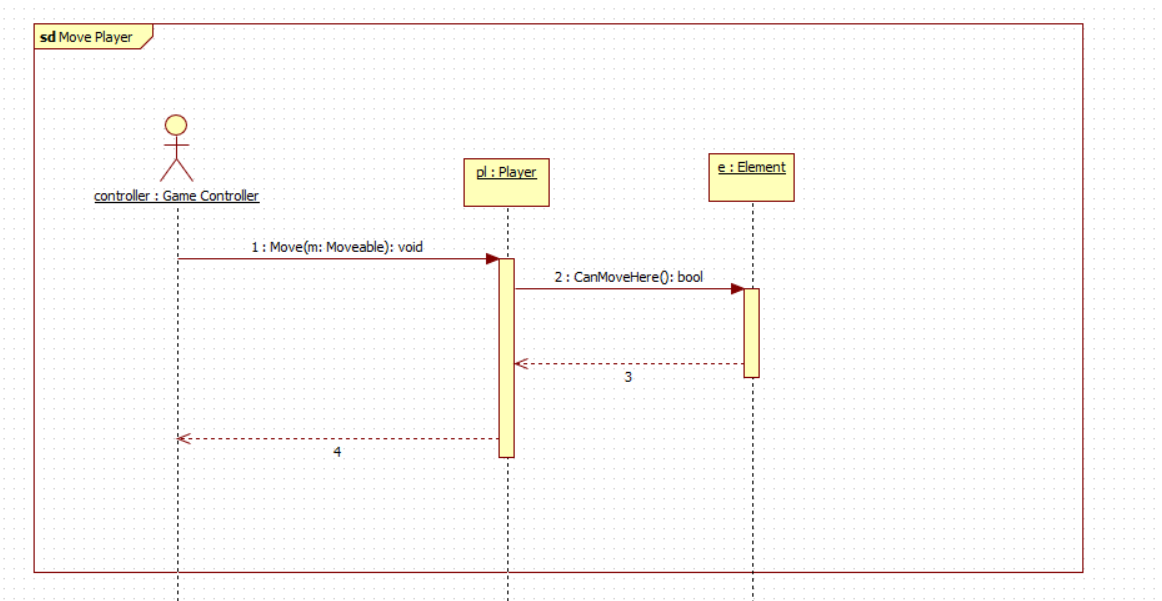
**3.4.3 Broke Pump**

****

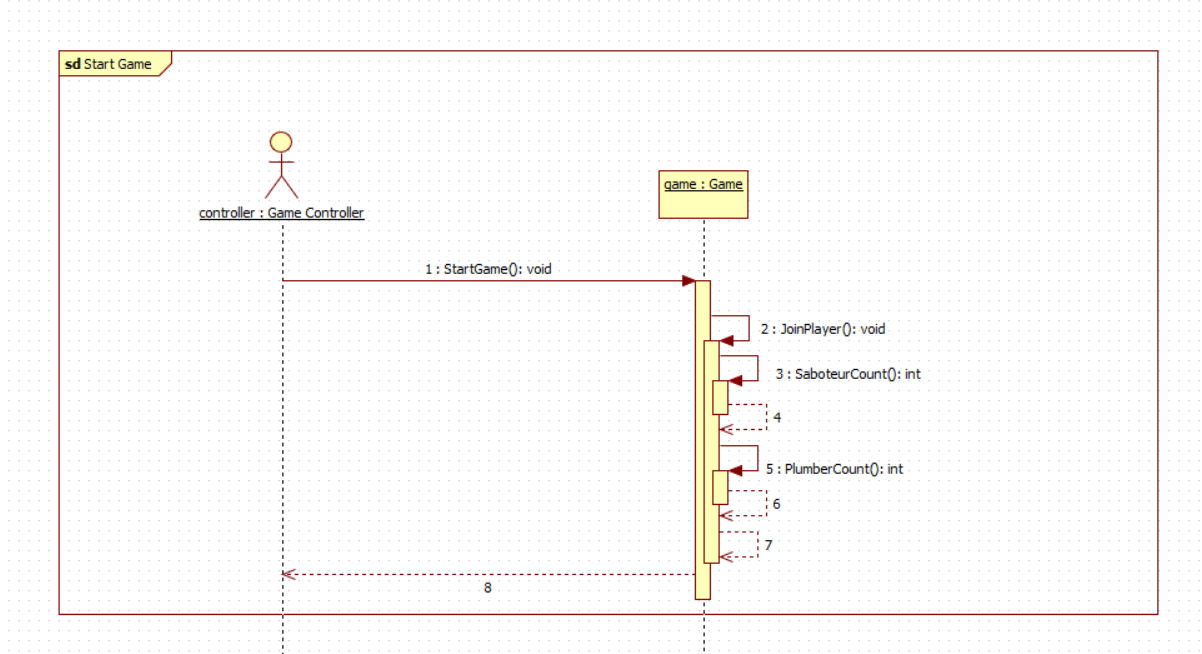
**3.4.4 Fix pump**

****

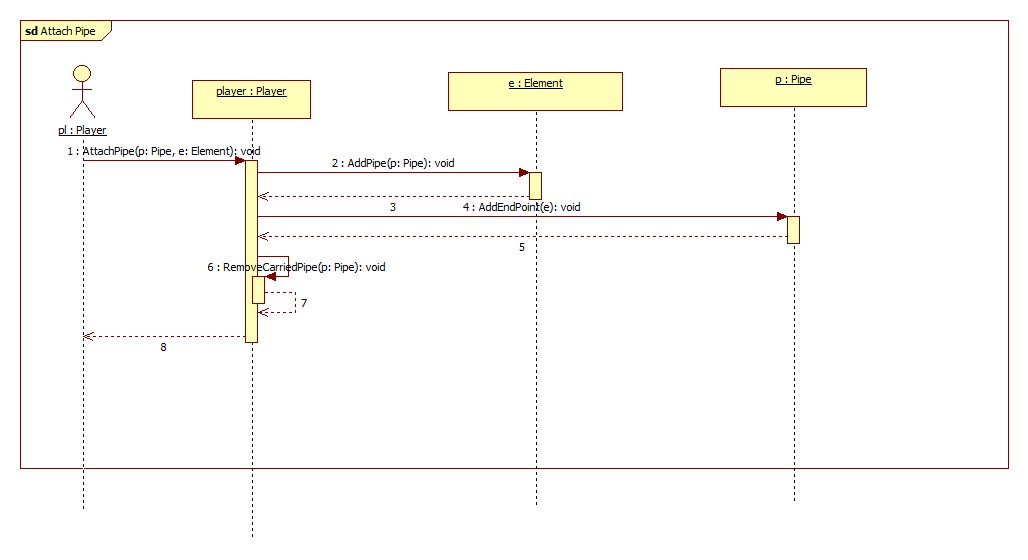
**3.4.5 Move Player**

****

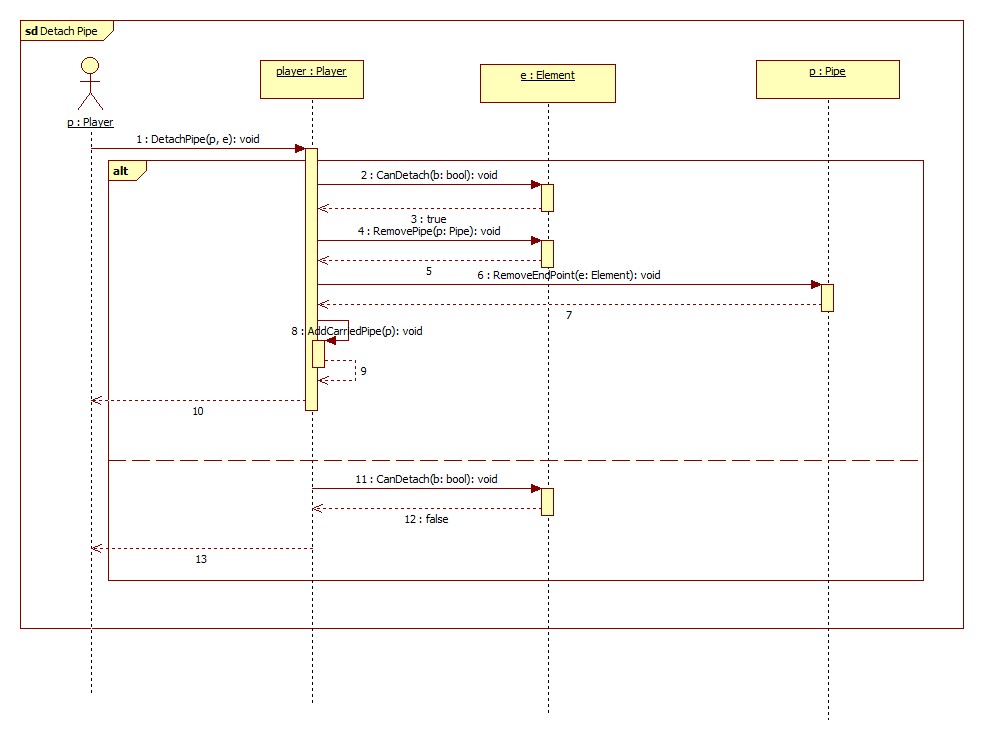
**3.4.6 Start Game**

****

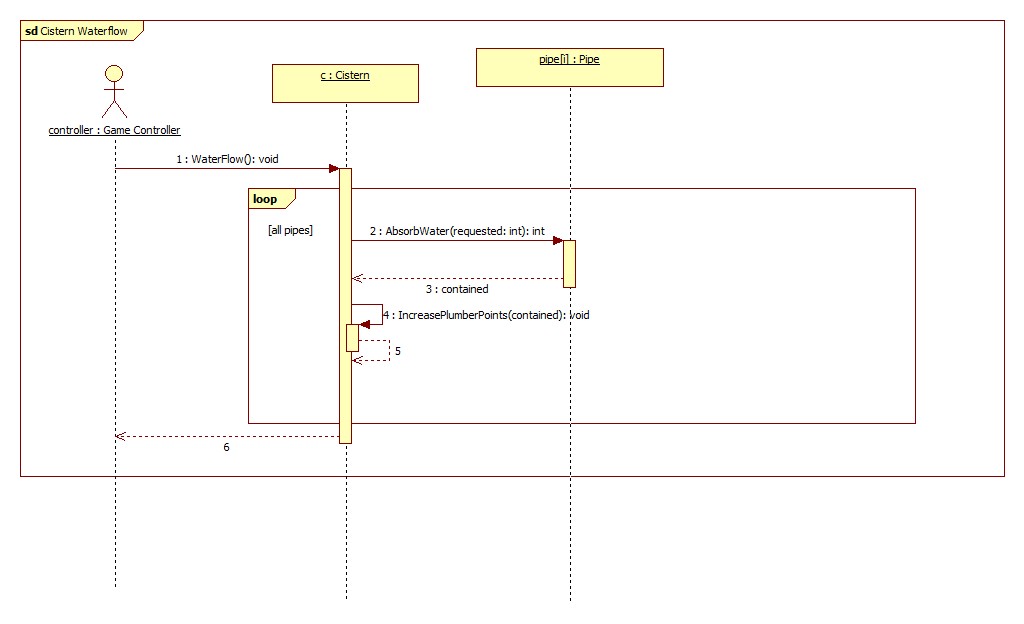
**3.4.7 Attach Pipe**

****

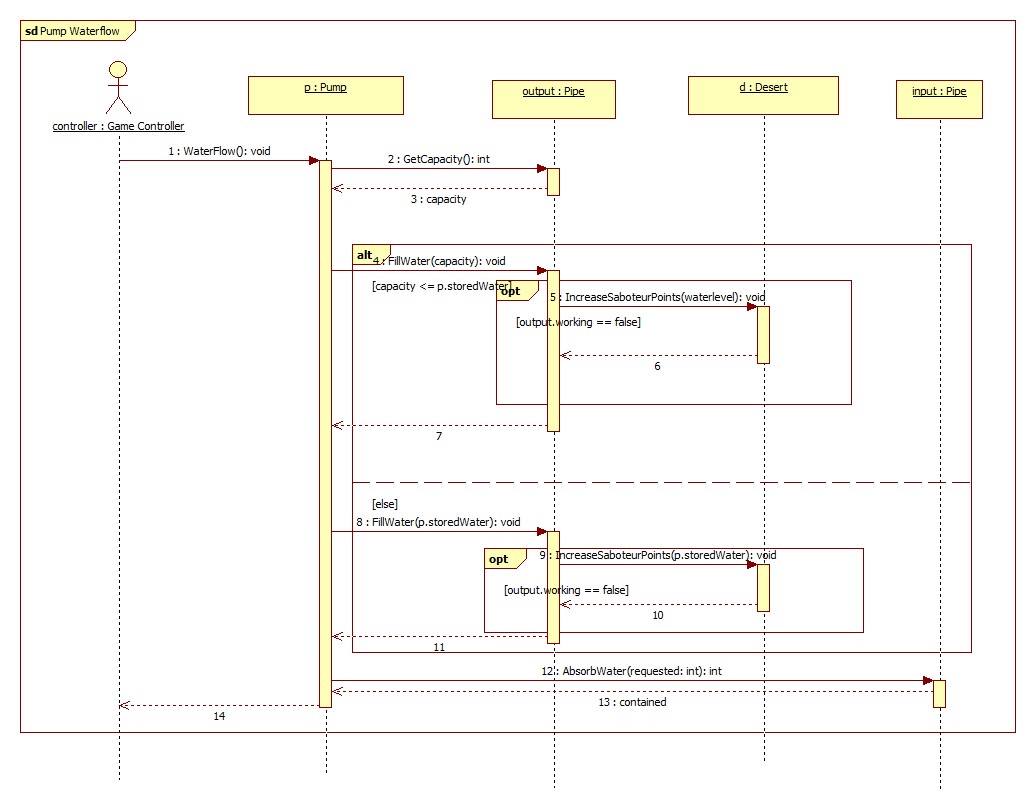
**3.4.8 Detach Pipe**

****

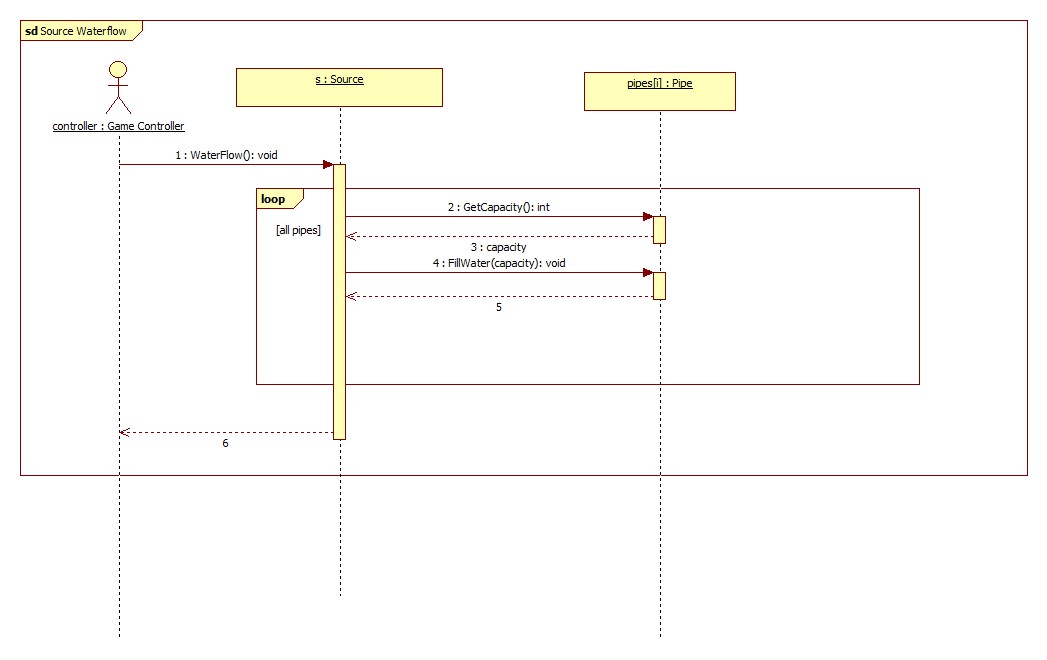
**3.4.9 Cistern Waterflow**

****

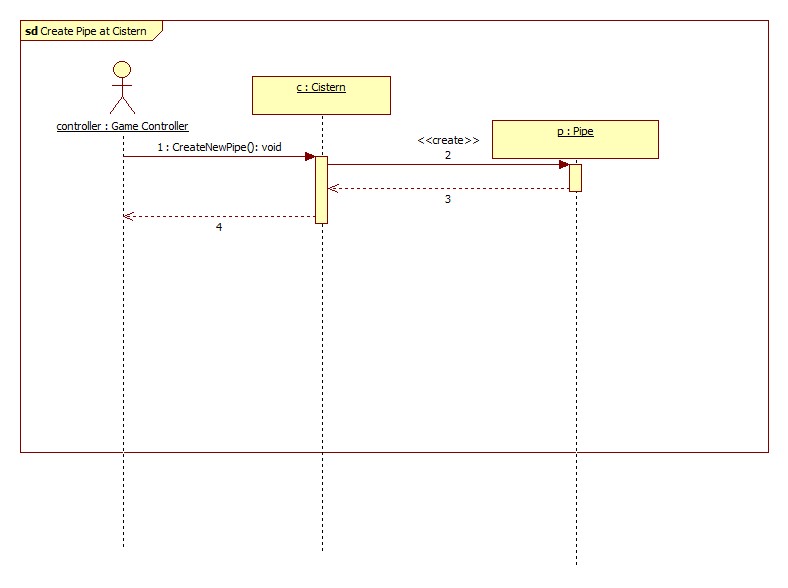
**3.4.10 Pump Waterflow**

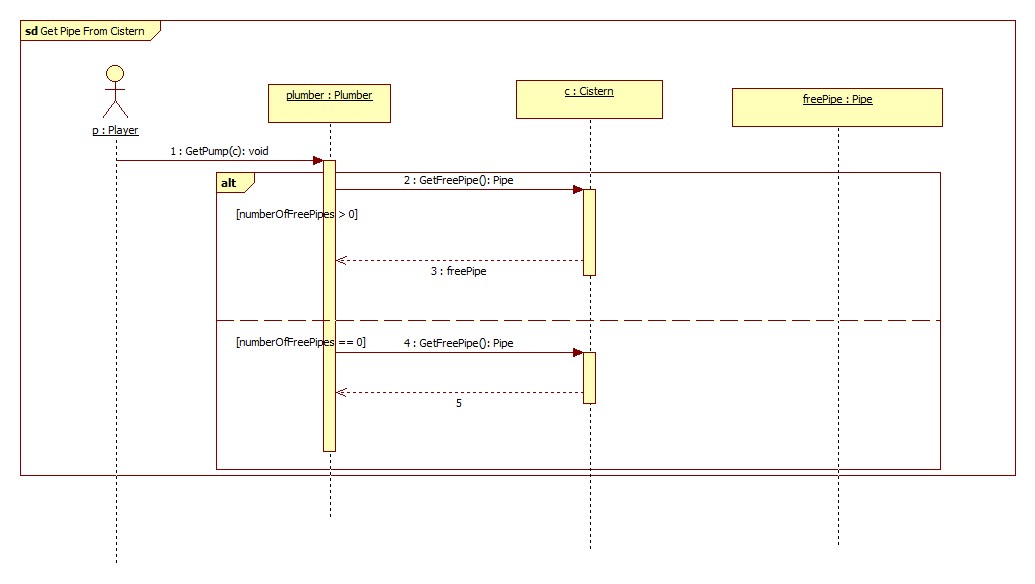
****

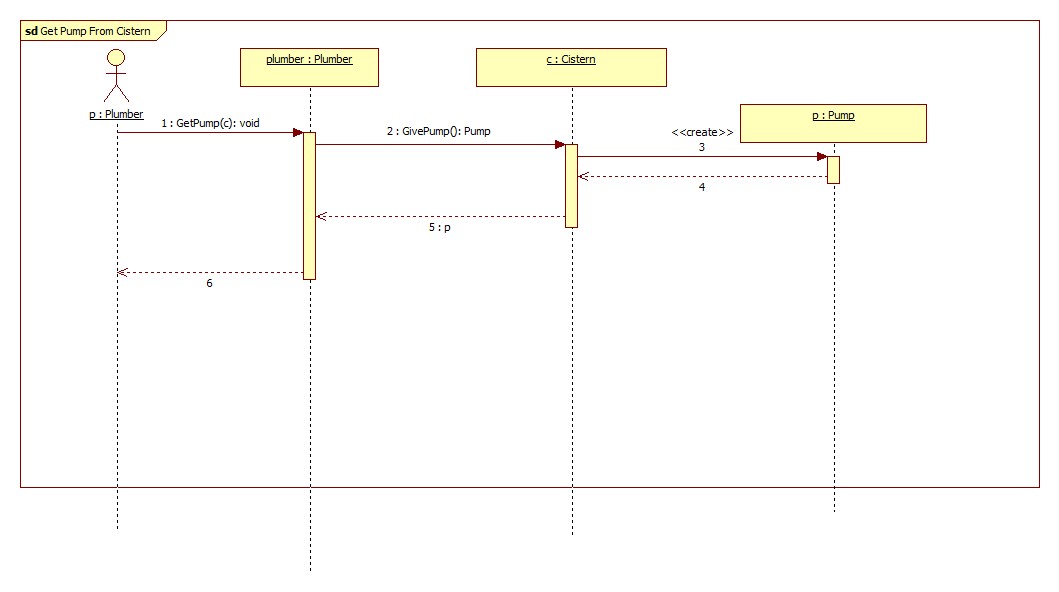
**3.4.11 Source Waterflow**

****

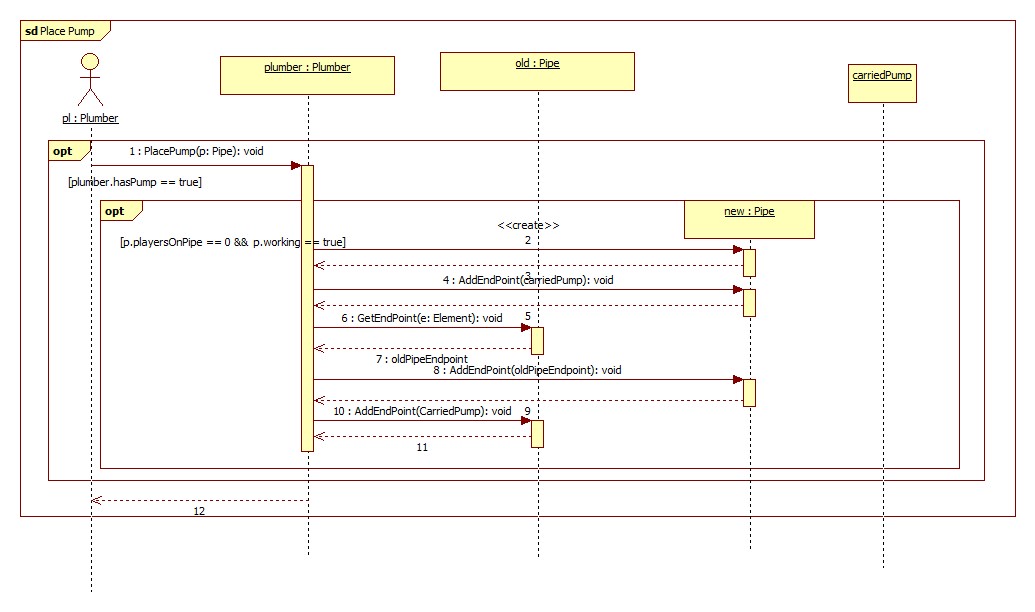
**3.4.12 Create Pipe at Cistern**

****

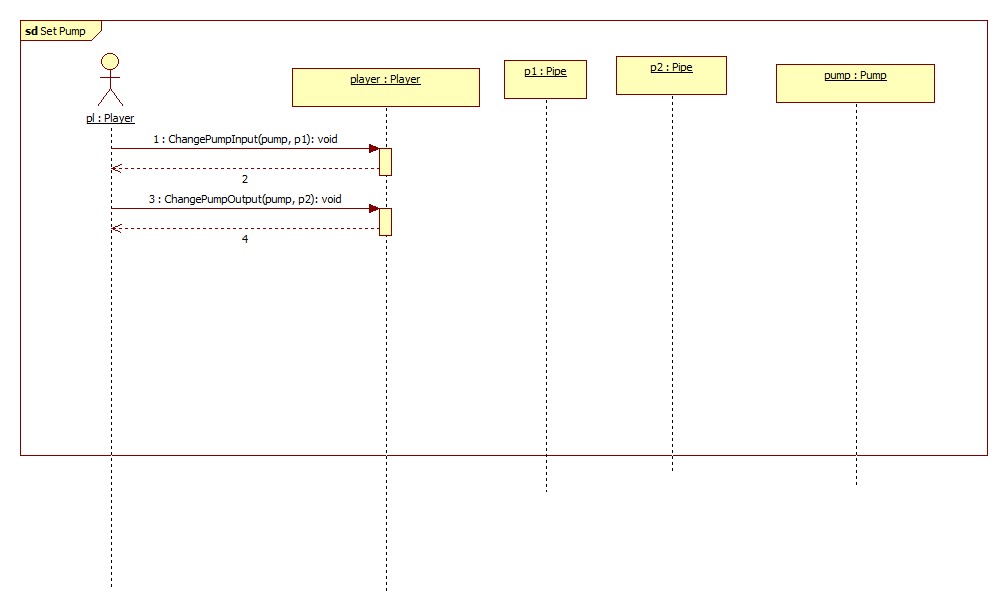
**3.4.13 Get Pipe from Cistern**

**3.4.14 Get Pump from Cistern**

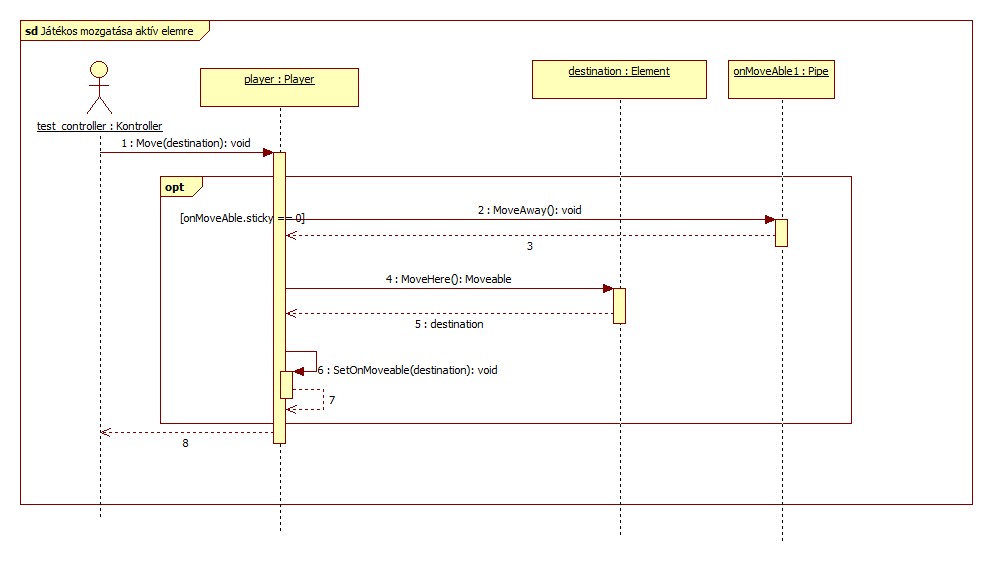
**3.4.15 Place Pump**

****

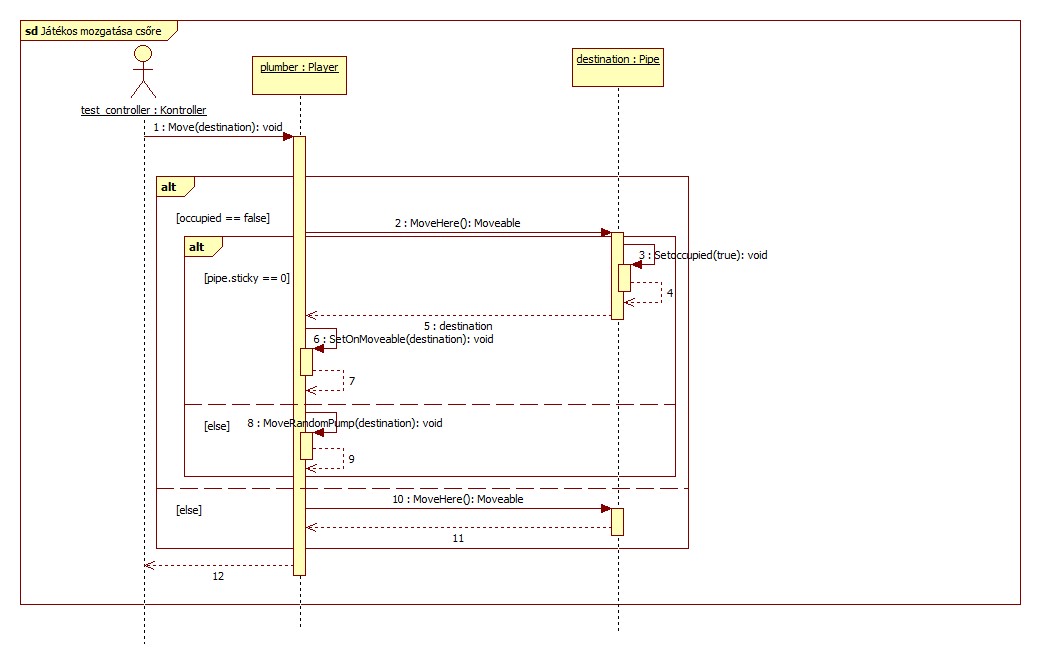
**3.4.16 Set Pump**

****

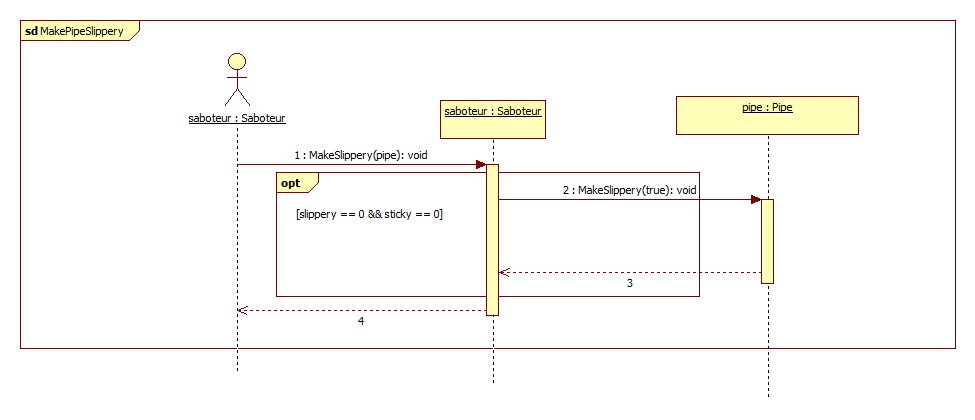
**3.4.17** **Játékos mozgatása aktív elemre**



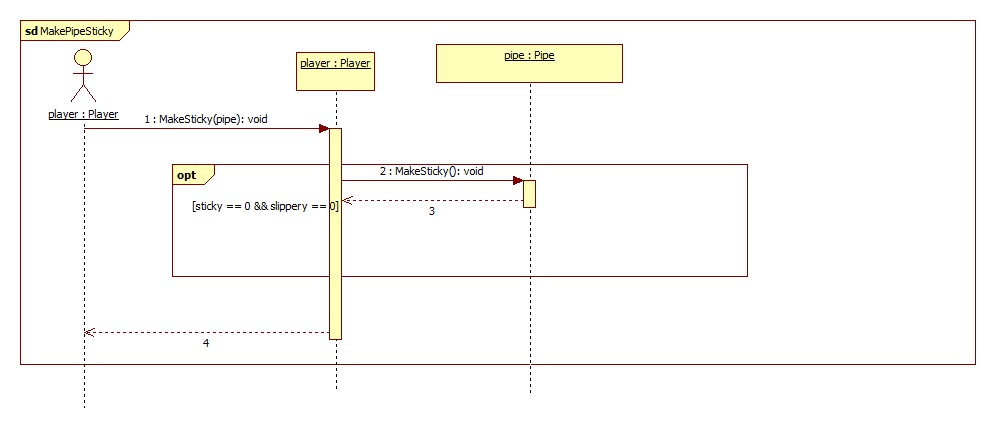
**3.4.18** **Játékos mozgatása csőre**



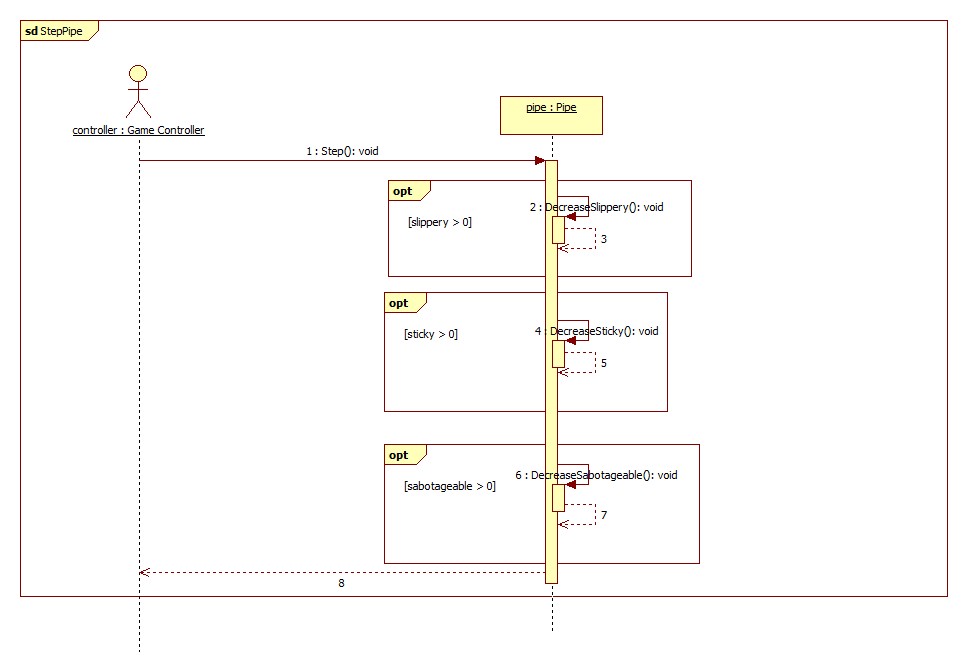
**3.4.19** **Cső csúszóssá tétele**



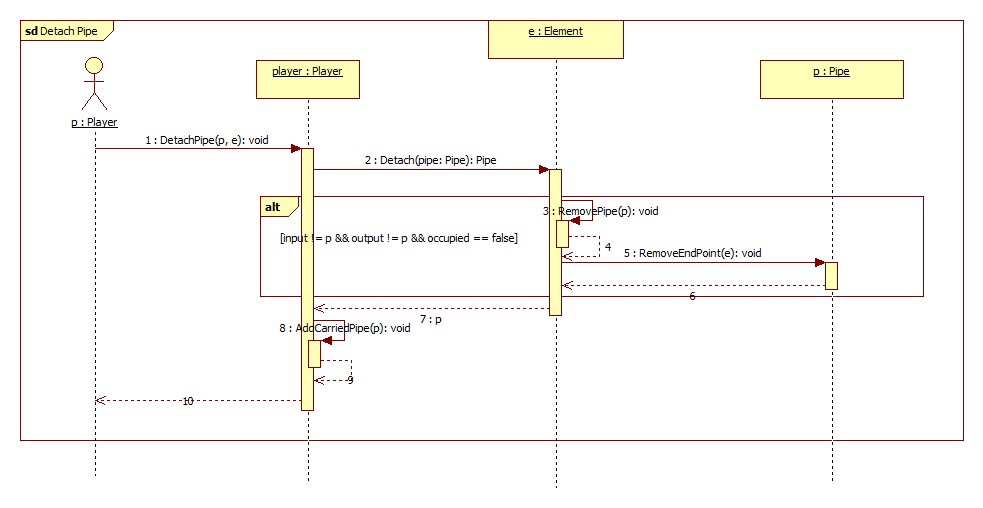
**3.4.20** **Cső ragadóssá tétele**



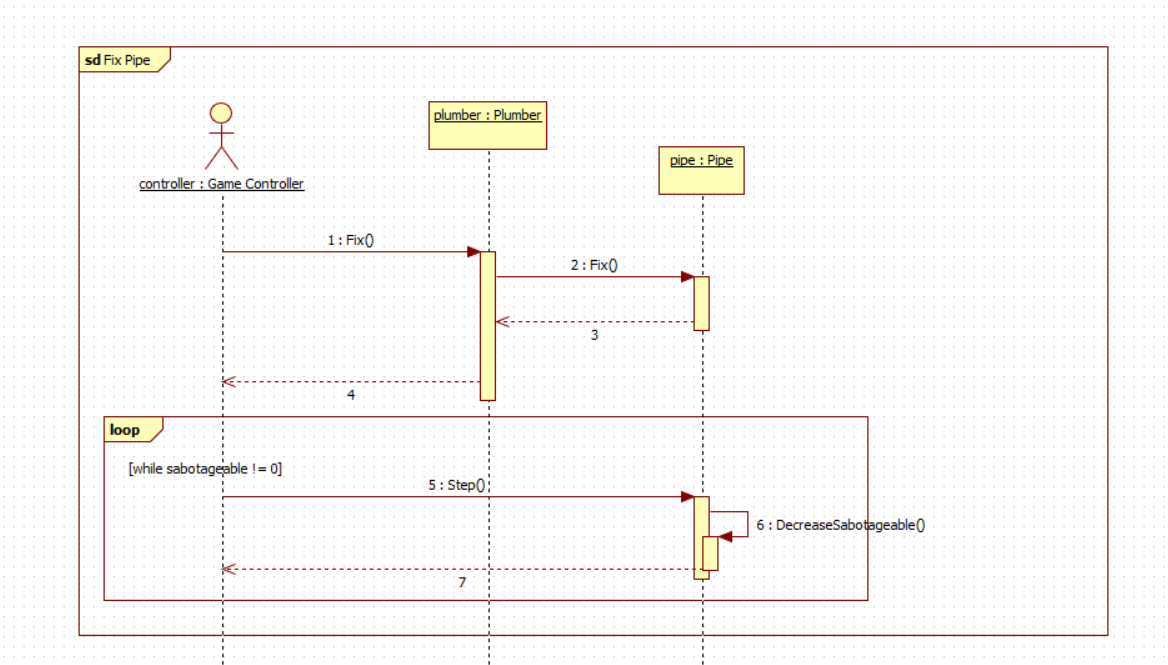
**3.4.21** **Cső timer step**



**3.4.22** **Cső lecsatlakoztatása**



**3.4.23** **Cső megjavítása**



## 3.5 State-chartok

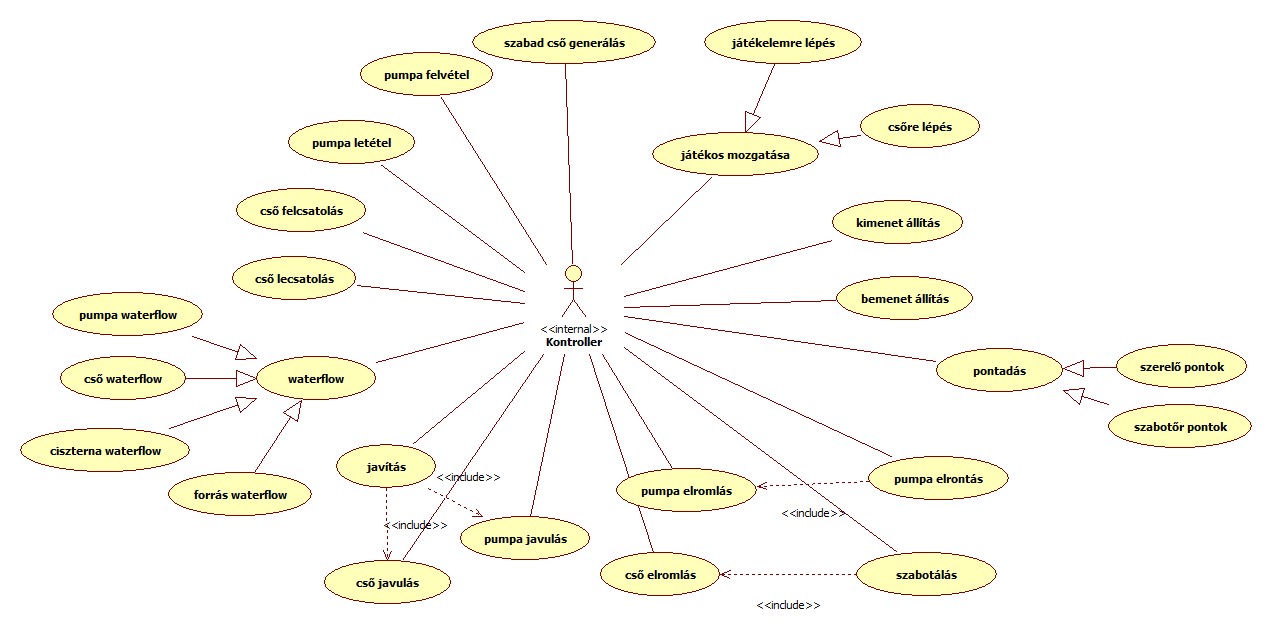
## 3.6 Napló

| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| --- | --- | --- | --- |
| 2023..03.15. 16:00 | 4 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet.  Döntés: Chu és Bognár elkészíti az objektumkatalógust csütörtök (03.16.) reggel 10:00 óráig. Az objektumkatalógus elkészülte után Gerencsér elkészíti az osztálydiagramot csütörtök (03.16.) este 20:00 óráig. Sütheö és Vörös megcsinálja az osztályok, valamint azok publikus metódusainak leírását péntek (03.17.) reggel 10:00 óráig. |
| 2023.03.15. 18:00 | 1,5 óra | Chu | Tevékenység:  Chu elkészítette a Játékprogram, Játékos, Szerelő, Szabótőr leírásokat az objektumkatalógusból. (3.1.1-3.1.4) |
| 2023.03.15. 20:30 | 2 óra | Bognár | Tevékenység: Bognár elkészítette a Cső, Pumpa, Ciszterna, Forrás, Játékelem leírásokat az objektumkatalógusból. (3.1.5-3.1.9) |
| 2023.03.16. 16:00 | 5 óra | Gerencsér | Tevékenység: Gerencsér elkészíti az osztálydiagramot. (3.2) |
| 2023.03.16. 21:30 | 2 óra | Sütheö | Tevékenység: Sütheö elkészíti a Cistern, Element, Game, Pipe, Player osztályok leírását. (3.3.1-3.3.5) |
| 2023.03.16. 20:45 | 2 óra | Vörös | Tevékenység: Vörös elkészíti a Plumber, Pump, Saboteur, Source osztályok leírását. (3.3.6-3.3.9) |
| 2023.03.17. 8:30 | 1 óra | Chu | Tevékenység: Chu kijavítja az eddig elkészített részekben a helyesírási hibákat. |
| 2023.03.17. 11:30 | 1,5 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet.  Döntés: Bognár és Vörös elkészíti a szekvenciadiagramokat szombat (03.18.) reggel 10:00 óráig. Sütheö elkészíti az state-chartokat szombat (03.18.) reggel 10:00 óráig. |
| 2023.03.17. 13:40 | 2 óra | Bognár | Tevékenység: Bognár elkészített 7 darab szekvenciadiagramot. (3.4.1-3.4.7) |
| 2023.03.17. 17:00 | 3 óra | Vörös | Tevékenység: Vörös elkészített 10 darab szekvenciadiagramot. (3.4.8-3.4.17) |
| 2023.03.17. 18:30 | 2 óra | Sütheö | Tevékenység: Sütheö elkészítette a sate-chartokat. (3.5.1-3.5.2) |
| 2023.03.18. 11:00 | 3 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet.  Döntés: Gerencsér kijavítja az osztálydiagramon az elírásokat, valamint a hibás láthatóságú attribútumokat vasárnap reggel 10:00-ig. |
| 2023.03.18. 14:00 | 1 óra | Gerencsér | Tevékenység: Gerencsér kijavítja az osztálydiagramon a 3 darab elírást, valamint két függvény láthatóságát átállítja (GetPump, FixPumpAction függvények). |
| 2023.03.19. 16:00 | 2 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet, a dokumentum teljes körű átvizsgálása.  Döntés: Chu átvizsgálja a dokumentumot elírásokat keresve. |
| 2023.03.19. 18:15 | 0,5 óra | Chu | Tevékenység: Chu átnézte a dokumentumot, elírást nem talált benne. |
| 2023.03.22. 17:00 | 3 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet, a javítandó feladatok kiosztása.  Döntés: Chu kiveszi az absztrakciós jellegű objektumokat az objektum katalógusból ma (03.22.) 22:00 óráig. Chu és Bognár átdolgozza az osztálydiagramot péntek (03.24.) 10:00 óráig. Gerencsér és Vörös átdolgozza a javításra szoruló szekvenciadiagramokat, majd felveszik azokat amelyek hiányoztak péntek (03.24.) 22:00 óráig. Sütheö átdolgozza a state-chartokat péntek (03.24.) 22:00 óráig |
| 2023.03.22. 20:30 | 1 óra | Chu | Tevékenység:  Chu törli a Játékprogram,Játékos, Játékelem objektumokat az objektumkatalógusból.  (3.1.) |
| 2023.03.23. 17:00 | 4 óra | Bognár | Tevékenység:  Bognár felveszi a PointCounter, WaterFlow interfészeket, valamit a Desert osztályt. Az ezekhez kapcsolódó metódusokat, kapcsolatokat, leszármaztatásokat felveszi.  (3.2.) |
| 2023.03.23. 22:15 | 3 óra | Chu | Tevékenység:  Chu felveszi a Fixable és Moveable interfészeket. Az ezekhez kapcsolódó metódusokat, kapcsolatokat felveszi a megfelelő osztályok között.  (3.2) |
| 2023.03.24. 12:00 | 3,5 óra | Vörös | Tevékenység:  Vörös újból elkészíti a 3.4.1-3.4.8 as szekvenciadiagramokat. |
| 2023.03.24. 18:45 | 4 óra | Gerencsér | Tevékenység  Gerencsér újból elkészíti a 3.4.9-3.4.16 os szekvenciadiagramokat. |
| 2023.03.24. 19:00 | 2,5 óra | Sütheö | Tevékenység:  Sütheö újból elkészíti a state-chartokat a pumpa, valamint csövek állapotairól.  (3.5.1-3.5.2) |
| 2023.03.25. 13:00 | 5 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet, a dokumentum javításainak átvizsgálása.  Döntés: Sütheö felveszi az osztályok leírásához az újonnan felvett osztályok, valamint interfészeket az osztályok leírása részbe, ezenfelül pedig átírja az Element osztály megváltozott metódusait szintén az osztályok leírása részbe ma (03.25.) 22:00 óráig. |
| 2023.03.25. 18:15 | 1,5 óra | Sütheö | Tevékenység:  Sütheö megírja a PointCounter, WaterFlow, Fixable, Moveable interfészek leírását, valamint a Desert osztály leírását.  (3.3.2, 3.3.4, 3.3.6, 3.3.10, 3.3.15) |

# 5. Szkeleton tervezése

## 5.1 A szkeleton modell valóságos use-case-ei

### 5.1.1 Use-case diagram



### 5.1.2 Use-case leírások

5.1.2.1

| **Use-case neve** | **játékos mozgatása** |
| --- | --- |
| **Rövid leírás** | A játékos a csöveken és játékelemeken keresztül tud közlekedni a pályán. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.**  A játékos kiválasztja, hogy melyik elemre akar mozogni.  **2.** A játékos rálép a választott elemre. |
| **Alternatív forgatókönyv** | **1. A.** Játékos nem tud rálépni az elemre, ezért marad a helyén. |
| **Refek** | Szekvencia diagram(ok):  5.3.11 5.3.12 Kommunikációs diagram(ok): 5.4.11 5.4.12 |

5.1.2.2

| **Use-case neve** | **csőre lépés** |
| --- | --- |
| **Rövid leírás** | A játékosok a csöveken tudnak átmenni játékelemek között. Egyszerre egy csövön csak egy játékos tartózkodhat. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** A játékos egy játékelemen állva ki tudja választani melyik csőre akar lépni.  **2.** A játékos rálép a kiválasztott csőre. |
| **Alternatív forgatókönyv** | **1.A.** Már áll azon a csövön egy másik játékos, ezért a helyén marad, aki oda akart mozogni. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.11 Kommunikációs diagram(ok): 5.4.11 |

5.1.2.3

| **Use-case neve** | **játékelemre lépés** |
| --- | --- |
| **Rövid leírás** | A játékosok egy csövön állva tudnak játékelemekre mozogni és interaktálni azokkat. Ezekről tudnak utána további csövekre mozogni. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** A játékos kiválasztja melyik játékelemre akar lépni.  **2.** A játékos rálép a választott elemre. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.12 Kommunikációs diagram(ok): 5.4.12 |

5.1.2.4

| **Use-case neve** | **szabad cső generálás** |
| --- | --- |
| **Rövid leírás** | A ciszternán cső végek jönnek létre. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** Adott időközönként új cső keletkezik a ciszternánál, amelynek egyik vége a ciszternához van kötve, a másik vége szabad. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.16 Kommunikációs diagram(ok): 5.4.16 |

5.1.2.5

| **Use-case neve** | **pumpa felvétel** |
| --- | --- |
| **Rövid leírás** | A szerelő pumpát tud felvenni a ciszternán. Egyszerre csak egyet tud felvenni. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** A szerelő megpróbálja felvenni a pumpát a ciszternán  **2.** A szerelő felveszi a pumpát. |
| **Alternatív forgatókönyv** | **1.A.** Ha a szerelőnek már van egy pumpa a kezében akkor nem tudja felvenni a másikat. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.13 Kommunikációs diagram(ok): 5.4.13 |

5.1.2.6

| **Use-case neve** | **pumpa letétel** |
| --- | --- |
| **Rövid leírás** | A szerelő a kezében lévő pumpát tudja letenni egy csőre. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** A szerelő leteszi a pumpát a csőre  **2.** A cső kettéválik  **3.** A 2 cső a pumpára lesznek csatlakozva |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.14 Kommunikációs diagram(ok): 5.4.14 |

5.1.2.7

| **Use-case neve** | **cső felcsatolás** |
| --- | --- |
| **Rövid leírás** | Ha a szerelőnél van cső és egy játékelemen áll, fel tudja rá csatlakoztatni, ahonnan lecsatlakoztatta. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** Ha nincs nála szabad csővég vagy nem játékelemen áll, nem történik semmi. |
| **Alternatív forgatókönyv** | **1.A.** Ha van nála szabad csővég és játékelemen áll, akkor kiválasztja azt a játékelemet a szabad csővégnek.  **2.A.** Összeköti a csövet a kiválasztott játékelemmel. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.8 Kommunikációs diagram(ok): 5.4.8 |

5.1.2.8

| **Use-case neve** | **cső lecsatolás** |
| --- | --- |
| **Rövid leírás** | A játékos kiválasztja melyik csövet akarja lecsatlakoztatni arról a játékelemről, ahol áll. Csak akkor tud lecsatlakoztatni ha játékelemen áll, nem csövön. Csak akkor lehet lecsatolni csövet ha nem pumpa be vagy kimenete. A bele menő víz elfolyik a sivatagba. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** Ha egy pumpa be vagy kimenete a kiválasztott cső akkor nem történik semmi. |
| **Alternatív forgatókönyv** | **1.A.** Ha a cső nem ki/bemenet akkor kiválasztja ezt a csövet a játékos.  **2.A.** A cső lecsatolandó végét leválasztja a játékelemről ahol a játékos áll.   1. Ha van benne víz, akkor az elfolyik a sivatagba. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.9 Kommunikációs diagram(ok): 5.4.9 |

5.1.2.9

| **Use-case neve** | **waterflow** |
| --- | --- |
| **Rövid leírás** | A víz folyását szabályozza. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** A Az elemek között adja át a vizet kimenetre és bemenetre, ha működik az elem. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.7 5.3.9 5.3.15 Kommunikációs diagram(ok): 5.4.7 5.4.9 5.4.15 |

5.1.2.10

| **Use-case neve** | **pumpa waterflow** |
| --- | --- |
| **Rövid leírás** | A pumpa víz folyását szabályozza. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** A tárolóban lévő víz a pumpa kimeneti csövébe kerül a cső kapacitásának mértékében.  **2.** A bemeneti cső vize a pumpa víztárolójába jut. |
| **Alternatív forgatókönyv** | **2.A.** Ha a pumpa el van romolva, akkor nem pumpál vizet a kimeneti elemre. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.15 Kommunikációs diagram(ok): 5.4.15 |

5.1.2.11

| **Use-case neve** | **cső waterflow** |
| --- | --- |
| **Rövid leírás** | A cső bemetébe víz jön be és a másik oldalán lévő játék elemnek adja át a vizet. A kapacitásnak megfelelő vízmennyiség tud csak bejönni a csőbe. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** A csőbe befolyik a víz.  **2.** A cső átadja a vizet a kimeneti végén lévő elemnek. |
| **Alternatív forgatókönyv** | **2.A.** Ha a cső lyukas, akkor nem tud vizet átadni a kimeneti elemnek. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.7 5.3.9 5.3.15 Kommunikációs diagram(ok): 5.4.7 5.4.9 5.4.15 |

5.1.2.12

| **Use-case neve** | **ciszterna waterflow** |
| --- | --- |
| **Rövid leírás** | A ciszterna víz folyását írja le. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** A ciszternába víz folyik be. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.7 Kommunikációs diagram(ok): 5.4.7 |

5.1.2.13

| **Use-case neve** | **forrás waterflow** |
| --- | --- |
| **Rövid leírás** | A forrás víz folyását írja le. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** A forrás vizet visz az összes kimeneti elemekre |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.10 Kommunikációs diagram(ok): 5.4.10 |

5.1.2.14

| **Use-case neve** | **Javítás** |
| --- | --- |
| **Rövid leírás** | A szerelő megjavítja a kiválasztott elemet, ha lehetséges. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.**  Ha a szerelő már megjavított elemen, ciszternán vagy forráson áll, akkor ezzel nem történik semmi. |
| **Alternatív forgatókönyv** | **1.A.** Ha a szerelő csövön áll és el van romolva, akkor megjavítja a csövet.  **2.A.** A cső megjavul és újra továbbítja a vizet a kimenetére. |
| **Alternatív forgatókönyv** | **1.B.** Ha a szerelő pumpán áll és el van romolva, akkor megjavítja a pumpát.  **2.B.** A pumpa megjavul, újra beveszi a vizet a bemenetéről és továbbítja a kimenetére. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.2 5.3.4 Kommunikációs diagram(ok): 5.4.2 5.4.4 |

5.1.2.15

| **Use-case neve** | **Szabotálás** |
| --- | --- |
| **Rövid leírás** | A szabotőr szabotálja a kiválasztott elemet, ha lehetséges. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** Ha a szabotőr már rossz elemen vagy játékelemen áll, akkor nem történik semmi. |
| **Alternatív forgatókönyv** | **1.A.** Ha a szabotőr működő csövön áll, akkor azt elrontja.  **2.A.** Elromlik a cső és hogyha megy bele víz, akkor azt a sivatagba folyatja. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.1 Kommunikációs diagram(ok): 5.4.1 |

5.1.2.16

| **Use-case neve** | **pumpa elrontás** |
| --- | --- |
| **Rövid leírás** | A játék közben bizonyos időnként elromlanak pumpák. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** A játék kiválasztja melyik pumpák romoljanak el.  **2.** Elromlik a pumpa, ezért nem fog a bemenetéről vizet bevenni és a kimenetén kiadni, amíg meg nem javítják. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.3 Kommunikációs diagram(ok): 5.4.3 |

5.1.2.17

| **Use-case neve** | **pontadás** |
| --- | --- |
| **Rövid leírás** | A pontokat számolja a játék során. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** Ha érkezik a játék alatt pont a játékosoknak hozzáadja az eddigiekhez. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.7 5.3.9 5.3.10 5.3.15 Kommunikációs diagram(ok): 5.4.7 5.4.9 5.4.10 5.4.15 |

5.1.2.18

| **Use-case neve** | **szabotőr pontok** |
| --- | --- |
| **Rövid leírás** | Növeli a szabotőrök pontját. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** Ha elfolyik a sivatagban víz akkor növeli a szabotőrök pontját. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.9 5.3.10 5.3.15 Kommunikációs diagram(ok): 5.4.9 5.4.10 5.4.15 |

5.1.2.19

| **Use-case neve** | **szerelő pontok** |
| --- | --- |
| **Rövid leírás** | Növeli a szerelők pontját. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** ha a ciszternába víz folyik, akkor a szerelők pontot kapnak. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.7 Kommunikációs diagram(ok): 5.4.7 |

5.1.2.20

| **Use-case neve** | **bemenet állítás** |
| --- | --- |
| **Rövid leírás** | Az elem bemeneti csövét állítja be a kiválasztott csőre. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** Elzárja a jelenlegi bemeneti csövet.  **2.** Beállítja, hogy a kiválasztott cső legyen a bemenet. |
| **Alternatív forgatókönyv** | **2.A.** Ha a kiválasztott cső az elem kimeneti csöve, akkor nem állítja be bemenetként. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.6 Kommunikációs diagram(ok): 5.4.6 |

5.1.2.21

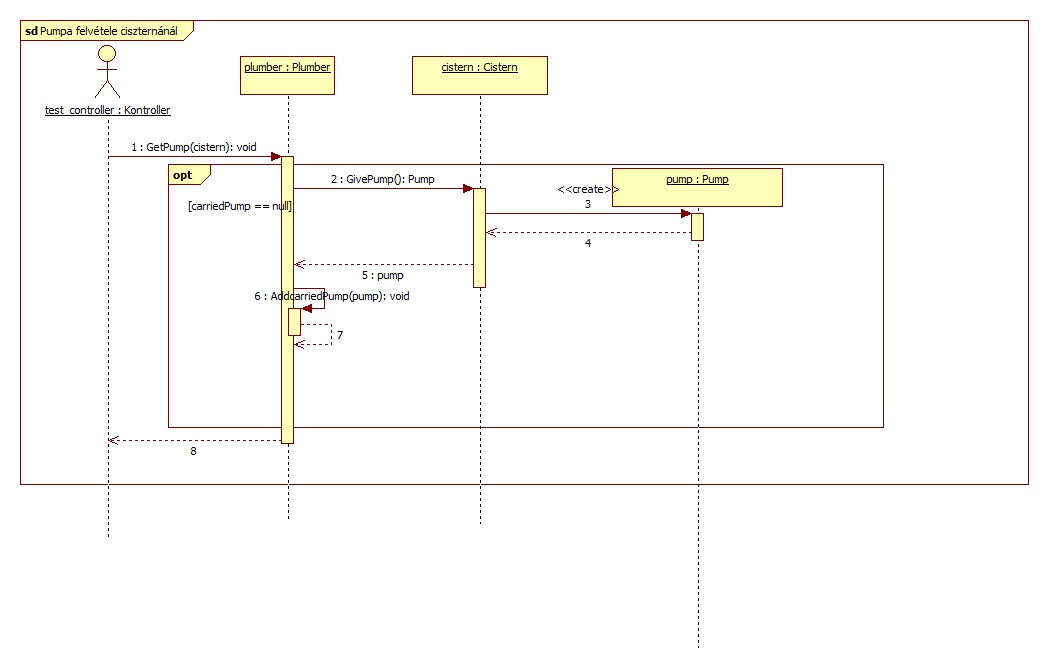
| **Use-case neve** | **kimenet állítás** |
| --- | --- |
| **Rövid leírás** | Az elem kimeneti csövét állítja be a kiválasztott csőre |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | **1.** Elzárja a jelenlegi kimeneti csövet.  **2.** Beállítja, hogy a kiválasztott cső legyen a kimenet. |
| **Alternatív forgatókönyv** | **2.A.** Ha a kiválasztott cső az elem bemeneti csöve, akkor nem állítja be bemenetként. |
| **Kapcsolatok** | Szekvencia diagram(ok): 5.3.6 Kommunikációs diagram(ok): 5.4.6 |

## 5.2 A szkeleton kezelői felületének terve, dialógusok

A szkeleton indulásakor egy menülista jelenik meg, amelyben a teszt use-casek vannak felsorolva. A felhasználó a menüpont sorszámának beütésével futtathatja le az egyes teszteket. A teszt futása során az alábbi leírás alapján lehet követni a teszt futását.

Függvények hívásakor kiíródik ki hívta meg a függvényt, milyen objektumon hívódott meg, milyen paraméterekkel, milyen további függvényeket hív meg az adott függvény, valamint milyen visszatérési értékkel tér vissza. Ha egy objektum konstruktora hívódik meg az ctor előtaggal jelenik meg. Ezt követően kiíródnak a létrejövő objektumnak a tagváltozói, valamint azok kezdeti értékei. A base. jelölésű változók az ősosztály változóit jelölik.

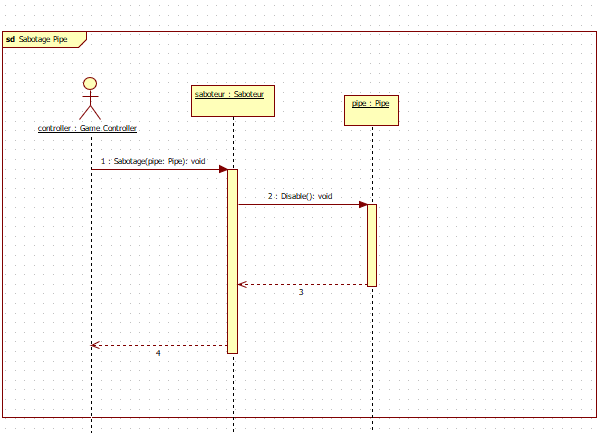
Példa szekvencia futására:

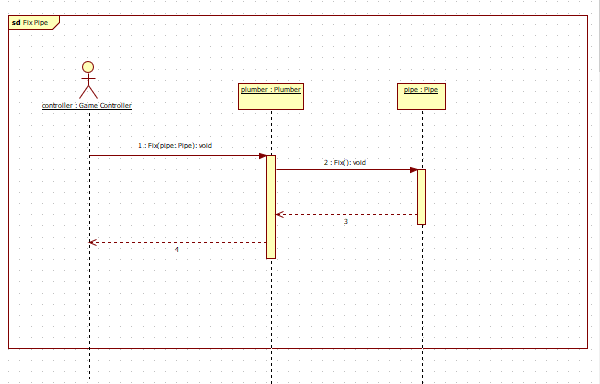


GetPump(cistern) : void  
hívó: test\_controller  
kin hívódott: plumber : Plumber  
hívott függvények:  
 GivePump() : Pump  
 hívó: plumber : plumber: Plumber  
 kin hívódott: plumber : cistern : Cistern  
 hívott függvények:   
 ctor Pump() : pump   
 tulajdonságok:  
 base.maxPipes: 10  
 base.endpoints: null  
 output: null  
 input: null  
 storedWater: 0  
 maxStorage: 20  
 working: true  
 visszatérési érték: -  
 visszatérési érték: pump  
 set: carriedPump: null -> pump  
visszatérési érték: -

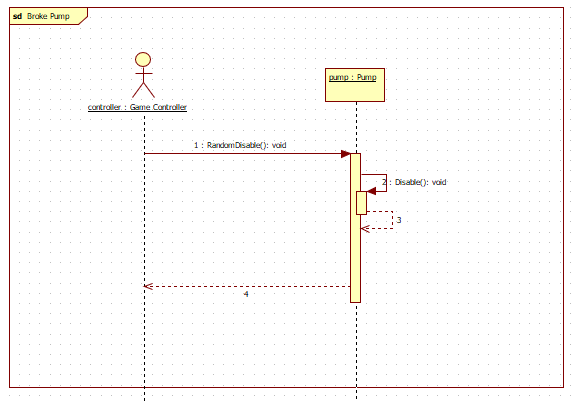
## 5.3 Szekvencia diagramok a belső működésre

5.3.1 Sabotage Pipe

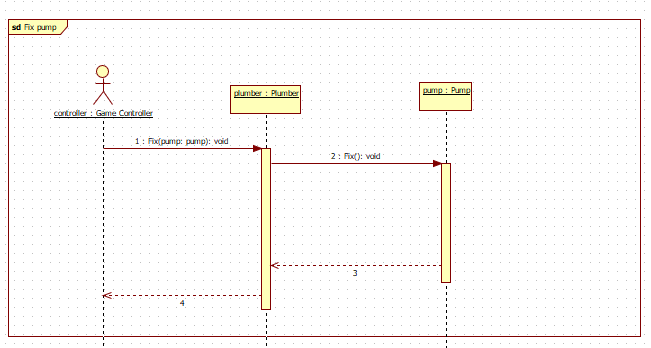


5.3.2 Fix Pipe

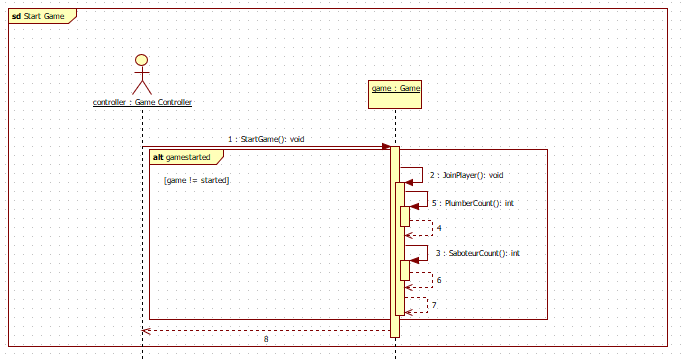
5.3.3 Broke Pump



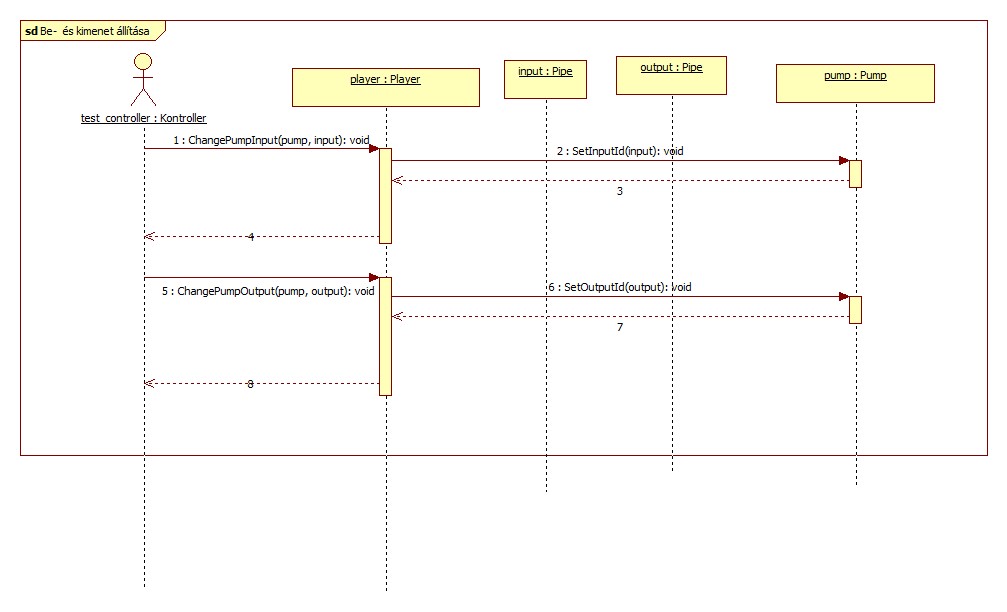
5.3.4 Fix Pump



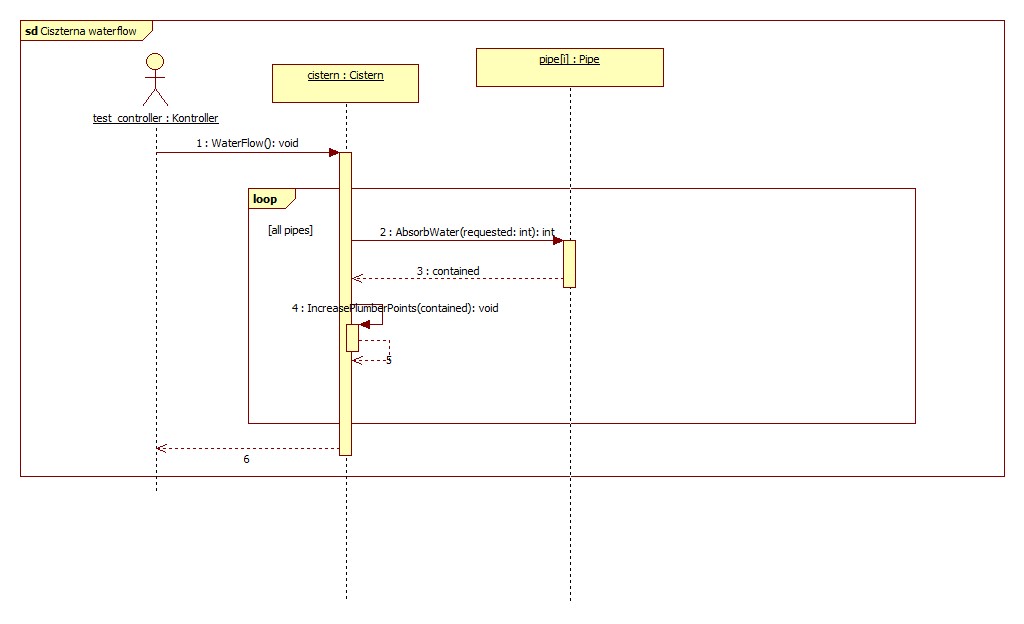
5.3.5 Start Game



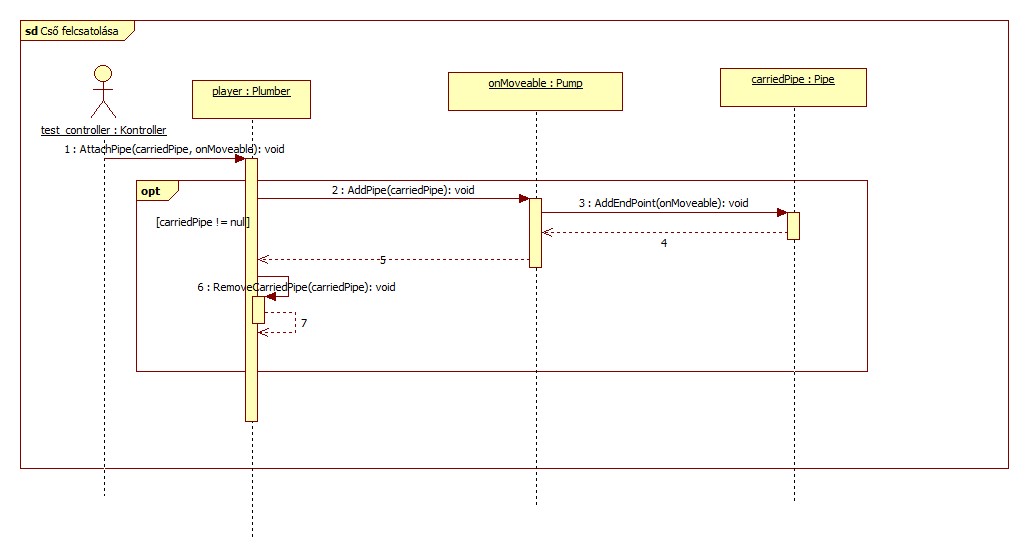
5.3.6 Be- és kimenet állítása



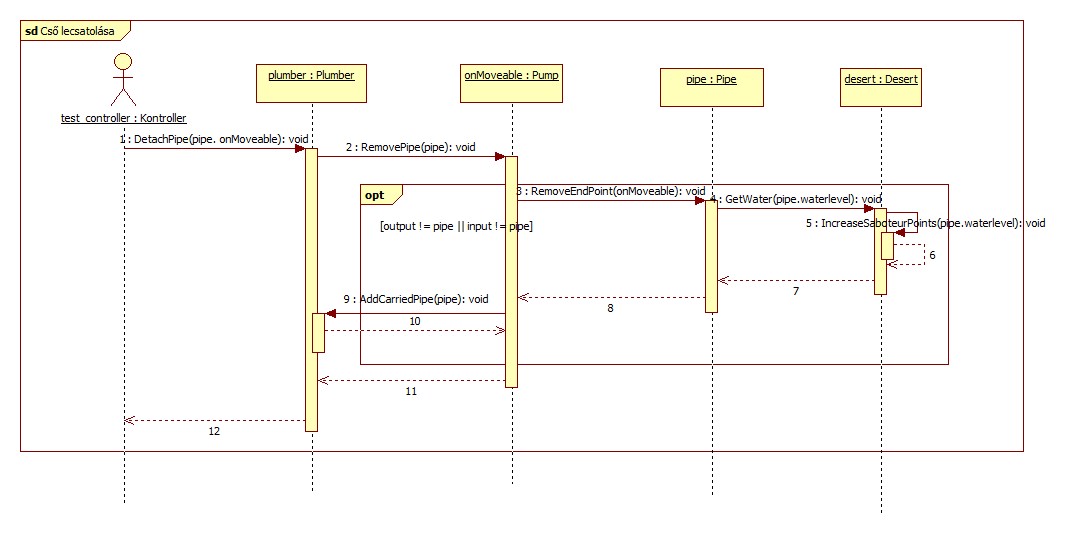
5.3.7 Ciszterna waterflow



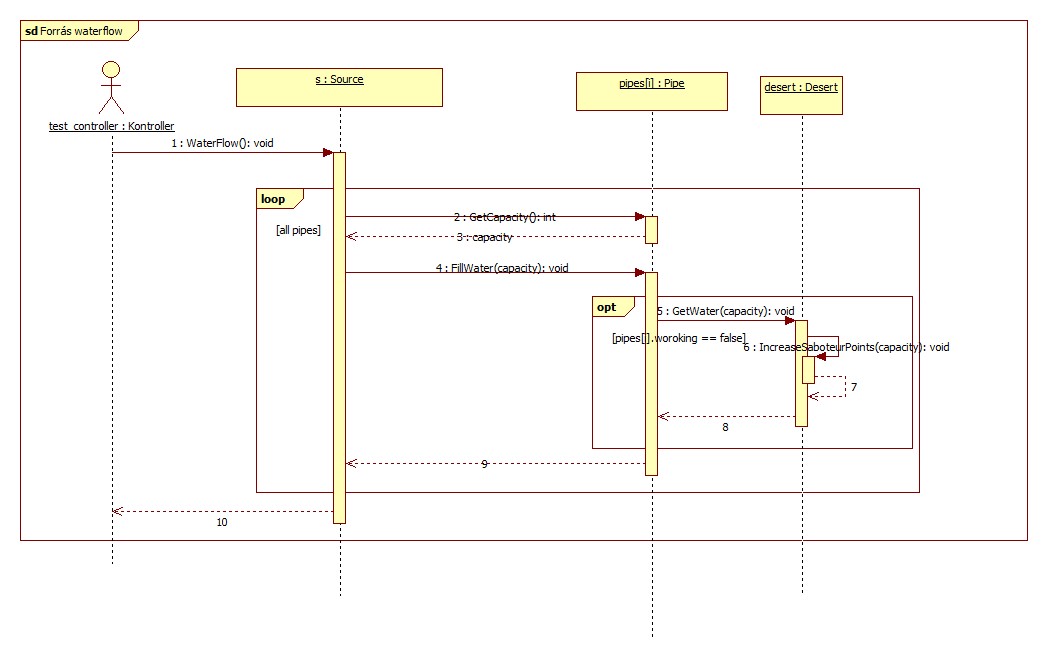
5.3.8 Cső felcsatolása



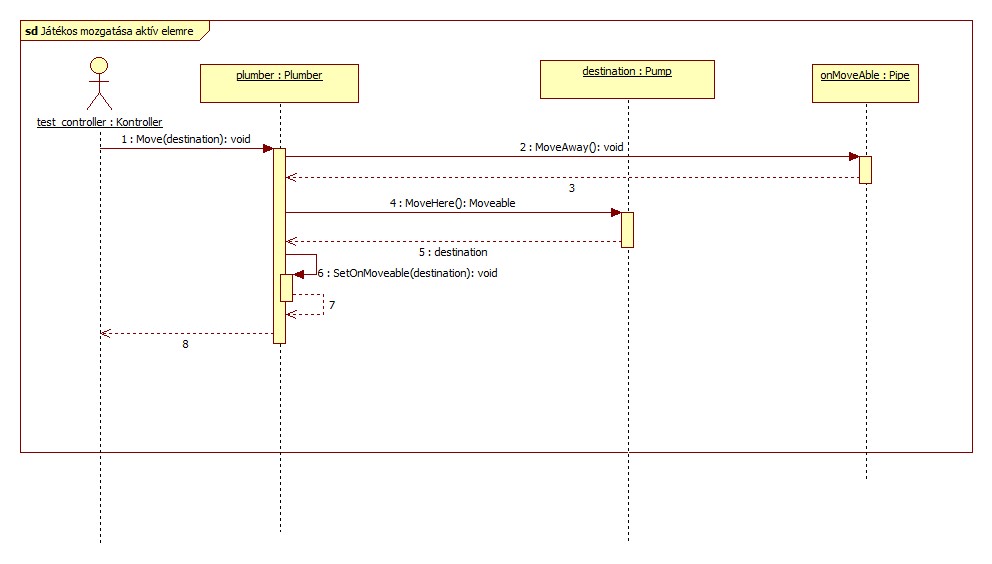
5.3.9 Cső lecsatolása



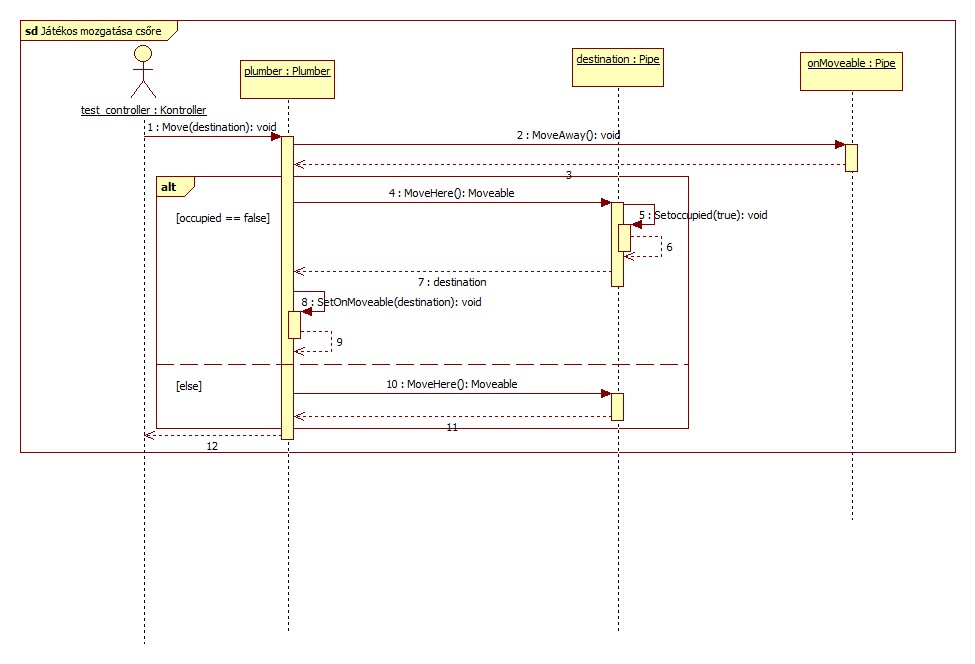
5.3.10 Forrás waterflow



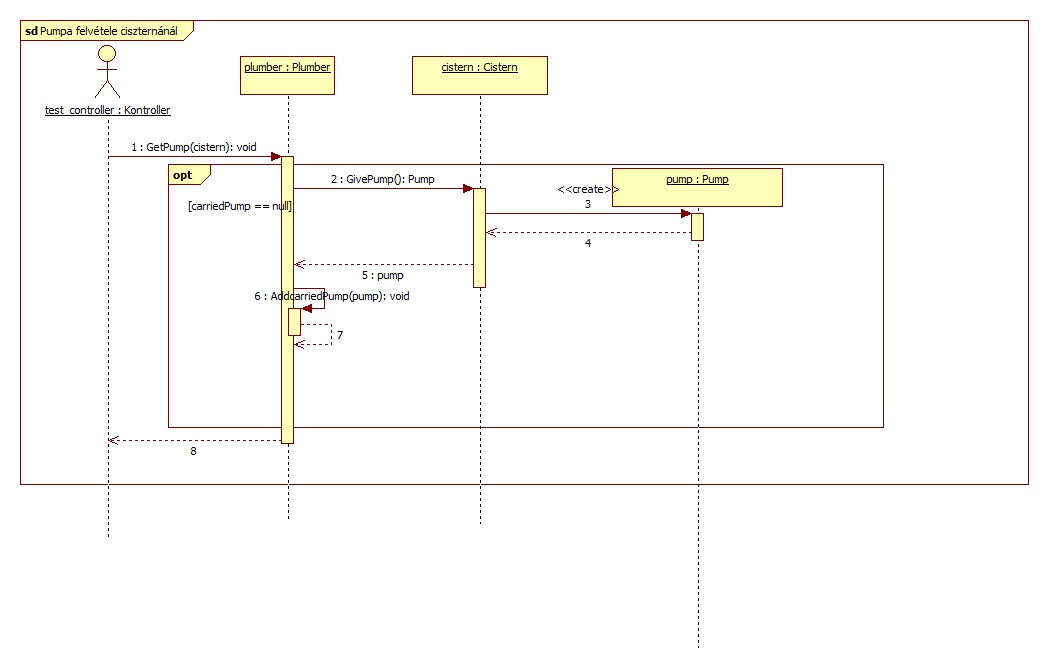
5.3.11 Játékos mozgása aktív elemre



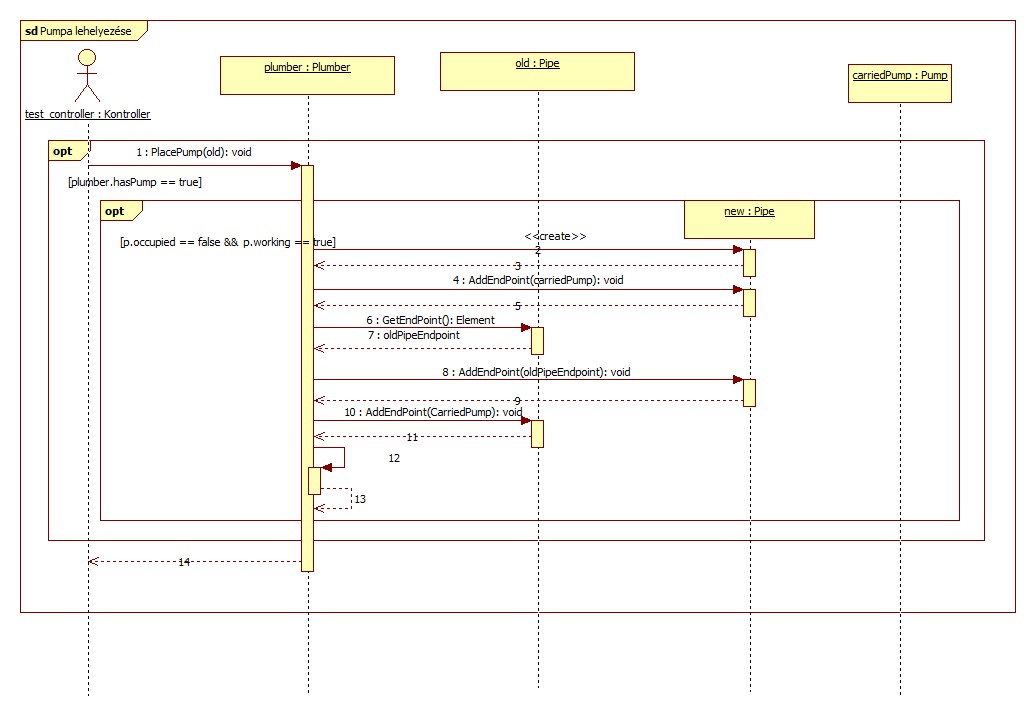
5.3.12 Játékos mozgása csőre



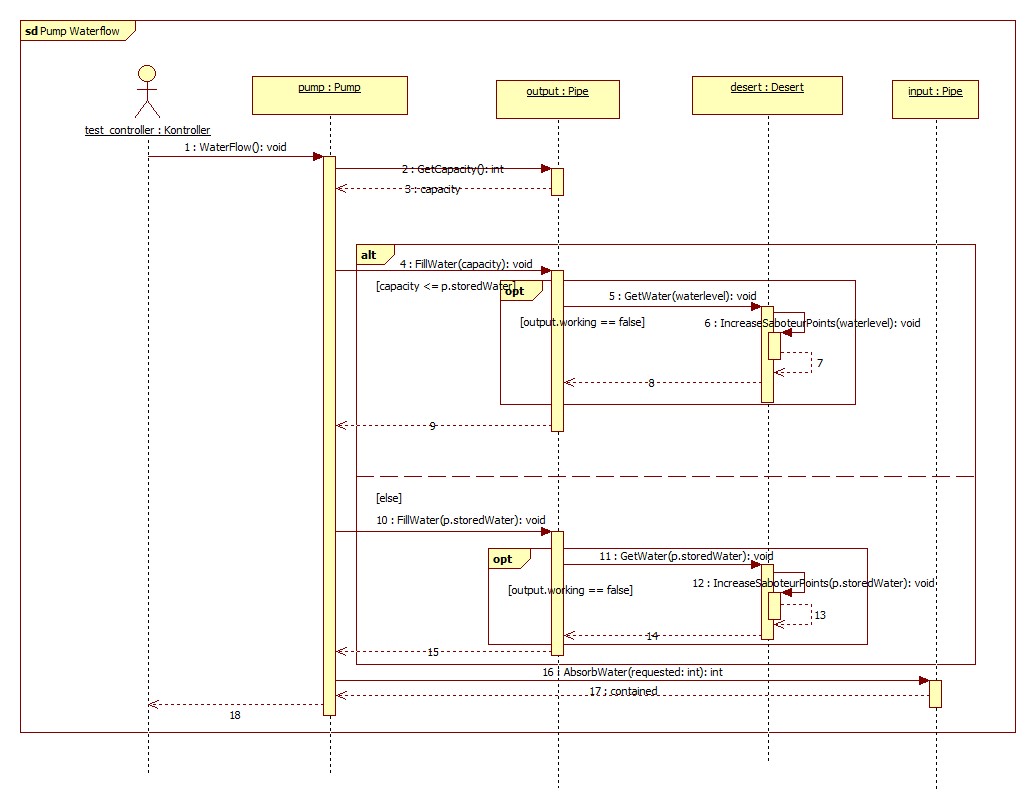
5.3.13 Pumpa felvétele ciszternánál



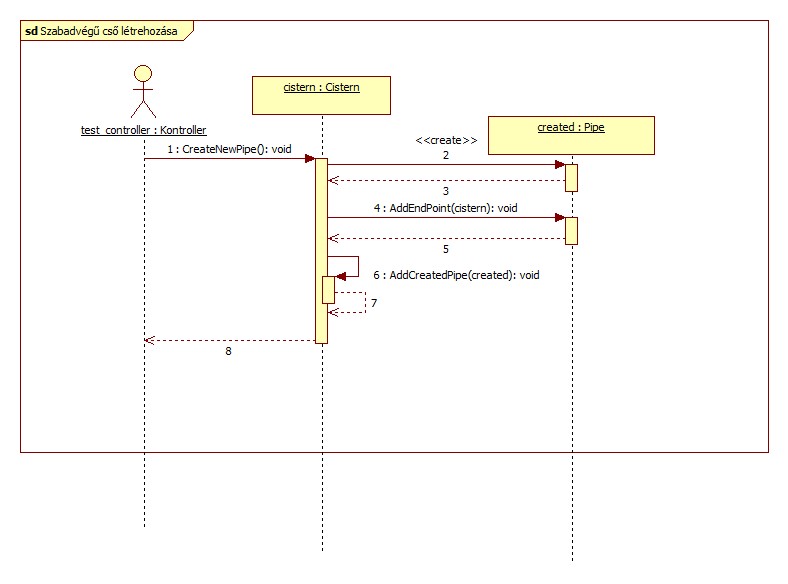
5.3.14 Pumpa lehelyezése



5.3.15 Pumpa waterflow

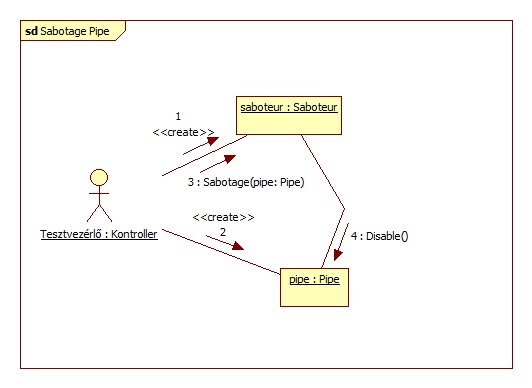


5.3.16 Szabadvégű cső létrehozása

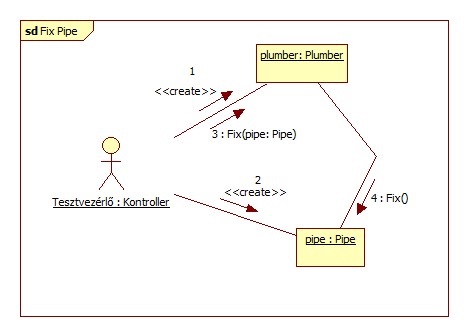


## 5.4 Kommunikációs diagramok

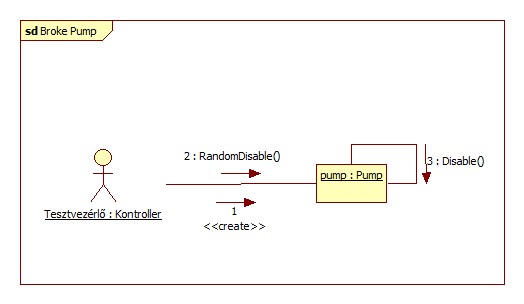
5.4.1 Sabotage Pipe



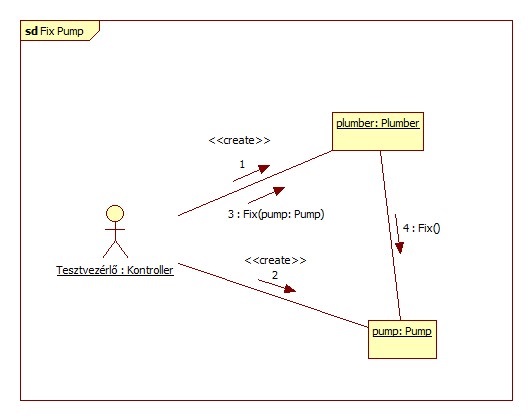
5.4.2 Fix Pipe



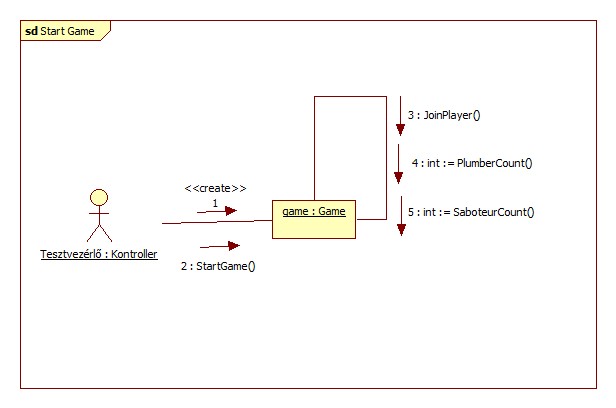
5.4.3 Broke Pump



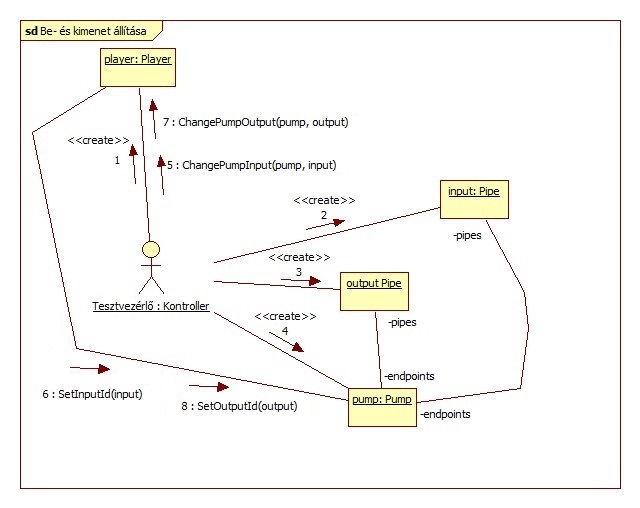
5.4.4 Fix Pump



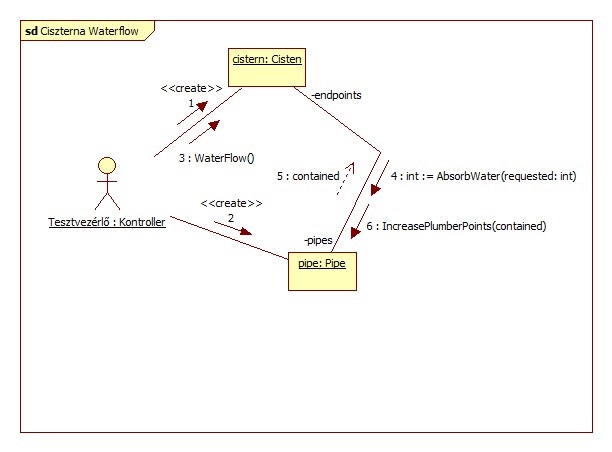
5.4.5 Start Game



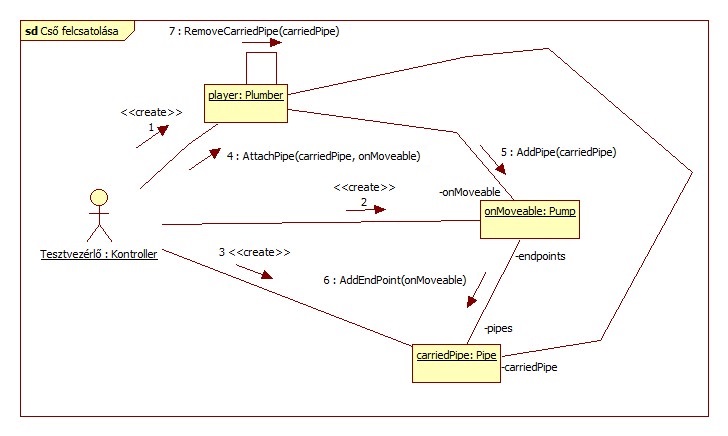
5.4.6 Be- és kimenet állítása



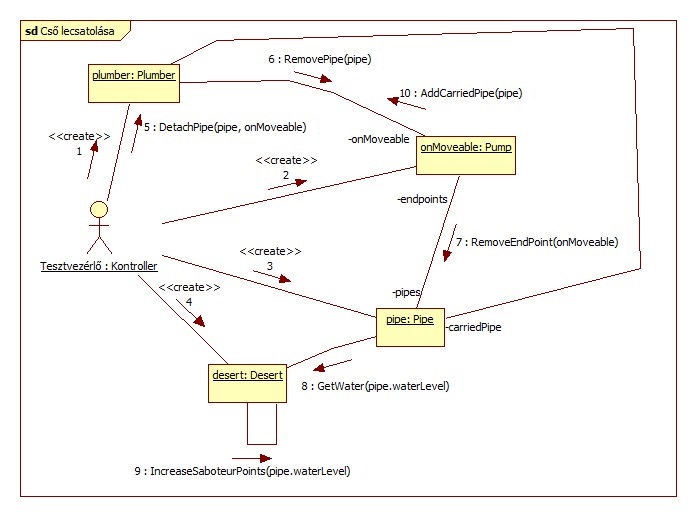
5.4.7 Ciszterna waterflow



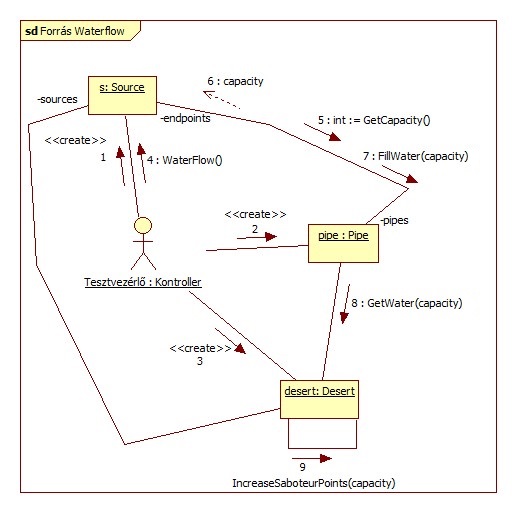
5.4.8 Cső felcsatolása



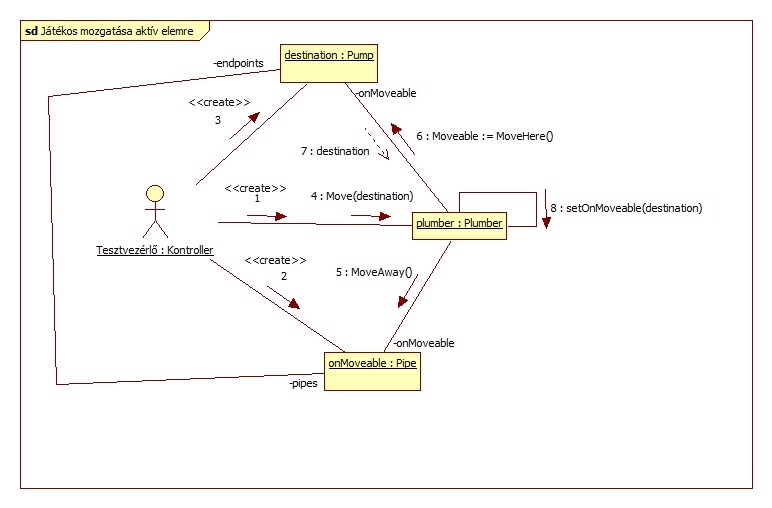
5.4.9 Cső lecsatolása



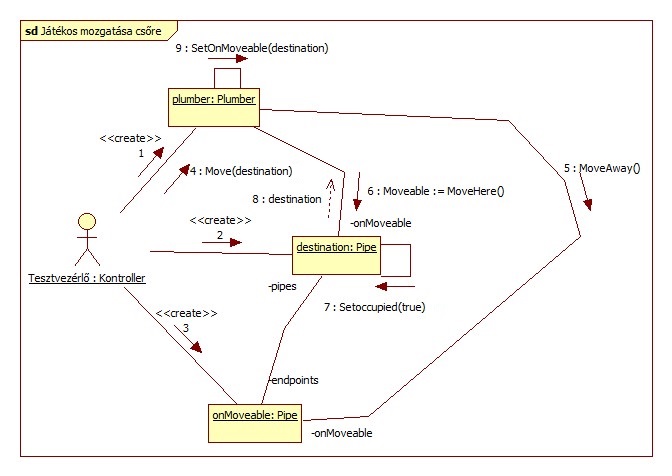
5.4.10 Forrás Waterflow



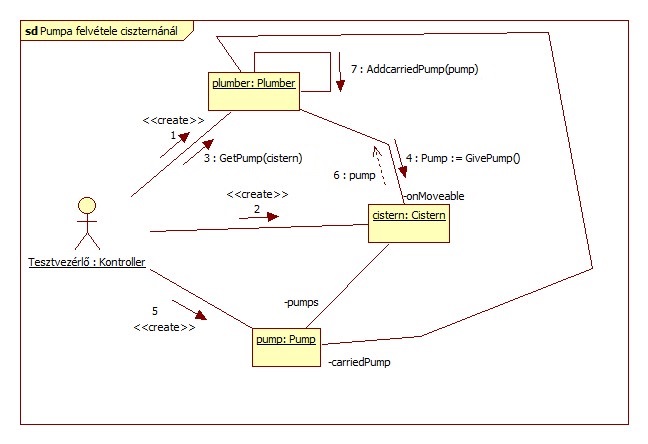
5.4.11 Játékos mozgása aktív elemre



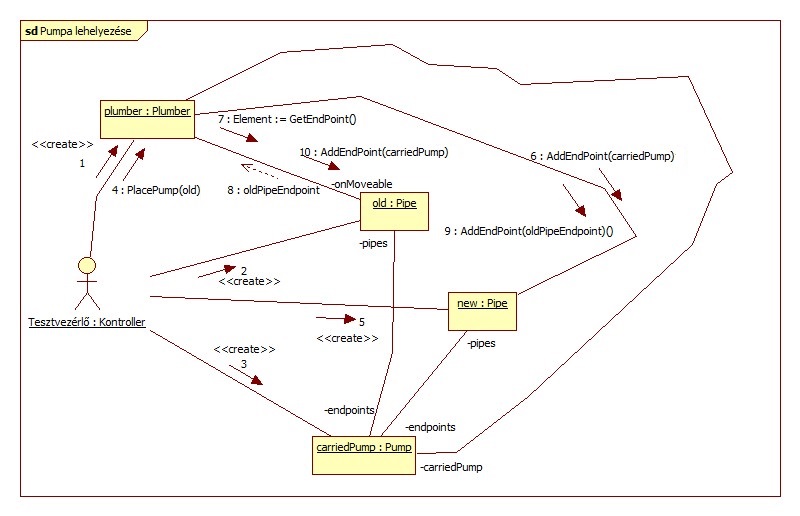
5.4.12 Játékos mozgása csőre



5.4.13 Pumpa felvétele ciszternánál

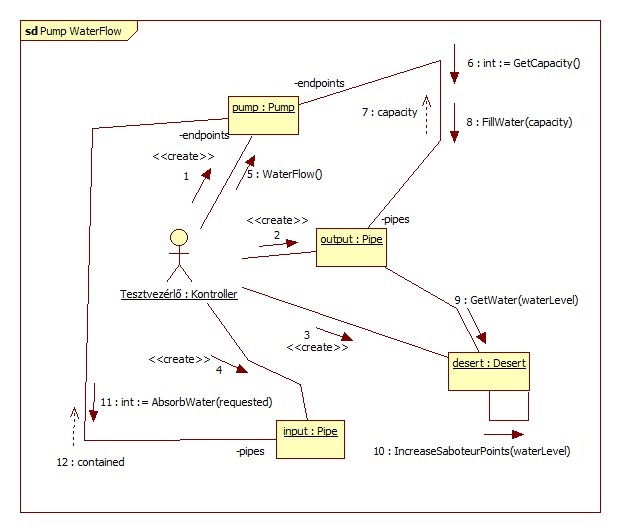


5.4.14 Pumpa lehelyezése



5.4.15 Pumpa waterflow

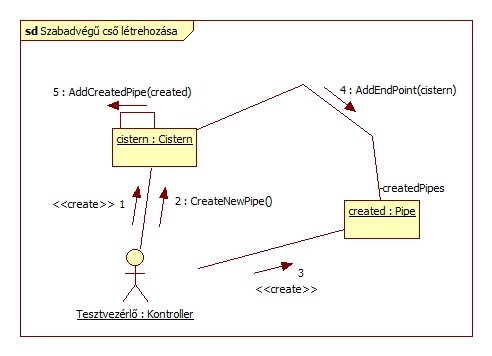
* 5.4.15.1 (ha a cső kapacitása kisebb a pumpa tartályában lévő víz)



* 5.4.15.2 (ha a cső kapacitása legalább akkora, mint a pumpa tartályában lévő víz)



5.4.16 Szabadvégű cső létrehozása



## 5.5 Napló

| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| --- | --- | --- | --- |
| 2023.03.29. 17:00 | 3 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet.  Döntés: Bognár kijavítja az osztálydiagramon felmerülő hibákat (TDA-k) ma (03.29) este 23:00-ig. Bognár elkészíti a use-case diagramot csütörtök (03.30) 19:00-ig. Chu és Bognár elkészítik a use-case leírásokat péntek délután (03.31) 18:00-ig. Vörös és Gerencsér elkészítik a szekvencia diagramokat szombat (04.01) 10:00-ig. Vörös elkészíti a szkeleton kezelői felületének szombat (04.01) 14:00-ig. Sütheö elkészíti a kommunikációs diagramokat vasárnap (04.02) 12:00-ig. |
| 2023.03.29. 17:30 | 1 óra | Bognár | Tevékenység: Bognár elkészíti a kijavított osztálydiagramot. (5.0) |
| 2023.03.30. 14:00 | 3 óra | Bognár | Tevékenység: Bognár elkészíti a use-case diagramot. (5.1.1) |
| 2023.03.31. 13:00 | 5,5 óra | Chu | Tevékenység: Chu elkészít a use-case leírásokból 16 darabot. (5.1.2.1-5.1.2.16) |
| 2023.03.31. 17:00 | 1,5 óra | Bognár | Tevékenység: Bognár elkészít a use-case leírásokból 5 darabot. (5.1.2.17-5.1.2.21) |
| 2023.03.31. 20:00 | 2 óra | Vörös | Tevékenység: Vörös elkészít 5 darab szekvenciadiagramot. (5.3.1-5.3.5) |
| 2023.03.31. 21:30 | 4 óra | Gerencsér | Tevékenység: Gerencsér elkészít 6 darab szekvenciadiagramot. (5.3.6-5.3.11) |
| 2023.04.01. 10:00 | 2 óra | Gerencsér | Tevékenység: Gerencsér elkészít 5 darab szekvenciadiagramot. (5.3.12-5.3.16) |
| 2023.04.02. 14:30 | 3 óra | Vörös | Tevékenység: Vörös elkészíti a szkeleton kezelői felületének tervét, dialógusait. Ehhez egy példát is megad szemléltetésképp. (5.2) |
| 2023.04.01. 16:00 | 5,5 óra | Sütheö | Tevékenység: Sütheö elkészíti a kommunikációs diagramokat. (5.4.1-5.4.16) |
| 2023.04.02. 13:00 | 1,5 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet, a dokumentum teljes körű átvizsgálása.  Döntés: Bognár felvesz alfajtákra tagolja az egyes teszteseteket a use-case diagramban ma (04.02) este 19:00-ig. |
| 2023.04.02. 15:00 | 0,5 óra | Bognár | Tevékenység: Bognár alfajtákra bontja a Waterflow, Játékos mozgatása, Pontadás teszteseteket a use-case diagramon. (5.1.1) |

# 6. Szkeleton beadás

## 6.1 Fordítási és futtatási útmutató

### 6.1.1 Fájllista

| **Fájl neve** | **Méret** | **Keletkezés ideje** | **Tartalom** |
| --- | --- | --- | --- |
| Cistern.java | 3,37 KB | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A ciszternák megfelelő metódusait tartalmazza, a beleérkező vízmennyiség alapján növeli a szerelők pontszámát. |
| Controller.java | 16,8 KB | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A tesztesetek ebben a fájlban találhatóak, a konzolos feladatválasztás is itt történik meg. |
| Desert.java | 1,08 KB | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A sivatag megfelelő metódusait tartalmazza, az elfolyó vízmennyiség alapján növeli a szabotőrök pontszámát. |
| Element.java | 2,78 KB | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | Az aktív elemeket reprezentálja, tartalmazza az általános függvényeiket, amelyeket a leszármazott osztályok felülír(hat)nak. |
| Fixable.java | 71 bájt | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | Interfész, amely a megjavíható, ezáltal elrontható dolgokat reprezentálja. Ezen metódusokat a leszármazott osztályok felülírják. |
| FunctionCall.java | 2,31 KB | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A hívott függvények sorozatának tárolására, valamint kiiratására szolgál. (ki, kit, milyen paraméterkkel, melyik függvényt, mi változott) |
| Game.java | 365 bájt | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A játékot reprezentáló osztály, elkezdi, ellenőrzi, valamint befejezi azt. |
| Movable.java | 82 bájt | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | Interfész, amely azon elemeket reprezentálja, amelyekre a játékosok képesek rálépni. |
| Pipe.java | 8,59 KB | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A csövek változóit és függvényeit valósítja meg. |
| Player.java | 5,32 KB | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A játékosok általános jellemzőit, tevékenységeit reprezentáló osztály. |
| Plumber.java | 3,43 KB | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A szerelők változóit és csapast specifikus függvényeit valósítja meg. |
| PointCounter.java | 74 bájt | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | Interfész, amelyet a pontok számlálására alkalmas metódusokat tartalmazza. |
| Pump.java | 5,67 KB | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A pumpák megfelelő változóit és metódusait valósítja meg. |
| PumpController.java | 65 bájt | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | Interfész, amely a pumpa véletlenszerű elrontására szolgál. |
| Saboteur.java | 596 bájt | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A szabótőr csapat specifikus függvényeit valósítja meg. |
| Source.java | 1,00 KB | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | Osztály amely a játékba áramló vízet generálja a belevezetett csövekbe. |
| WaterFlow.java | 56 bájt | 2023. ‎április ‎12., szerda, ‏‎22:48:08 | A játék során a vízfolyást imitáló interfész. |

### 6.1.2 Fordítás

Először is csomagoljuk ki a zip fájlban található .java fájlokat egy **Test** nevű mappába az **Asztalra** (rakhatjuk bármilyen más mappába is, csak a **cd** paranccsal adjuk meg a mappa elérési útját). Ezután indítsunk egy Command Prompt -ot (cmd.exe) és írjuk be ezt a két sort:

*cd C:\Users\cloud\Desktop\Test*

*javac Cistern.java Controller.java Desert.java Element.java Fixable.java FunctionCall.java Game.java Moveable.java Pipe.java Player.java PointCounter.java Pump.java PumpController.java Saboteur.java Source.java WaterFlow.java*

### 6.1.3 Futtatás

Ha nem zártuk be a Command Prompt -t (cmd.exe) akkor elég ezt az egy sort beírni:

*java Controller*

És a listából a megfelelő számot beírva lefut a megfelelő teszteset, ennek a függvényhívás sorát tudjuk végigkövetni.

Ha bezártuk a Command Prompt-ot (cmd.exe) akkor indítsunk egy újat és írjuk be a következő parancsot két parancsot:

*cd C:\Users\cloud\Desktop\Test*

*java Controller*

Ezek után úgy mint az előző esetben, meg tudjuk hívni a megfelelő teszteseteket a sorszámok beírásával.

## 6.2 Értékelés

| **Tag neve** | **Tag neptun** | **Munka százalékban** |
| --- | --- | --- |
| Sütheö István | XOBJYX | 20 |
| Vörös László | SLLV0S | 20 |
| Gerencsér Máté | YVSPMN | 20 |
| Bognár Zsombor | BNO3XE | 20 |
| Duy Anh Chu | G4H99J | 20 |

## 6.3 Napló

| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| --- | --- | --- | --- |
| 2023.04.11. 16:00 | 4,5 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet.  Döntés: Bognár elkészíti az osztálydiagram alapján a szkeleton vázát, azaz felveszi a megfelelő osztályokat, valamint a függvény defíniciókat. 04.12. 16:00 óráig. Gerencsér, Vörös, Sütheö elkészíti a teszteseteket 04.14. 16:00 óráig. Chu elkészíti a Fordítási és futtatási útmutatót 04.15. 18:00 óráig. |
| 2023.04.12. 8:00 | 3 óra | Bognár | Tevékenység:  Bognár elkészíti az osztálydiagram alapján a megfelelő osztályokat, valamint felveszi azok attribútumait, valamint függvény deklarációit. Létrehozza a Controller, valamint FunctionCall osztályt. |
| 2023.04.12. 11:30 | 1 óra | Bognár | Tevékenység: Bognár eltávolítja az osztálydiagramról a felesleges függvényeket, valamint az Element osztály pipes attribútumának láthatóságát privat-ről protected-re állítja. |
| 2023.04.12. 19:00 | 4 óra | Gerencsér | Tevékenység: Gerencsér elkészít 4 darab tesztesetet (Sabotage Pipe, Fix Pipe, Broke Pump, Fix Pump), valamint elkészíti a FunctionCall osztályt, valamint annak metódusait. |
| 2023.04.13. 14:00 | 4 óra | Sütheö | Tevékenység: Sütheö elkészít 6 darab tesztesetet. (Move to Pipe, Pick up Pump, Place Pump, Pump WaterFlow, Create Free Pipe, Move to Active Element) |
| 2023.04.13. 19:00 | 3,5 óra | Vörös | Tevékenység: Vörös elkészít 6 darab tesztesetet. (Set Input ,Set Output, Cistern Waterflow, Attach Pipe, Detach Pipe, Source Waterflow) |
| 2023.04.14. 20:00 | 4 óra | Chu | Tevékenység: Chu elkészítit a Fordítási és futtatási útmutatót. (6.1.1, 6.1.2, 6.1.3) |
| 2023.04.15. 19:00 | 2 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet, eddigi teljesítmények értékelése.  Döntés: Sütheö kitölti az értékelést. Vörös létrehozza a tesztesetválasztó use-case szerkezetet 04.15 23:59 -ig. |
| 2023.04.15. 21:15 | 0,5 óra | Sütheö | Tevékenység: Sütheö kitölti a csapattagok értékelésére szolgáló 6.2-es részt. |
| 2023.04.15. 21:30 | 0,5 óra | Vörös | Tevékenység: Vörös elkészíti a tesztesetválasztó use-case szerkezetet a Controller osztályban. |

**7. Prototípus koncepciója**

1 - Ballers

Konzulens:

Dobos-Kovács Mihály

Csapattagok

| Sütheö István | XOBJYX | istvan.sutheo@gmail.com (kapcsolattartó) |
| --- | --- | --- |
| Vörös László | SLLV0S | lacko1001@gmail.com |
| Gerencsér Máté | YVSPMN | [gerencsermate02@gmail.com](mailto:gerencsermate02@gmail.com) |
| Bognár Zsombor  Duy Anh Chu | BNO3XE  G4H99J | bzsombi99@gmail.com  chuduyanh@gmail.com |

2023.04.24.

# 

# 

# 7. Prototípus koncepciója

## 7.1 Prototípus interface-definíciója

A prototípusunk egy konzolos felületű, parancsokkal (7.1.2-ben definiált parancsokkal) lesz irányítható. Egy tesztesetet akkor veszünk sikeresnek, ha annak kimenete az elvárt kimenettel megegyezik.

**7.1.1** **Az interfész általános leírása**

A tesztesetek bemenete érkezhet a standard bemenetről vagy fájlból. Az input feldolgozása soronként történik. Az automatikus futtatás a fájlokból történik, a kézi tesztelés pedig a standard bemenettel történik. A kiíratás ugyanígy történhet a standard kimenetre, vagy fájlba is. A bemeneti, illetve kimeneti nyelv szintaktikáját, szemantikáját a következőkben fogjuk ismertetni.  
Két nemdeterminisztikus történés van a modellben, az egyik a pumpák véletlenszerű elromlása, a másik pedig a csúszós csőről egy aktív elemre véletlenszerű mozgás. Ez előbbit a Move paranccsal, az utóbbit pedig a Sabotage paranccsal lehet determinisztikusan futtatni.

### 7.1.2 Bemeneti nyelv

**CreatePlumber**

**Leírás:** Létrehoz egy új szerelőt.

**Opciók:** Meg lehet adni a hasPump, id, carriedPump, onMoveable, carriedPipe paramétereket.

**CreateSaboteur**

**Leírás:** Létrehoz egy új szabotőrt.

**Opciók:** Meg lehet adni a, id, onMoveable, carriedPipe paramétereket.

**CreatePump**

**Leírás:** Létrehoz egy új pumpát.

**Opciók:** Meg lehet adni az output, input, storedWater, maxStorage, working, pipes, maxPipes paraméteket.

**CreatePipe**

**Leírás:** Létrehoz egy új csövet.

**Opciók:** Meg lehet adni a waterLevel, working, capacity, occupied, sabotageable, sticky, slippery, endpoints paramétereket.

**CreateCistern**

**Leírás:** Létrehoz egy új ciszternát.

**Opciók:** Meg lehet adni a, numberOfFreePipes, createdPipes, , pipes, maxPipes paramétereket.

**CreateSource**

**Leírás:** Létrehoz egy új forrást.

**Opciók:** Meg lehet adni a pipes, maxPipes paramétereket.

**List**

**Leírás:** Kilistázza az összes példányát a megadott osztálynak.

**Opciók:** Megadható a Plumber, Saboteur, Pump, Source, Pipe, Cistern listázható osztályként.

**Info**

**Leírás:** Kiírja a megadott dolog tulajdonságait.

**Opciók:** Meg kell adni a kiírandó dolog nevét, ami a map-ben is szerepel.

**Load**

**Leírás:** Betölt egy pálya állást a megadott fájlból.

**Opciók:** Meg kell adni a fájl nevét.

**Save**

**Leírás:** Elmenti a pálya állását a megadott fájlba.

**Opciók:** Meg kell adni a fájl nevét.

**Points**

**Leírás:** Kiírja mindkét csapat pontszámát.

**Opciók:** -

**IncreasePoints**

**Leírás:** Megnöveli a pontokat a megadott csapatnak.

**Opciók:** Meg kell adni a csapatot és hogy mennyivel növelje a pontokat.

**PickUpPump:**

**Leírás:** Pumpa felvétele a kézbe.

**Opciók:** Meg kell adni első paraméterként, hogy ki veszi fel a pumpát és honnan akarja felvenni.

**PutDownPump:**

**Leírás:** Pumpa lerakása a csőre

**Opciók:** Meg kell adni első paraméterként, hogy ki rakja a pumpát. Ezután a csőt kell megadni.

**FixGameElement:**

**Leírás:** Javítható elem javítása

**Opciók:** Meg kell adni a szerelőt és 2. paraméternek, hogy melyik elemet akarja megjavítani

**MakeSlippery:**

**Leírás:** A cső változtatása, hogy csúszós legyen

**Opciók:** Meg kell adni a szabotőrt és hogy melyik csövet akarja csúszóssá tenni.

**MakeSticky:**

**Leírás:** A cső változtatása, hogy ragadós legyen

**Opciók:** Meg kell adni a játékost és a csövet, amit ragadóssá akar tenni.

**Sabotage:**

**Leírás:**  Cső elrontása.

**Opciók:** Meg kell adni a játékost és a csövet, amit a játékos ki akar lyukasztani.

**PickUpPipeEnd:**

**Leírás:** Csővég felvétele a kézbe.

**Opciók:** Meg kell adni a játékost és a csőt

**PutDownPipeEnd:**Cső vég lerakása és csatlakoztatása az elemhez

**Leírás:**

**Opciók:** Meg kell adni a játékost, az aktív elemet.

**ChangePumpInput:**

**Leírás:** Pumpa bemenetének változtatása

**Opciók:** Első paramétere, hogy melyik pumpának a bemenetét akarjuk megváltoztatni. 2. paraméter: Melyik cső lesz a bemenete a pumpának.

**ChangePumpInputByPlayer:**

**Leírás:** Pumpa bemenetének változtatása

**Opciók:** Első paramétere, hogy melyik pumpának a bemenetét akarjuk megváltoztatni. 2. paraméter: Melyik cső lesz a bemenete a pumpának, 3. paraméter: ki változtatja meg

**ChangePumpOutputByPlayer:**

**Leírás:** Pumpa kimenetének változtatása

**Opciók:**  Első paramétere, hogy melyik pumpának a kimenetét akarjuk megváltoztatni. 2. paraméter: Melyik cső lesz a kimenete a pumpának, 3. paraméter: ki változtatja meg

**ChangePumpOutput:**

**Leírás:** Pumpa kimenetének változtatása

**Opciók:**  Első paramétere, hogy melyik pumpának a kimenetét akarjuk megváltoztatni. 2. paraméter: Melyik cső lesz a kimenete a pumpának

**Move**

**Leírás:** Játékos mozgatása.

**Opciók:**  Meg kell adni a játékost és egy Moveable elemet , ahova a játékos szeretne menni.

**GetNeighbours:**

**Leírás:** Kiírja az összes szomszédos játékelemet .

**Opciók:**  Meg kell adni a játékelemet, aminek a szomszédait kérjük le/

**MoveSlippery:**

**Leírás:** Cső véletlenszerű végére mozgatja a játékost.

**AddPipe**

**Leírás:** Cső csatlakoztatása pumpához.

**Opciók:** Meg kell adni a pumpát és a csövet.

**WaterFlow**

**Leírás:** Aktív elemen folyat át vizet.

**Opciók:** meg kell adni az aktív elemet, amin vizet szeretnénk átfolyatni

**AddFreePipe**

**Leírás:** Szabad végűcső hozzáadása ciszternához

**Opciók:** Meg kell adni milyen néven jöjjön létre a cső és hogy hol(ciszterna neve).

**PickUpFreePipe**

**Leírás:** Szabad végűcső felvétele ciszternáról

**Opciók:** Meg kell adni hogy ki és honnan vegye fel a szabad végű csövet.

**A pályákat tartalmazó fájlok felépítése:**

A fileben egymás utáni sorokban a következő sorrendben lesznek kiírva az alábbi objektumok:  
(A saját id minden esetben a Map objektumban tárolt lisa elemének indexe lesz, ezt automatikusan kapják meg az inicializálásakor)

**Pump**  
[\*, <id>,<storedWater>,<maxStorage>,<working>, <input\_id>, <output\_id>, <csatlakozó csövek száma>, <cső\_id\_1>, …, <cső\_id\_n>]

**Cistern**  
[-, <id>,, <csatlakozó csövek száma>, <Pipe\_id\_1>, …, <Pipe\_id\_n>]

**Source**  
[+, <id>,, <csatlakozó csövek száma>, <Pipe\_id\_1>, …, <Pipe\_id\_n>]

**Pipe**  
[=, <id>,<waterLevel>,<working>,<capacity>, <occupied>, <sabotageable>, <sticky>, <slippery>, <endpoint1>, <endpoint2>]

**Plumber**  
[P, <id>, <hasPump>,<carriedPump>, <onMoveable\_id>, <carriedPipe\_id>]

**Saboteur**  
[S, <id>, <onMoveable\_id>, <carriedPipe\_id>]

### 7.1.3 Kimeneti nyelv

**CreatePlumber:** <Szerelő neve> szerelő létrejött  
Adatai: hasPump: <hasPump értéle>,ID: <id>, /\*canMove: <canMove értéke>\*/, carriedPump:<carriedPump neve>, location: <onMoveable értéke>, carriedPipes: <carriedPipe neve>.

**CreateSaboteur:** <Szabotőr neve>szabotőr létrejött  
Adatai:, hasPump: <hasPump értéle>,ID: <id>, canMove: <canMove értéke>, location: <onMoveable értéke>, carriedPipes: <carriedPipe neve>.

**CreatePump:**  <Pump neve > pumpa létrejött  
Adatai: output: <output>, input: <input>, storedWater: <storedWater>, maxStorage: < maxStorage>, working: <working>, pipes: <pipes nevei>, maxPipes: <maxPipes >.

**CreatePipe:** <cső neve > cső létrejött  
Adatai: waterLevel: <waterLevel>, working: <working>, capacity: <capacity>, occupied: <occupied>, sabotageable: <sabotageable>, sticky: < sticky>, slippery: <slippery>, endpoints: <endpoints>

**CreateCistern:** <ciszterna neve > ciszterna létrejött  
Adatai: /\*hasPump: < hasPump>\*/, numberOfFreePipes: <numberOfFreePipes>, createdPipes: <createdPipes>, /\*pumps: <pumps> \*/, pipes: < pipes nevei>, maxPipes: <maxPipes>

**CreateSource:** <ciszterna neve > ciszterna létrejött  
Adatai:pipes: <pipes nevei>, maxPipes: <maxPipes>

**List:** <A pálya összes elem neve>

**Info:** <Az elem adatai>

**Load:** A pályát <betöltés sikeressége> betöltöttük.

**Save:** A pályát <betöltés sikeressége> elmentettük.

**Points:** Szabotőr pontja: <szabotőr pont > \n

Szererlő pontja: <szerelő pont>

**IncreasePoints:** <csapata neve> pontszámát <noveles szama> ponttal növeltük

**PickUpPump:** <Szerelő neve><javítás sikeressége> vette le <Pumpa neve>-t <ciszterna neve>-ból

**PutDownPump:** <Szerelő neve><sikeressége> tette le <Pumpa neve>-t <cső neve>-re.

**FixGameElement :**<Szerelő neve><javítás sikeressége> javította <javítható elem neve>-t

**MakeSlippery:**<Játékos neve><sikeresség> csúszóssá tette <cső neve> csövet

**MakeSticky:** <Játékos neve><sikeresség> ragadóssá tette <cső neve> csövet

**Sabotage:** <Játékos neve><Szabotálás sikeressége> kilyukasztotta <cső neve>-t

**PickUpPipeEnd:** <Játékos neve><sikeresség> felvette <cső neve> cső végét <Element neve> játékelemről.

**PutDownPipeEnd:** <Játékos neve><sikeresség> letette <cső neve> cső végét <Element neve> játékelemrel.

**ChangePumpInputByPlayer:** <Játékos neve><sikeresség> megváltoztatta <pumpa neve> bemeneti csővét <cső neve> csőre.

**ChangePumpInput:** <pumpa neve> bemeneti csöve <cső neve> csőre változott.

**ChangePumpOutputByPlayer:** <Játékos neve><sikeresség> megváltoztatta <pumpa neve> bemeneti csővét <cső neve> csőre.

**ChangePumpOutput:** <pumpa neve> kimeneti csöve <cső neve> csőre változott.

**Move:** <Játékos neve><sikeresség> ment a <Játékelem neve> elemre.

**GetNeighbours:** <játékelem neve> szomdédjai: <összes szomszédos játékelem neve>

**AddPipe:** <Pumpa neve> aktív elemhez <Pipe neve> cső <sikeresség> csatlakoztatva, pipes : <pumpához csatlakoztatott csövek>

**WaterFlow** Input: <input cső \*forrás esetén null\*> =(<átfolyó víz mennyisége>)=>Aktív elem: <aktív elem neve> =(<átfolyó víz mennyisége>)=> Output: <output cső\*ciszterna esetén null\*>

**AddFreePipe:** <Ciszterna neve>ciszternához <Pipe neve> cső <sikeresség> hozzáadva

**PickUpFreePipe:** <Ciszterna neve>ciszternáról <Pipe neve> cső <sikeresség> felvéve <játékos neve> által

## 7.2 Összes részletes use-case

| **Use-case neve** | **CreatePlumber** |
| --- | --- |
| **Rövid leírás** | Létrehoz egy új szerelőt. |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Létrehoz egy új szerelőt a megadott paraméterekkel |
| **Alternatív forgatókönyv** | **1. A.** Nem jó paraméterek esetén hibát ad vissza a parancs. |

| **Use-case neve** | **CreateSaboteur** |
| --- | --- |
| **Rövid leírás** | Létrehoz egy új szabotőrt. |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Létrehoz egy új szabotőrt a megadott paraméterekkel |
| **Alternatív forgatókönyv** | **1. A.** Nem jó paraméterek esetén hibát ad vissza a parancs. |

| **Use-case neve** | **CreatePump** |
| --- | --- |
| **Rövid leírás** | Létrehoz egy új pumpát |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Létrehoz egy új pumpát a megadott paraméterekkel |
| **Alternatív forgatókönyv** | **1. A.** Nem jó paraméterek esetén hibát ad vissza a parancs. |

| **Use-case neve** | **CreatePipe** |
| --- | --- |
| **Rövid leírás** | Létrehoz egy új csövet. |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Létrehoz egy új csövet a megadott paraméterekkel |
| **Alternatív forgatókönyv** | **1. A.** Nem jó paraméterek esetén hibát ad vissza a parancs. |

| **Use-case neve** | **CreateCistern** |
| --- | --- |
| **Rövid leírás** | Létrehoz egy új ciszternát. |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Létrehoz egy új ciszterna a megadott paraméterekkel |
| **Alternatív forgatókönyv** | **1. A.** Nem jó paraméterek esetén hibát ad vissza a parancs. |

| **Use-case neve** | **CreateSource** |
| --- | --- |
| **Rövid leírás** | Létrehoz egy új forrást. |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Létrehoz egy új forrás a megadott paraméterekkel |
| **Alternatív forgatókönyv** | **1. A.** Nem jó paraméterek esetén hibát ad vissza a parancs. |

| **Use-case neve** | **List** |
| --- | --- |
| **Rövid leírás** | Kilistázza az összes példányát a megadott osztálynak. |
| **Aktorok** | Controller |
| **Forgatókönyv** | 1.Kiírja az összes játékelemet és játékost. |

| **Use-case neve** | **Load** |
| --- | --- |
| **Rövid leírás** | Betölt egy pálya állást a megadott fájlból. |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Betölti a pályát egy fájlból. |
| **Alternatív forgatókönyv** | **1. A.** Nem jó paraméter esetén hibát ad vissza a parancs és nem tölt be semmit. |

| **Use-case neve** | **Save** |
| --- | --- |
| **Rövid leírás** | Elmenti a pálya állását a megadott fájlba. |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Elmenti az aktuális pályát |

| **Use-case neve** | **Points** |
| --- | --- |
| **Rövid leírás** | Kiírja mindkét csapat pontszámát. |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Kiírja a 2 csapat pontszámát. |

| **Use-case neve** | **IncreasePoints** |
| --- | --- |
| **Rövid leírás** | Megnöveli a pontokat a megadott csapatnak. |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Megnöveli a kiválasztott csapatnak a pontszámát. |

| **Use-case neve** | **PickUpPump** |
| --- | --- |
| **Rövid leírás** | Pumpa felvétele a kézbe. |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** Felvesz a megadott ciszternáról egy pumpát, ha még nincs nála. |

| **Use-case neve** | **PutDownPump** |
| --- | --- |
| **Rövid leírás** | Pumpa lerakása a csőre |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** Ha van a szerelőnél pumpa, leteszi a megadott csőre.  **2.** Összeköti a csövet a letett pumpával.  **3.** Létrehoz egy új csövet.  **4.** Összeköti az új csövet a régi cső eredeti végpontjával és a letett pumpával.  **5.** A játékos a régi csövön marad. |

| **Use-case neve** | **FixGameElement** |
| --- | --- |
| **Rövid leírás** | Javítható elem javítása |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** Megjavítja az elromlott játékelemet a szerelő, ha az el volt romolva.  **2.** Ha csövet javít meg, beállítja hogy valamennyi ideig ne lehessen újra kilyukasztani. |

| **Use-case neve** | **MakeSlippery** |
| --- | --- |
| **Rövid leírás** | A cső változtatása, hogy csúszós legyen |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** A játékos csúszóssá teszi a kiválasztott csövet |
| **Alternatív forgatókönyv** | **1. A. 1.** Ha a kiválasztott cső ragadós, akkor a csövet nem lehet csúszóssá tenni. |

| **Use-case neve** | **MakeSticky** |
| --- | --- |
| **Rövid leírás** | A cső változtatása, hogy ragadós legyen |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** A játékos ragadóssá teszi a kiválasztott csövet |

| **Use-case neve** | **Sabotage** |
| --- | --- |
| **Rövid leírás** | Cső elrontása. |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** Kilyukasztja a kiválasztott csövet. |
| **Alternatív forgatókönyv** | **1. A. 1.** A játékos nem tudja kilyukasztani a kiválasztott csövet, ha a cső lyukasztás időzítője nem nulla. |

| **Use-case neve** | **PickUpPipeEnd** |
| --- | --- |
| **Rövid leírás** | Csővég felvétele a kézbe. |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** A játékos felveszi a kiválasztott csövet  **2.** A cső vége lecsatlakozik az elemről, ahol a játékos van |
| **Alternatív forgatókönyv** | **1. A. 1.** Ha játékosnak teli van a keze, akkor nem tud felvenni több csövet. |

| **Use-case neve** | **PutDownPipeEnd** |
| --- | --- |
| **Rövid leírás** | Cső vég lerakása és csatlakoztatása az elemhez |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** A játékos lerakja a kezében tartott csővéget, ahol áll éppen.  **2.** Ahol áll hozzáadótik a kimeneteihez a játékos által letett cső.  **3.** A letett cső végpontjaihoz hozzáadódik ahol áll a játékos. |
| **Alternatív forgatókönyv** | **1. A. 1.** Ha csövön vagy ciszternán áll a játékos vagy nincs már hely a játékelemen a csőnek ahol áll nem tudja letenni. |

| **Use-case neve** | **ChangePumpInput** |
| --- | --- |
| **Rövid leírás** | Pumpa bemenetének változtatása |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** Átállítjuk a kiválasztott pumpának a kimenetét a kiválasztott csőre. |
| **Alternatív forgatókönyv** | **1. A. 1.** Ha a kiválasztott cső a pumpa bemenete, akkor a cső nem állítódik át. |

| **Use-case neve** | **ChangePumpOutput** |
| --- | --- |
| **Rövid leírás** | Pumpa kimenetének változtatása |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** Átállítjuk a kiválasztott pumpának a bemenetét a kiválasztott csőre. |
| **Alternatív forgatókönyv** | **1. A. 1.** Ha a kiválasztott cső a pumpa kimenete, akkor a cső nem állítódik át. |

| **Use-case neve** | **Move** |
| --- | --- |
| **Rövid leírás** | Játékos mozgatása. |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** A játékos ellép a mostani elemről.  **2.** A játékos átlép a kiválasztott elemre. |
| **Alternatív forgatókönyv** | **2. A. 1.** Ha csövön áll a játékos, ami csúszós, amire lép, akkor véletlenszerűen átkerül a cső valamelyik végére. |
| **Alternatív forgatókönyv** | **1. A. 1.** Ha egy csövön áll a játékos, ami ragadós nem tud továbblépni. |
| **Alternatív forgatókönyv** | **1. B. 1.** Ha a játékos egy csőre akar lépni, amin már áll egy másik játékos vagy le van csatolva a másik vége nem tud továbblépni. |

| **Use-case neve** | **GetNeighbours** |
| --- | --- |
| **Rövid leírás** | Kiírja az összes szomszédos játékelemet . |
| **Aktorok** | Controller |
| **Forgatókönyv** | **1.** Kírja kiválasztott elem összes szomszédját. |

| **Use-case neve** | **MoveSlippery** |
| --- | --- |
| **Rövid leírás** | Cső véletlenszerű végére mozgatja a játékost |
| **Aktorok** | Player |
| **Forgatókönyv** | **1.** Cső véletlenszerű végére mozgatja a játékost. |

## 7.3 Tesztelési terv

7.3.1

| **Teszt-eset neve** | Pálya betöltése |
| --- | --- |
| **Rövid leírás** | Betöltjük a pályát a megadott fájlból. |
| **Teszt célja** | Ellenőrizzük, hogy az objektumok és a köztük lévő kapcsolatok helyesen létrejönnek-e. |

7.3.2

| **Teszt-eset neve** | Játékos mozgatása aktív elemről csőre, ha a cső nem foglalt. |
| --- | --- |
| **Rövid leírás** | Egy játékos egy aktív elemen áll. Átmozgatjuk egy, a tartózkodási helyéhez csatlakozó csőre. Ez a cső nem foglalt, ekkor azt várjuk, hogy a játékosunk átkerül a csőre. |
| **Teszt célja** | Megnézzük, hogy játékos át tud-e lépni egy üres csőre. |

7.3.3

| **Teszt-eset neve** | Játékos mozgatása aktív elemről csőre, ha a cső foglalt. |
| --- | --- |
| **Rövid leírás** | Egy játékos egy aktív elemen áll. Átmozgatjuk egy, a tartózkodási helyéhez csatlakozó csőre. Ez a cső foglalt, ekkor azt várjuk, hogy a játékosunk az aktív elemen marad. |
| **Teszt célja** | Leellenőrizzük, hogy egy csövön egyszerre csak játékos tud-e tartózkodni. |

7.3.4

| **Teszt-eset neve** | Játékos mozgatása aktív elemről csőre, ha a cső nem foglalt és csúszós. |
| --- | --- |
| **Rövid leírás** | Egy játékos egy aktív elemen áll. Átmozgatjuk egy, a tartózkodási helyéhez csatlakozó csőre. Ez a cső nem foglalt, viszont csúszós. Ekkor azt várjuk, hogy a játékosunk vagy az eredeti helyén marad vagy a cső túlsó végére kerül. |
| **Teszt célja** | Ellenőrizzük, hogy ha egy cső csúszós, akkor nem tudunk rálépni, helyette valamelyik végpontra kerülünk. |

7.3.5

| **Teszt-eset neve** | Játékos mozgatása csőről aktív elemre, úgy hogy az aktív elemen áll egy másik játékos. |
| --- | --- |
| **Rövid leírás** | A játékosunk egy csövön áll, egy csőhöz csatlakozó aktív elemre próbál lépni, amelyen már áll egy másik játékos. Ekkor azt várjuk el, hogy a játékosunk az aktív elemre kerül. |
| **Teszt célja** | Ellenőrizzük, hogy egy aktív elemen egyszerre több játékos is tartózkodhat. |

7.3.6

| **Teszt-eset neve** | Játékos mozgatása csőről aktív elemre, ha cső ragadós.. |
| --- | --- |
| **Rövid leírás** | Egy játékos egy csövön áll. Megpróbáljuk átmozgatni egy, a tartózkodási helyéhez csatlakozó aktív elemre. A cső ragadós. Ekkor azt várjuk, hogy a játékosunk nem tud átlépni az aktív elemre, a csövön marad. |
| **Teszt célja** | Ellenőrizzük, hogy ha egy cső ragadós, akkor nem tudunk elmenni róla. |

7.3.7

| **Teszt-eset neve** | Játékos mozgatása aktív elemről olyan csőre, amelynek egyik vége le van csatolva. |
| --- | --- |
| **Rövid leírás** | Egy játékos egy aktív elemen áll. Átmozgatjuk egy, a tartózkodási helyéhez csatlakozó csőre. Ez a cső nem foglalt, viszont a másik vége le van csatlakoztatva. Ekkor azt várjuk, hogy a játékosunk az eredeti helyén marad. |
| **Teszt célja** | Ellenőrizzük, hogy ha egy cső egyik vége le van csatolva, akkor nem tudunk rálépni. |

7.3.8

| **Teszt-eset neve** | Víz átjuttatása forrásból a hozzá csatlakozó nem lyukas, üres csőbe, amelynek másik vége is csatlakoztatva van. |
| --- | --- |
| **Rövid leírás** | A forrás teletölti a hozzá csatlakozó csövet, amelynek vízszintje a maximumra emelkedik. |
| **Teszt célja** | Ellenőrizzük, hogy egy mindkét végén csatlakoztatott, nem lyukas cső nem folyatja ki magából a vizet. |

7.3.9

| **Teszt-eset neve** | Víz átjuttatása forrásból a hozzá csatlakozó nem lyukas, üres csőbe, amelynek másik vége nincs csatlakoztatva. |
| --- | --- |
| **Rövid leírás** | A forrás vizet tölt a hozzá csatlakozó csőbe, amelyből az összes víz a sivatagba folyik. |
| **Teszt célja** | Ellenőrizzük, hogy egy egyik végén nem csatlakoztatott, nem lyukas csőből a sivatagba folyik a víz. |

7.3.10

| **Teszt-eset neve** | Víz átjuttatása forrásból a hozzá csatlakozó lyukas, üres csőbe, amelynek másik vége is csatlakoztatva van. |
| --- | --- |
| **Rövid leírás** | A forrás vizet tölt a hozzá csatlakozó csőbe, amelyből az összes víz a sivatagba folyik. |
| **Teszt célja** | Ellenőrizzük, hogy egy mindkét végén csatlakoztatott, lyukas csőből a sivatagba folyik a víz. |

7.3.11

| **Teszt-eset neve** | Víz átjuttatása forrásból a hozzá csatlakozó nem lyukas, teli csőbe, amelynek másik vége is csatlakoztatva van. |
| --- | --- |
| **Rövid leírás** | A forrás nem tölt több vizet a hozzá csatlakozó csőbe. |
| **Teszt célja** | Ellenőrizzük, hogy egy mindkét végén csatlakoztatott, nem lyukas teli csőbe nem tudunk több vizet tölteni. |

7.3.12

| **Teszt-eset neve** | Víz átjuttatása ciszternába egy hozzá csatlakozó csőből. |
| --- | --- |
| **Rövid leírás** | A ciszterna az összes vizet kipumpálja a csőből. |
| **Teszt célja** | Ellenőrizzük, hogy egy ciszterna ki tudja pumpálni a vizet egy csőből. |

7.3.13

| **Teszt-eset neve** | Nem csúszós, nem ragadós cső lyukasztása |
| --- | --- |
| **Rövid leírás** | A cső nem csúszós és nem is ragadós, úgymond megfelelő állapotban van. Egy játékos rálép erre a csőre, majd kilyukasztja. |
| **Teszt célja** | Ellenőrizzük, hogy egy megfelelő állapotú csövet ki tud lyukasztani egy szabotőr és egy szerelő is. |

7.3.14

| **Teszt-eset neve** | Ragadós cső lyukasztása |
| --- | --- |
| **Rövid leírás** | A csövet beállítottuk ragadós állapotúra. Egy játékos rálép erre a csőre, majd kilyukasztja. |
| **Teszt célja** | Ellenőrizzük, hogy egy szabotőr és egy szerelő akkor is ki tudja lyukasztani a csövet, ha az ragadós állapotú. |

7.3.15

| **Teszt-eset neve** | Lyukas cső lyukasztása |
| --- | --- |
| **Rövid leírás** | A csövet már korábban kilyukasztottuk. Egy játékos rálép erre a csőre, majd meg próbálja kilyukasztja. |
| **Teszt célja** | Ellenőrizzük, hogy egy szabotőr és egy szerelő ha megpróbál kilyukasztani egy csövet, akkor nem történik semmilyen változás. |

7.3.16

| **Teszt-eset neve** | Foltozott cső lyukasztása a megadott időn belül |
| --- | --- |
| **Rövid leírás** | A csövet már korábban kilyukasztottuk és azóta meg is javítottuk. Ámbár az az idő még nem járt le, ami után újra lyukaszthatunk. Egy játékos rálép erre a csőre, majd megpróbálja kilyukasztani. |
| **Teszt célja** | Ellenőrizzük, hogy ha egy szabotőr vagy egy szerelő ha megpróbál kilyukasztani egy csövet, azon az időn belül, amikor még nem lehetséges, akkor az nem sikerül neki. |

7.3.17

| **Teszt-eset neve** | Megfelelő állapotban lévő pumpa elrontása |
| --- | --- |
| **Rövid leírás** | Egy megfelelően működő pumpát elrontunk, szabotálunk. |
| **Teszt célja** | Ellenőrizzük, hogy az elromlott pumpa nem továbbítja a vizet. |

7.3.18

| **Teszt-eset neve** | Hibás pumpa elrontása |
| --- | --- |
| **Rövid leírás** | Egy működő pumpa már el van romolva és újra megpróbáljuk elrontani. |
| **Teszt célja** | Ellenőrizzük, hogy ha egy pumpa már el van romolva, akkor ha még egyszer el akarjuk rontani, akkor ne történjen változás. |

7.3.19

| **Teszt-eset neve** | Lyukas cső megjavítása |
| --- | --- |
| **Rövid leírás** | Egy csövet kiválasztunk, amit kilyukasztunk. Majd ezt a csövet megjavítjuk. |
| **Teszt célja** | Ellenőrizzük, hogy egy megjavított cső helyesen továbbítja a vizet. |

7.3.20

| **Teszt-eset neve** | Jó állapotban lévő cső megjavítása |
| --- | --- |
| **Rövid leírás** | Egy csövet kiválasztunk, ami tökéletesen működik. Majd ezt a csövet megpróbáljuk megjavítani. |
| **Teszt célja** | Ellenőrizzük, hogy ha egy jó állapotban lévő csövet akarunk megjavítani, akkor az továbbra is jól működik. |

7.3.21

| **Teszt-eset neve** | Hibás pumpa megjavítása |
| --- | --- |
| **Rövid leírás** | Egy pumpát kiválasztunk, amit elrontunk. Majd ezt a pumpát megjavítjuk. |
| **Teszt célja** | Ellenőrizzük, hogy egy megjavított pumpa megfelelően működjön. |

7.3.22

| **Teszt-eset neve** | Jó állapotban lévő pumpa megjavítása |
| --- | --- |
| **Rövid leírás** | Egy pumpát kiválasztunk, ami tökéletesen működik. Majd ezt a pumpát megpróbáljuk megjavítani. |
| **Teszt célja** | Ellenőrizzük, hogy ha egy jó állapotban lévő pumpát akarunk megjavítani, akkor az továbbra is jól működik. |

7.3.23

| **Teszt-eset neve** | Működő pumpa vizet pumpál, a bemeneti cső nem üres. A pumpa több vizet tartalmaz, mint a kimeneti cső kapacitása. |
| --- | --- |
| **Rövid leírás** | A pumpa a bementi csőből a tartályába pumpálja a vizet, onnan pedig a kimeneti csőbe. A kimeneti csövet a maximum szintig tölti. |
| **Teszt célja** | Ellenőrizzük, pumpa csak annyi vizet pumpál-e a kimeneti csőbe, mint amennyi belefér. |

7.3.24

| **Teszt-eset neve** | Működő pumpa vizet pumpál, a bemeneti cső üres. |
| --- | --- |
| **Rövid leírás** | A pumpa a saját tartályából pumpál vizet a kimenetbe, amíg ki nem ürül. |
| **Teszt célja** | Ellenőrizzük, hogy a pumpa akkor is képes vizet továbbítani a kimeneti csőbe, ha a bemeneti cső üres. |

7.3.25

| **Teszt-eset neve** | Működő pumpa vizet pumpál, a bemeneti cső nem üres. A pumpa kevesebb vizet tartalmaz, mint a kimeneti cső kapacitása. |
| --- | --- |
| **Rövid leírás** | A pumpa a bemeneti csőből a tartályába pumpálja a vizet, a tartályban még így is kevesebb víz van, mint amit a kimenet fogadni tudna. Ekkor a pumpa a tartályából az összes vizet a kimenetbe juttatja. |
| **Teszt célja** | Ellenőrizzük, hogy a pumpa mindig a lehető legtöbb vizet juttatja a kimeneti csőbe. |

7.3.26

| **Teszt-eset neve** | Működő pumpa vizet pumpál, a bemeneti cső és a pumpa tartálya is üres. |
| --- | --- |
| **Rövid leírás** | A pumpa a saját tartályából próbál vizet pumpálni a kimenetbe, azonban ez üres. A bemeneti csőből sem tudja feltölteni a tartályát ugyanis ez is üres, ekkor nem kerül víz a kimeneti csőbe. |
| **Teszt célja** | Ellenőrizzük, hogy ha pumpa tartálya és a bemeneti cső is üres, akkor nem kerül víz a kimenetbe. |

7.3.27

| **Teszt-eset neve** | Pumpa vizet pumpál, a bemeneti csőben több víz van, mint ami a pumpa tartályába fér. |
| --- | --- |
| **Rövid leírás** | A pumpa csak annyi vizet vesz fel a bemeneti csőből, amennyi a tartályába fér. |
| **Teszt célja** | Ellenőrizzük, hogy a pumpa csak annyi vizet vesz fel a bemeneti csőből, mint amennyit képes eltárolni. |

7.3.28

| **Teszt-eset neve** | Pumpa vizet pumpál, a kimeneti cső tele van. |
| --- | --- |
| **Rövid leírás** | A pumpa nem továbbít vizet a kimeneti csőbe. |
| **Teszt célja** | Ellenőrizzük, hogy a pumpa nem továbbít vizet olyan csőbe, amelyik nem képes több vizet fogadni. |

7.3.29

| **Teszt-eset neve** | Pumpa nem működik. |
| --- | --- |
| **Rövid leírás** | A pumpa elromlott ekkor, nem tölt vizet a tartályába a bementi csőből és nem tölt a kimeneti csőbe se vizet. |
| **Teszt célja** | Ellenőrizzük, hogy a nem működő pumpa nem továbbít vizet. |

7.3.30

| **Teszt-eset neve** | Lyukas csőbe pumpálunk vizet. |
| --- | --- |
| **Rövid leírás** | Egy olyan csőbe juttatunk vizet, amely ki van lyukasztva, ekkor a csőből az összes víz a sivatagba folyik és minden egység sivatagba ömlött víz után 1 pontot kapnak a szabotőrök. |
| **Teszt célja** | Ellenőrizzük, hogy a lyukas csőből kifolyik a víz és kifolyt víz után a szabotőrök pontot kapnak. |

7.3.31

| **Teszt-eset neve** | Olyan csőbe juttatunk vizet, amelynek a másik vége le van csatolva. |
| --- | --- |
| **Rövid leírás** | Egy olyan csőbe juttatunk vizet, amelynek másik vége le van csatolva, ekkor a csőből az összes víz a sivatagba folyik és minden egység sivatagba ömlött víz után 1 pontot kapnak a szabotőrök. |
| **Teszt célja** | Ellenőrizzük, hogy a lecsatolt végű csőből kifolyik a víz és kifolyt víz után a szabotőrök pontot kapnak. |

7.3.32

| **Teszt-eset neve** | Cső lecsatlakoztatása, a cső nem be- vagy kimeneti cső |
| --- | --- |
| **Rövid leírás** | A játékos egy pumpán áll és a pumpához csatlakozó csövek közül lecsatol egyet. A lecsatolandó cső nem be- vagy kimenete a pumpának ekkor a játékoshoz kerül a cső vége és ezt magával tudja cipelni. |
| **Teszt célja** | Ellenőrizzük, hogy le tudunk-e csatolni egy csövet. |

7.3.33

| **Teszt-eset neve** | Cső lecsatlakoztatása, a cső be- vagy kimeneti cső |
| --- | --- |
| **Rövid leírás** | A játékos egy pumpán áll és a pumpához csatlakozó csövek közül lecsatol egyet. A lecsatolandó cső be- vagy kimenete a pumpának ekkor a játékos nem tudja lecsatolni a csövet. |
| **Teszt célja** | Ellenőrizzük, hogy be- vagy kimeneti csövet nem tudunk lecsatolni. |

7.3.34

| **Teszt-eset neve** | Pumpa lehelyezése csőre, a cső nem lyukas, a játékosnál van pumpa. |
| --- | --- |
| **Rövid leírás** | A játékos lehelyez egy pumpát arra a csőre, amin éppen áll. Ekkor keletkezik egy új cső amely a lehelyezett pumpa és a cső egyik végpontját köti össze, a régi cső pedig a lehelyezett pumpát és a másik végpontot köti össze. |
| **Teszt célja** | Ellenőrizzük, hogy le tudunk-e helyezni egy pumpát a nem lyukas csőre és lehelyezés után a lehelyezett és a régi pumpák megfelelően vannak összekötve. |

7.3.34

| **Teszt-eset neve** | Pumpa lehelyezése csőre, a cső lyukas, a szerelőnél van pumpa. |
| --- | --- |
| **Rövid leírás** | A szerelő megpróbál lehelyezni egy pumpát arra a csőre, amelyen éppen áll, azonban a cső lyukas. Ekkor a szerelő nem tudja lehelyezni a pumpát. |
| **Teszt célja** | Ellenőrizzük, hogy lyukas csőre nem tudunk pumpát lehelyezni. |

7.3.35

| **Teszt-eset neve** | Pumpa lehelyezése csőre, a cső nem lyukas, a szerelőnél nincs pumpa. |
| --- | --- |
| **Rövid leírás** | A szerelő megpróbál lehelyezni egy pumpát arra a csőre, amelyen éppen áll, azonban a nincs nála pumpa. Ekkor a szerelő nem tud pumpát lehelyezni. |
| **Teszt célja** | Ellenőrizzük, hogy csak akkor tud egy szerelő pumpát lehelyezni, ha van nála pumpa. |

7.3.36

| **Teszt-eset neve** | Pumpa felvétele ciszternánál, a szerelőnél nincsen pumpa |
| --- | --- |
| **Rövid leírás** | A szerelő a ciszternán áll és nincs nála pumpa. Megpróbál felvenni egy pumpát és sikeres lesz a művelet. |
| **Teszt célja** | Ellenőrizzük, játékos tud-e felvenni pumpát. |

7.3.37

| **Teszt-eset neve** | Pumpa felvétele ciszternánál, a szerelőnél már van egy pumpa |
| --- | --- |
| **Rövid leírás** | A szerelő a ciszternán áll és van nála pumpa. Megpróbál felvenni egy pumpát és sikertelen lesz a művelet. |
| **Teszt célja** | Ellenőrizzük,szerelő egyszerre csak egy pumpát tud magánál tartani. |

7.3.38

| **Teszt-eset neve** | Szabad végű cső létrehozása ciszternánál. |
| --- | --- |
| **Rövid leírás** | A ciszternán keletkezik egy új cső, amelynek egyik vége a ciszternához csatlakozik, a másik vége szabad. A szabad véget a játékosok fel tudják venni ha a ciszternára lépnek. |
| **Teszt célja** | Ellenőrizzük, hogy létrejön-e a cső a ciszternán és az egyik vége szabad-e, továbbá a másik tényleg a ciszternához csatlakozik-e. |

7.3.39

| **Teszt-eset neve** | Szabad csővég felvétele ciszternánál, a játékosnál már van egy csővég |
| --- | --- |
| **Rövid leírás** | A játékos a ciszternán áll és van nála csővég. Megpróbál felvenni egy szabad csővéget és sikertelen lesz a művelet. |
| **Teszt célja** | Ellenőrizzük, játékos egyszerre csak egy csővéget tud magánál tartani. |

7.3.40

| **Teszt-eset neve** | Csővég felvétele ciszternánál, a játékosnál nincsen csővég |
| --- | --- |
| **Rövid leírás** | A játékos a ciszternán áll és nincs nála csővég. Megpróbál felvenni egy pumpát és sikeres lesz a művelet. |
| **Teszt célja** | Ellenőrizzük, játékos tud-e felvenni felvenni szabad csővéget a ciszternáról.. |

7.3.41

| **Teszt-eset neve** | Pumpa kimenetének állítása |
| --- | --- |
| **Rövid leírás** | A játékos egy pumpán áll. A csatlakozó csövek közül egyet beállít kimenetként. Ezt követően a pumpa az új kimenetbe továbbítja a vizet. |
| **Teszt célja** | Ellenőrizzük,hogy át tudjuk-e állítani a pumpa kimenetét. |

7.3.42

| **Teszt-eset neve** | Pumpa bemenetének állítása |
| --- | --- |
| **Rövid leírás** | A játékos egy pumpán áll. A csatlakozó csövek közül egyet beállít bemenetként. Ezt követően a pumpa az új bemenetből szív ki vizet. |
| **Teszt célja** | Ellenőrizzük,hogy át tudjuk-e állítani a pumpa bemenetét. |

## 7.4 Tesztelést támogató segéd- és fordítóprogramok specifikálása

A prototípust a standard bemenetről a 7.1.2-ben definiált bemeneti nyelv segítségével lehet, soronként egy parancs beírásával, az elvárt kimeneteket a különböző állapotlekérdező parancsok segítségével tudja a felhasználó ellenőrizni. Fájlból olvasáshoz 7.1.2 végén definiált struktúrában kell a fájlokat elkészíteni, hogy a prototípus fel tudja dolgozni.  
A kimenet standard kimenetre, és a fájlba is a 7.1.3-ban definiált módon fog kiírni, tehát minden bemeneti sorhoz egy kimeneti sor fog tartozni.

## 7.5 Napló

| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| --- | --- | --- | --- |
| 2023.04.19. 18:30 | 3 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet.  Döntés: Sütheö elkészíti a módosult osztálydiagramot, cső állapotgráfját, valamint a új vagy megváltozott metódusokat 04.20. 19:00 óráig. Gerencsér elkészíti a módosult szekvenciadiagramokat 04.21. 15:00 óráig. Chu elkészíti prototípus interface-defínicióját 04.22. 15:00 óráig. Bognár elkészíti a részletes use-case leírásokat 04.23. 10:00 óráig. Vörös elkészíti a tesztelési tervet 04.23. 21:59 -ig. |
| 2023.04.20. 15:00 | 3 óra | Sütheö | Tevékenység: Sütheö elkészíti a módosult osztálydiagramot, a cső állapotgráfját, valamint az új vagy megváltozott metódusokat. (7.0.0, 7.0.1, 7.0.2) |
| 2023.04.21. 9:00 | 3,5 óra | Gerencsér | Tevékenység: Gerencsér elkészíti a módosult vagy új szekvenciadiagramokat. (7.0.3.1-7.0.3.7) |
| 2023.04.21. 16:15 | 2 óra | Chu | Tevékenység: Chu elkészíti az interfész általános leírását, valamint a bemeneti nyelvet (7.1.1, 7.1.2) |
| 2023.04.22. 10:00 | 1,5 óra | Chu | Tevékenység: Chu elkészíti az interfész kimeneti nyelvét. (7.1.3) |
| 2023.04.22. 16:45 | 3 óra | Bognár | Tevékenység: Bognár elkészíti a részletes use-case leírásokat (7.2) |
| 2023.04.23. 11:00 | 3 óra | Vörös | Tevékenység: Vörös elkészíti a tesztelési terveket, tehát a tesztelési eseteket felveszi. (7.3) |
| 2023.04.23. 22:30 | 1 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet, a dokumentum átnézése.  Döntés: Sütheö elkészíti a tesztelést támogató segéd- és fordító programok leírását 04.24. 10:00 óráig. |
| 2023.04.24. 9:00 | 0,5 óra | Sütheö | Tevékenység: Sütheö elkészíti a tesztelést végző segédprogramok leírását. (7.4) |

# 

# 8. Részletes tervek

## 8.1 Osztályok és metódusok tervei.

**8.1.1 Cistern**

#### · Felelősség

A pálya végén helyezkedik el. Csak bemenetei lehetnek, amikből a játék során folyamatosan készülnek újak, szabad végű csövek. Az ide bemenő összes vizet elnyeli és ez alapján szereznek pontokat a szerelők. Innen tudnak a szerelők új pumpákat felvenni a csőrendszer bővítéséhez.

#### · Ősosztályok

Element

#### · Interfészek

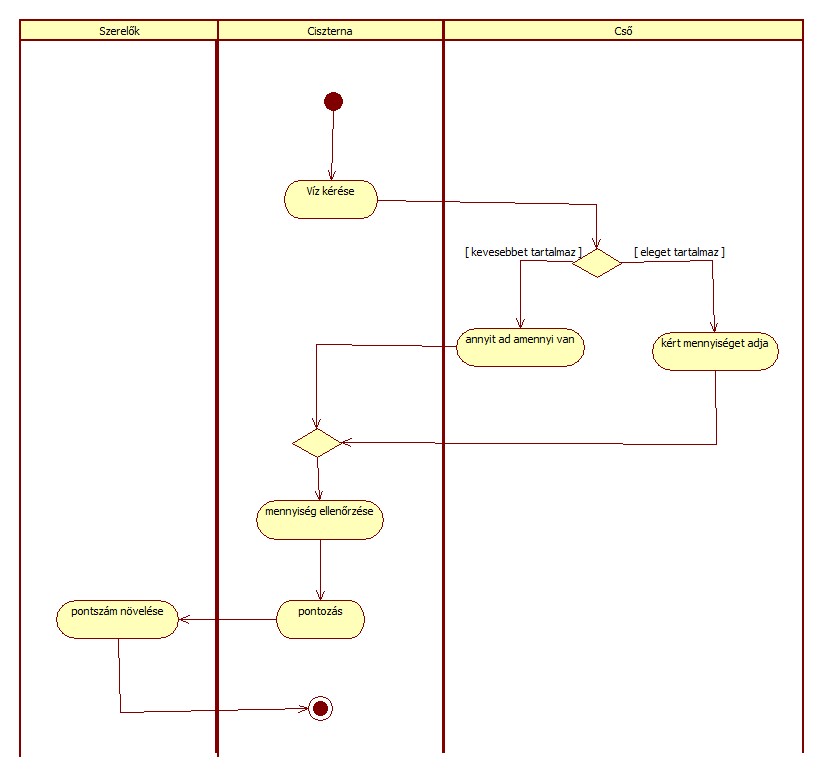
#### PointCounter

#### · Attribútumok

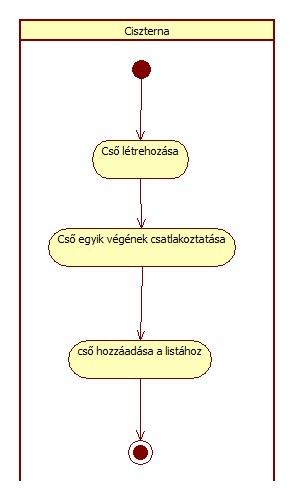
* **-bool hasPump:** A ciszternában van a felvehető pumpa.
* **-int numberOfFreePipes:** Hány szabad végű cső van még a ciszternában.
* **Pump pump:** A tárolt pumpa, amit fel tudnak venni a szerelők.
* **-List<Pipe> createdPipes:** A szabad csöveket tárolja, amik a ciszternában jönnek létre, a létrejövés idejének sorrendjében.

#### · Metódusok

* **+void WaterFlow():** Minden a pipes változójában lévő csőre meghívja az AbsorbWater függvényt 10 értékű paraméterrel, mivel ez a csövek kapacitása, ennél több nem jöhet belőle. Ez a függvény visszaad minden csőnél egy értéket, amennyivel növelni kell a szerelők pontját és meghívódik az IncreasePoints függvény az előző visszatérési értékkel.



* **+Pump GivePump():** Visszaadja a pump változóját és új pumpát hoz létre annak a pump változóba.
* **+void IncreasePoints(int amount):** Megnöveli a szerelők pontját amount értékkel.
* **+void CreateNewPipe():** Létrehoz egy új csövet, majd meghívja az AddEndpoint függvényét saját magával paraméterként, és ezzel hozzácsatlakoztatja a csövet magához. Utána hozzáadja a createdPipes változó legvégére a létrehozott csövet az időrendi sorrend megtartásához. Ezt az egészet egy try-catchben kell megvalósítani, mivel a cső csatlakozása dobhat kivételt, ha már mindkét vége volt csatlakoztatva a csőnek.



* **+Pipe GetFreePipe():** Visszaadja a createdPipes legelső elemét, ami a legrégebben jött létre és kitörli azt a listából.

**8.1.2 Desert**

#### · Felelősség

A pályát reprezentálja. A játékelemeket és csöveket tárolja el, amiből az egész pályát fel lehet építeni és a forrást, amiből a vízfolyást kell futtatni. Ide folyik el a víz a kilyukadt csövekből. Az elfolyt víz alapján ad pontokat a szabotőröknek.

#### · Ősosztályok

-

#### · Interfészek

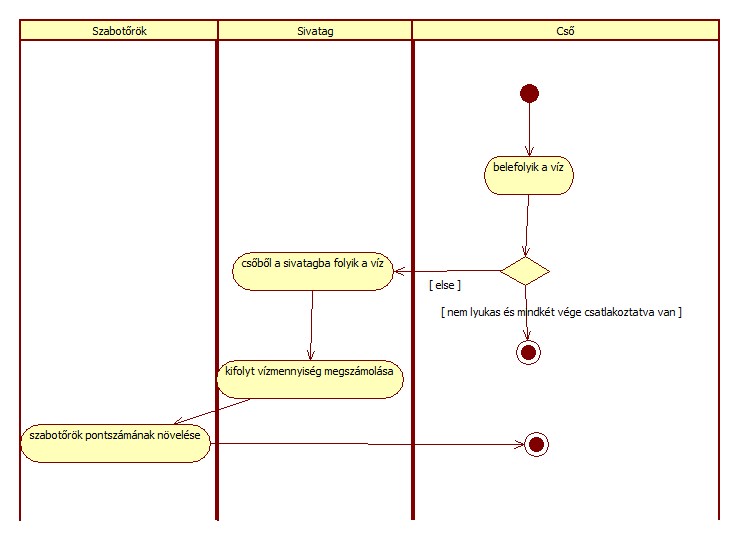
PointCounter

#### · Attribútumok

* **-Source source:** A pálya forrása.
* **-List<Element> element:** A játékelemek.
* **-List<Pipe> pipes:** A csövek.

#### · Metódusok

* **+GetWater(int amount):** Megkapja az elfolyt vizet a csövekből paraméterként és meghívja az IncreasePoints függvényt az amount étrékkel.



* **+void IncreasePoints(int amount):** Növeli a szabotőrök pontját amount értékkel.

**8.1.3 Element**

#### · Felelősség

A játékban szereplő és azokkal interaktálható elemek. Ezekből épül fel a csőrendszer. Ezeken tudnak a játékosok mozogni és tudnak interaktálni velük. Ezenfelül a játék két fő csomópontja a ciszterna, valamint a forrás is ebből származik le. A víz szállítására, valamint a víz forrásaként/céljaként szolgálnak a játék során.

#### · Ősosztályok

-

#### · Interfészek

WaterFlow, Moveable

#### · Attribútumok

* **-int maxPipes:** Az elemhez maximálisan csatlakoztatható csövek száma.
* **-List<Pipe> pipes:** Az elemhez csatlakozó csövek, csatlakoztatás időbeli sorrendjében.

#### · Metódusok

* **+void WaterFlow():** Absztrakt függvény, amelyet a leszármazott játékelemek definiálnak. A víz folyását szimuláljuk ezzel a függvénnyel. Az egyes elemek esetében más-más a hatása.
* **+MoveAway():** Nem csinál semmit csak az interface miatt kell megvalósítani.
* **+Moveable MoveHere(Moveable from):** Ha a from paraméter benne van a pipes listában, vagyis össze van kötve ezzel a hellyel ahonnan ide akar mozogni a játékos, meghívja a from MoveAway függvényét és visszatér a this értékével. Különben kivételt dob “Nem lehet ide lépni” üzenettel.
* **+void RemovePipe(Pipe pipe):** Kitörli a pipes változóból a megadott pipe csövet, ezzel lecsatlakoztatva azt.
* **+void AddPipe(Pipe pipe):** Hozzáadja a pipes változóhol a megadott pipe csövet a lista végére, ezzel felcsatlakoztatva azt és csatlakozás szerint időrendi sorrendben tartva a listát, ha még van hely a játékelemen. Különben kivételt dob “Nincs hely a csőnek” üzenettel.

**8.1.4 Fixable**

#### · Felelősség

Interface. Megvalósítja az elromolható/javítható funkciókat.

#### · Ősosztályok

-

#### · Interfészek

-

#### · Attribútumok

-

#### · Metódusok

* **+void Fix():** Megjavítja az elromlott elemet.
* **+void Disable():** Elrontja a működő elemet.

**8.1.5 Game**

#### · Felelősség

Játékosokat ad hozzá a játékhoz és tárol. Elindítja a játékot. Minimum 4 ember kell a játék elindításához. Leállítja a játékot egy bizonyos idő után. Számon tartja a csapatok pontszámait, eldönti ki nyert a játék végén.Méri a játékidőt.

#### · Ősosztályok

-

#### · Interfészek

-

#### · Attribútumok

* **-List<Saboteur> saboteurs:** A játékban lévő szabotőrök listája belépési sorrendben.
* **-List<Plumber> plumbers:** A játékban lévő szerelők listája belépési sorrendben.

#### · Metódusok

* **+void JoinSaboteurs():** Játékost ad hozzá a szabotőr csapathoz.
* **+void JoinPlumbers():** Játékost ad hozzá a szerelő csapathoz.
* **+void StartGame():** Ha van elegendő játékos elindítja a játékot.
* **+void EndGame():** Leállítja a játékot. Kiírja melyik csapat nyert.
* **+int SaboteurCount():** Visszaadja, hogy mennyi szabotőr van a játékban.
* **+int PlumberCount():** Visszaadja, hogy mennyi szerelő van a játékban.

**8.1.6 Moveable**

#### · Felelősség

Interface. Azok a játékelemek, amikre a játékosok rá tudnak lépni.

#### · Ősosztályok

-

#### · Interfészek

-

#### · Attribútumok

-

#### · Metódusok

* **+Moveable MoveHere(Moveable from):** Visszaadja a helyet, hogyha tud idemozogni a játékos.
* **+void MoveAway():** Elmozgatja a játékost az elemről.

**8.1.7 Pipe**

#### · Felelősség

A vizet szállítja. Egy adott csövön, egy bizonyos mennyiségű víz folyhat egyszerre, ez a cső kapacitása. A szabadvégű csövek végén a víz elfolyik a homokba. Ha a csövön egy lyuk keletkezik, akkor a cső kimenetére már nem jut el a víz, a betáplált víz elfolyik a sivatagba.

#### · Ősosztályok

-

#### · Interfészek

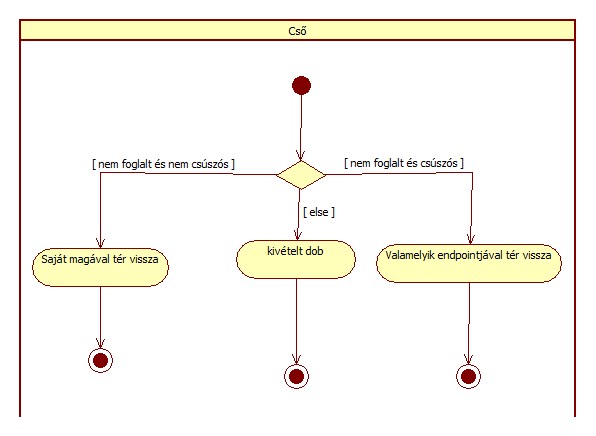
Moveable, Fixable

#### · Attribútumok

* **-int waterLevel:** Az aktuális vízmennyiséget mutatja a csőben.
* **-bool working:** A cső működik vagy nem.
* **-int capacity:** Az adott cső kapacitása, amennyi víz maximálisan lehet benne egyszerre.
* **-bool occupied:** Vannak-e a csövön éppen.
* **+bool pickedUp:** Fel van-e véve legalább egy vége a csőnek.
* **-int sabotageable:** Ha nulla az értéke ki lehet lyukasztani a csövet. Különben megadja mennyi idő van még, ameddig újra kilyukasztható lesz a cső.
* **-int sticky:** Ha nulla az értéke nem ragad a cső. Különben megadja mennyi ideig ragad még.
* **-int slippery:** Ha nulla az értéke nem csúszós a cső. Különben megadja mennyi ideig csúszós még.
* **-List<Element> endpoints:** Egy maximum két elemű lista, amiben a cső két végpontja van tárolva.

#### · Metódusok

* **+void Fix():** Igazra állítja a working változót és átállítja a sabotageable változó értékét véletlen számra.
* **+void Disable():** Hamisra állítja a working változót.
* **+void MoveAway():** Hamisra állítja az occupied változót.
* **+Moveable MoveHere(Moveable from):** Ha az occupied változó hamis és a from paraméter benne van az endpoints listában, igazra állítja az occupied-ot, meghívja a from MoveAway függvényét és visszatér a this értékével. Ha az occupied változó hamis és a slippery tagváltozó értéke 0-nál nagyobb, akkor valamelyik endpointjával tér vissza. Különben kivételt dob “Nem lehet ide lépni” üzenettel.



* **+void AddEndPoint(Element element):** Ha mindkét endpoint null, az elsőt átírja az element-re. Hogyha csak az egyik null akkor ezt átírja az e-re. Hogyha egyiksem null, akkor kivételt dob “Nincs szabad vége a csőnek” üzenettel.
* **+Element GetEndPoint():** Ha az egyik endpoint sem null, visszadja az elsőt. Ha csak az egyik null, akkor visszaadja a nem null endpoint-ot. Különben kivételt dob “A cső nincs csatlakoztatva sehova” üzenettel.
* **+void RemoveEndPoint(Element element):** Kitörli a megadott element játékelemet az endpoints változóból.
* **+void FillWater(int amount):** Feltölti a waterLevel értékét amount-tal maximálisan a capacity értékéig, majd visszaadja, hogy mennyivel növekedett a waterLevel értéke.
* **+int AbsorbWater(int requested):** Lecsökkenti a waterLevel értékét requested-del maximálisan nulláig, majd visszaadja, hogy mennyivel csökkent a waterLevel értéke.
* **+void Step():** Csökkenti eggyel a sabotageable, sticky és slippery változók értékét maximum 0 értékig.

**8.1.8 Player**

#### · Felelősség

A játékosok tevékenységeit gyűjti össze általánosságban. Belőle származnak le a két csapat tagjai (Saboteur, Plumber). Külön azonosítóval rendelkezik minden játékos a megkülönböztetésük érdekében. Egy adott játékosnál egyszerre maximum 1 csővég lehet.

#### · Ősosztályok

-

#### · Interfészek

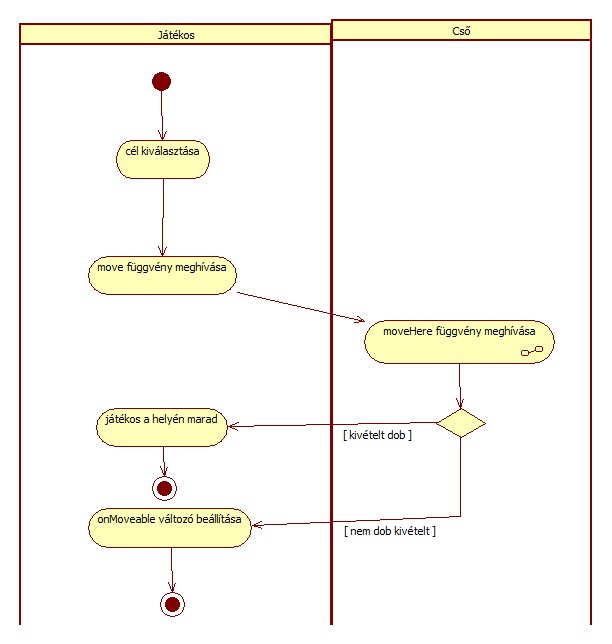
-

#### · Attribútumok

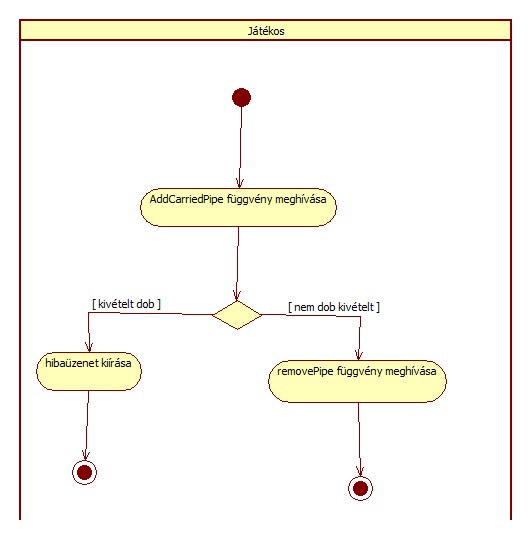
* **-int id:** A játékos egyedi azonosítója.
* **-bool canMove:** Tud-e mozogni a játékos.
* **-Moveable onMoveable:** Melyik mozogható helyen van éppen a játékos.
* **-List<Pipe> carriedPipe:** Egy maximum két elemű lista, amiben el van tárolva 1-szer vagy 2-szer ugyanaz a cső attól függően, hogy egy vagy mindkét vége fel van véve a játékoshoz.

#### · Metódusok

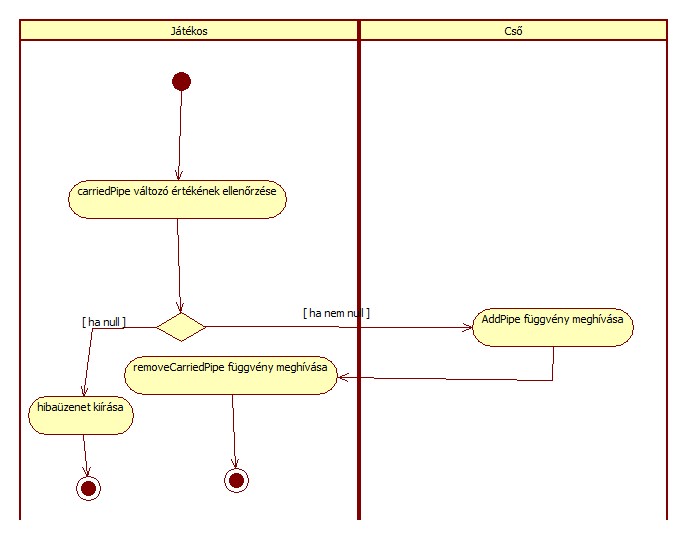
* **+void Move(Moveable destination):** A megadott destination paraméterre meghívja a MoveHere-t és az onMoveable-t beállítja a visszakapott értékre. Ezt egy try-catch blokkba kell tenni, mivel ha nem lehet a destination paraméterre lépni, kivételt fog dobni a program.



* **+void ChangePumpOutput(Pump pump, Pipe pipe):** Meghívja a SetOutput fgv.t a paraméterben megadott pumpára pipe paraméterrel.
* **+void ChangePumpInput(Pump pump, Pipe pipe):** Meghívja a SetInput fgv.t a paraméterben megadott pumpára pipe paraméterrel.
* **+void DetachPipe(Pipe pipe, Element element):** Meghívja az AddCarriedPipe-ot a pipe paraméterrel. Ezután meghívja a RemovePipe-ot a paraméterben megadott elementre pipe paraméterrel. Ezt egy try-catch blokkban kell megvalósítani, mivel az AddCarriedPipe kivételt dobhat, ha már van felvéve másik cső a játékosnál.



* **+void AttachPipe(Pipe pipe, Element element):** Ha a carriedPipe null, akkor Exception-t dob “Nincs felvéve cső” üzenettel. Különben meghívja a AddPipe-ot a paraméterben megadott Elementre. Aztán hívja magára a RemoveCarriedPipe függvényt pipe paraméterrel.



* **+void AddCarriedPipe(Pipe pipe):** Hogyha nincs még elem a carriePipe-ban hozzáadja, különben kivételt dob “Más cső van a játékosnál” üzenettel.
* **+void RemoveCarriedPipe(Pipe pipe):** Kitörli a pipe csövet a carriedPipe listából. Kivételt dob, ha a pipe cső nincs a listában “Nincs ilyen cső felvéve” üzenettel.
* **+Sabotage(Pipe pipe):** Ha a pipe sabotageable változója 0, vagyis ki lehet lyukasztani a csövet, meghívja a pipe Disable fgv-ét. Különben kivételt dob “Nem lehet még kilyukasztani a csövet” üzenettel.
* **+MakeSticky(Pipe pipe):** Meghívja a pipe slippery értéke 0, átállítja a pipe sticky értékét 5-re. Különben kivételt dob “Már ragadós volt a cső” üzenettel.

**8.1.9 Plumber**

#### · Felelősség

A csöveket tudja kilyukasztani. Csöveket/pumpákat tudja megjavítani. Meglévő játékelemeket tudnak a ciszternáknál készülő új csövekkel összekötni. Új pumpákat tudnak a játékban lehelyezni egy adott csőszakaszra, valamint ezek ki- és bemenetét állítani (egyszerre csak egy ki- és bemenete lehet). Új pumpákat a ciszternáktól tudnak venni, Egyszerre egy pumpa lehet egy szerelőnél. A játékelemeken tudnak közlekedni. A csöveken nem férnek el egymás, illetve a szabotőrök mellett csak a többi játékelemen.

#### · Ősosztályok

Player

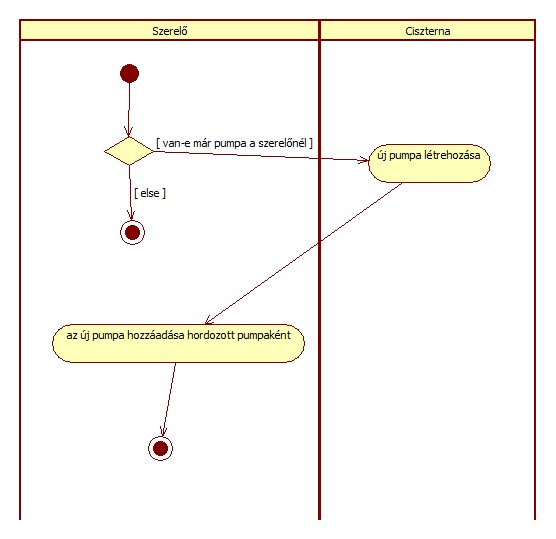
#### · Interfészek

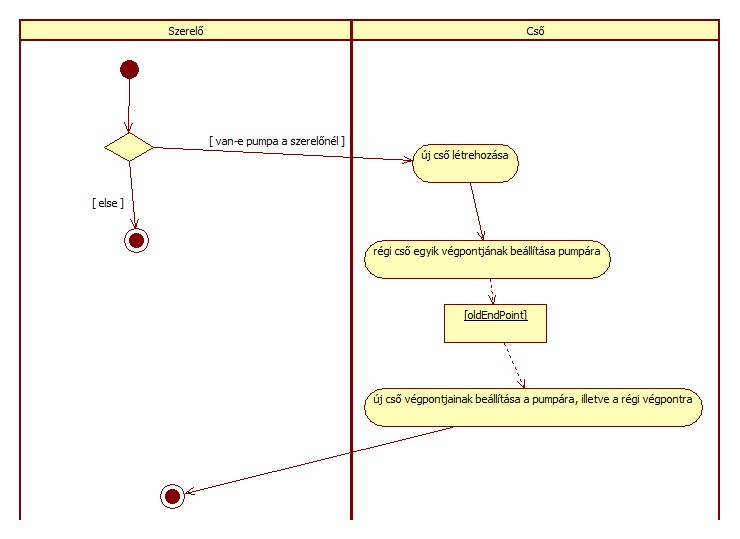
-

#### · Attribútumok

* **-bool hasPump:** Van-e az adott szerelőnél pumpa.
* **-Pump carriedPump:** A szerelőnél lévő pumpa, amit majd le tud tenni egy csőre.

#### · Metódusok

* **+void Fix(Fixable f):** Egy javítható játékelem megjavítása/befoltozása. Meghívja a fixable paraméter Fix függvényét.
* **+void GetPump(Cistern cistern):** Ha a carriedPump null, akkor meghívja a cistern GivePump függvényét, ami vissza ad egy új pumpát. Meghívja az AddCarriedPumpot az új pumpával paraméterként.   
  Ez hozzáadja a carriedPump változóba az új pumpát. Viszont ha carriedPump nem null volt, akkor kivételt dob “Van már felvett pumpa” üzenettel.
* **+void PlacePump(Pipe pipe):** Ha a carriedPump null, kivételt dob “Nincs letehető pumpa” üzenettel. Ha van akkor, létrehoz egy új csövet newPipe néven. Lekéri egy végpontját a pipe-nak a GetEndPoint függvénnyel és ezt elmenti oldEndpoint néven majd meghívja a pipe RemoveEndPoint-ját oldEndpoint paraméterrel. Utána hozzáadja a pipe-hoz a kitörölt végpont helyett a letett pumpát az AddEndPoint-tal carriedPump paraméterrel. Ezután beállítja a newPipe végpontjait. Meghívja a newPipera az AddEndPoint-ot oldEndpoint, majd carriedPump paraméterrel. Ezt try-catch blokkban kell megvalósítani, mivel kivételek érkezhetnek.



* **+void AddcarriedPump(Pump pump):** Hozzáadja a carriedPump változóba az új pumpát.
* **+void RemoveCarriedPump():** Átállítja null-ra a carriedPump változót.

**8.1.10 PointerCounter**

#### · Felelősség

Egy interface. A játék pontszerzési metódusait valósítja meg.

#### · Ősosztályok

-

#### · Interfészek

-

#### · Attribútumok

-

#### · Metódusok

* **+IncreasePoints(int amount):** Hozzáadja a paraméterben megadott értéket a megfelelő csapatnak.

**8.1.11 Pump**

#### · Felelősség

A csövekben lehet elágazásokat létrehozni vele. Sok cső kapcsolódhat hozzá, de egyszerre csak egy be- és kimenete lehet. Van tartálya, amiben a bemeneti csőből érkező vizet először eltárolja, mielőtt továbbadja a kimeneti cső felé. A ki- és bemenetet a játékosok tudják állítani. Bizonyos időközönként el tud romlani, ilyenkor nem ad ki a kimenetén vizet.

#### · Ősosztályok

Element

#### · Interfészek

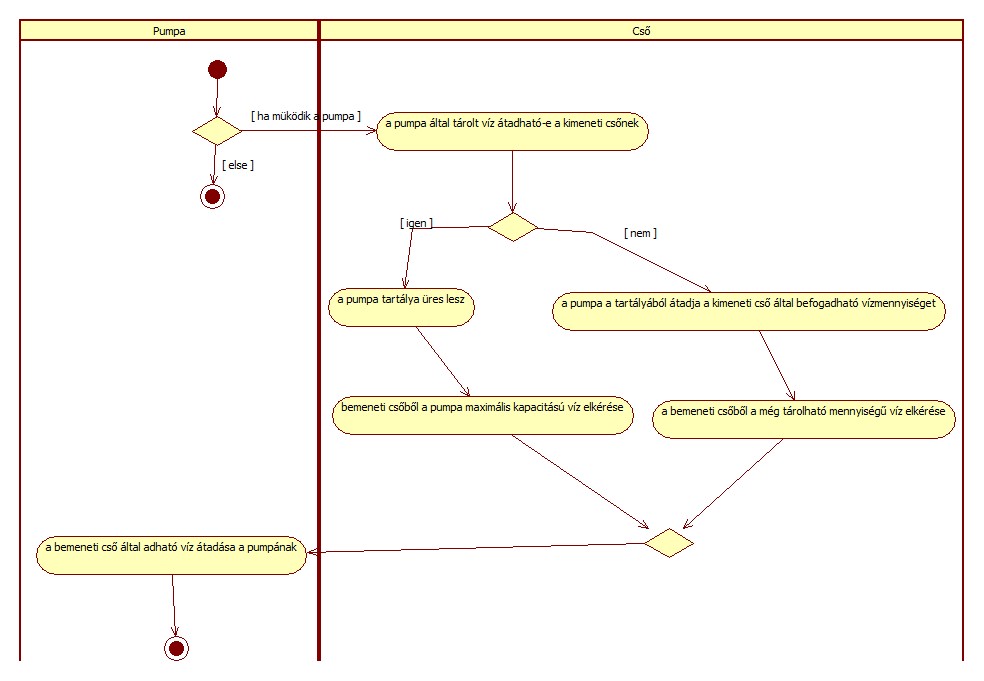
-

#### · Attribútumok

* **-Pipe output**: A cső amely felé továbbítja a vizet.
* **-Pipe input:** A cső amely felől kapja a vizet.
* **-bool working:** Megmondja, hogy tud-e vizet továbbítani a pumpa.
* **-int storedWater:** A tárolójában lévő víz mennyisége.
* **-int maxStorage:** A maximum vízmennyiség, amit a pumpa el tud tárolni.

#### · Metódusok

* **+void WaterFlow():** Ha a working változó igaz, akkor meghívja az output csőnek a FillWater függvényét storedWater paraméterrel. Majd visszakap a FillWater fgv.-től egy számot és ezt kivonja a storedWater változóból. (Amit visszakap érték, az azt jelenti, hogy mennyi vizet tudott elvinni a cső). Ezután a input csőnek meghívja az AbsorbWater függvényét maxStorage-storedWater értékkel (vagyis hogy mennyi vizet tud még eltárolni). Az AbsorbWater visszatérési értékével növeli a storedWater értékét. Ha working hamis, akkor nem pumpál vizet.



* **+void Fix():** Megjavítja a hibás pumpát. Átállítja a working változót igazzá.
* **+void Disable():** Elrontja a pumpát. Átállítja a working változót hamissá.
* **+void SetOutput(Pipe pipe):** Beállítja az output változót a paraméterben megadott csőre. Ha nincs csatlakoztatva a paraméterrel megadott cső a pumpára, akkor kivételt dob “Nincs a cső csatlakoztatva” üzenettel.
* **+void SetInput(Pipe pipe)**: Beállítja az input változót a paraméterben megadott csőre. Ha nincs csatlakoztatva a paraméterrel megadott cső a pumpára, akkor kivételt dob “Nincs a cső csatlakoztatva” üzenettel.
* **+void RandomDisable(bool random):** Hogyha a random paraméter hamis, átállítja a working változót hamisra. Különben véletlenszerűen átállítja a working változót hamisra vagy nem.

**8.1.12 PumpController**

#### · Felelősség

Interface.Véletlenszerűen elront egy véletlenszerű pumpát.

#### · Ősosztályok

-

#### · Interfészek

-

#### · Attribútumok

-

#### · Metódusok

* **+void RandomDisable(bool random):** Elrontja véletlenszerűen a pumpát. Ki lehet kapcsolni a random paraméterrel.

**8.1.13 Saboteur**

#### · Felelősség

A csöveket tudja kilyukasztani. A pumpákat át tudja állítani, hogy víz irányát megváltoztassa. A szabotőrök a játék elemeken tudnak mozogni. Viszont a csövön nem tudja megkerülni a másik játékost.

#### · Ősosztályok

Player

#### · Interfészek

-

#### · Attribútumok

-

#### · Metódusok

* **+void MakeSlippery(Pipe p):** Meghívja a pipe slippery értéke 0, átállítja a pipe slippery értékét 5-re. Különben kivételt dob “Már csúszós volt a cső” üzenettel.

**8.1.14 Source**

#### · Felelősség

A forrásból jön a végtelen mennyiségű víz, amiből a szerelők el tudják vezetni a vizet a ciszternákba. Ennek nincsen bemenetje, csak kimentje van. A forrásnak több kimenetje is lehet a játék folyamán.

#### · Ősosztályok

Element

#### · Interfészek

-

#### · Attribútumok

-

#### · Metódusok

* **+void WaterFlow():** Meghívja az összes pipes-ban lévő csőre a FillWater-t 10 értékkel paraméterként, mivel ez a kapacitása a csöveknek.

**8.1.15. Stepable**

#### · Felelősség

Interface. Az idő múlását reprezentálja.

#### · Ősosztályok

-

#### · Interfészek

-

#### · Attribútumok

-

#### · Metódusok

* **+void Step()**: Minden időpillanatban meghívódik.

**8.1.16. WaterFlow**

#### · Felelősség

Interface. A víz folyását valósítja meg .

#### · Ősosztályok

-

#### · Interfészek

-

#### · Attribútumok

-

#### · Metódusok

* **+void WaterFlow()**: Minden időpillanatban meghívódik.

## 8.2 A tesztek részletes tervei, leírásuk a teszt nyelvén

### 8.2.0 Pályagenerálás működése

* **Leírás**

Létrehozunk csöveket, aktív elemeket, szabotőröket és szerelőket.

**Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy az objektumok és a köztük lévő kapcsolatok helyesen létrejönnek-e. Ha nem működik, akkor a játék nem fog működni.

* **Bemenet**

CreatePump: KezdoPump1(null, null, 0, 0, true, null, 15)

CreatePump: KezdoPump2(null, null, 0, 0, true, null, 15)

CreatePipe:KezdoPipe(0, true, 15, false, 0, 0, 0, KezdoPump1, KezdoPump2)

CreateCistern:KezdoCistern(0, null, null, 15)

CreateSource: KezdoSource(null, 15)

CreatePlumber: KezdoPlumber(false, 1, false, null, KezdoPump1, null)

CreateSaboteur: KezdoSaboteur( 2, KezdoPipe, null)

* **Elvárt kimenet**

**CreatePump:**  KezdoPump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePump:**  KezdoPump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePipe:** KezdoPipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 15, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: KezdoPump1, KezdoPump2

**CreateCistern:** KezdoCistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreateSource:** KezdoSource forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePlumber:** KezdoPlumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: KezdoPump1, carriedPipes: null.

**CreateSaboteur:** KezdoSaboteur szabotőr létrejött  
Adatai: ID: 2 ,location: KezdoPump1, carriedPipes: null

### 8.2.1 Játékos mozgatása aktív elemről csőre, ha a cső nem foglalt

* **Leírás**

Egy játékos egy aktív elemen áll. Átmozgatjuk egy, a tartózkodási helyéhez csatlakozó csőre. Ez a cső nem foglalt, ekkor azt várjuk, hogy a játékosunk átkerül a csőre.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Megnézzük, hogy játékos át tud-e lépni egy üres csőre.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, false, 0, 0, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump1, null)

Move(Plumber, Pipe)

Info(Plumber)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Move:** Plumbersikeresen ment a Pipe elemre.

**Info:** Plumber  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pipe, carriedPipes: null.

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: true, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

### 8.2.2 Játékos mozgatása aktív elemről csőre, ha a cső foglalt.

* **Leírás**

Egy játékos egy aktív elemen áll. Átmozgatjuk egy, a tartózkodási helyéhez csatlakozó csőre. Ez a cső foglalt, ekkor azt várjuk, hogy a játékosunk az aktív elemen marad.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Leellenőrizzük, hogy egy csövön egyszerre csak játékos tud-e tartózkodni.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, true, 0, 0, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump, null)

Move(Plumber, Pipe)

Info(Plumber)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: - , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: true, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Move:** Plumbersikertelenül ment a Pipe elemre.

**Info:** Plumber  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Info:** Pipe  
Adatai: waterLevel: 0, working: false, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 5, endpoints: Pump1, Pump2

### 8.2.3 Játékos mozgatása aktív elemről csőre, ha a cső nem foglalt és csúszós

* **Leírás**

Egy játékos egy aktív elemen áll. Átmozgatjuk egy, a tartózkodási helyéhez csatlakozó csőre. Ez a cső nem foglalt, viszont csúszós. Ekkor azt várjuk, hogy a játékosunk vagy az eredeti helyén marad vagy a cső túlsó végére kerül.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy ha egy cső csúszós, akkor nem tudunk rálépni, helyette valamelyik végpontra kerülünk.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, true, 0, 0, 5, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump1, null)

Move(Plumber, Pipe)

Info(Plumber)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: - , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: true, sabotageable: 0, sticky: 0, slippery: 5, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Move:** Plumbersikertelenül ment a Pipe elemre.

**Info:** Plumber  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump2, carriedPipes: null.

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 5, endpoints: Pump1, Pump2

### 8.2.4 Játékos mozgatása csőről aktív elemre, úgy hogy az aktív elemen áll egy másik játékos

* **Leírás**

A játékosunk egy csövön áll, egy csőhöz csatlakozó aktív elemre próbál lépni, amelyen már áll egy másik játékos. Ekkor azt várjuk el, hogy a játékosunk az aktív elemre kerül.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy aktív elemen egyszerre több játékos is tartózkodhat.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, true, 0, 0, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump1, null)

CreateSaboteur: Saboteur( 2, Pipe, null)

Move(Saboteur, Pump1)

Info(Plumber)

Info(Saboteur)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: - , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: true, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai: ID: 2, location: Pipe, carriedPipes: null

**Move:** Saboteursikeresen ment a Pump1 elemre.

**Info:** Plumber  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Info:** Saboteur  
Adatai: ID: 2, location: Pump1, carriedPipes: null

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

### 8.2.5 Játékos mozgatása csőről aktív elemre, ha cső ragadós

* **Leírás**

Egy játékos egy csövön áll. Megpróbáljuk átmozgatni egy, a tartózkodási helyéhez csatlakozó aktív elemre. A cső ragadós. Ekkor azt várjuk, hogy a játékosunk nem tud átlépni az aktív elemre, a csövön marad.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy ha egy cső ragadós, akkor nem tudunk elmenni róla.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, true, 0, 0, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pipe, null)

Move(Plumber, Pump1)

Info(Plumber)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: - , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: true, sabotageable: 0, sticky: 5, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pipe, carriedPipes: null.

**Move:** Plumbersikertelenül ment a Pump1 elemre.

**Info:** Plumber  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pipe, carriedPipes: null.

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: true, sabotageable: 0, sticky: 5, slippery: 0, endpoints: Pump1, Pump2l

### 8.2.6 Játékos mozgatása aktív elemről olyan csőre, amelynek egyik vége le van csatolva.

* **Leírás**

Egy játékos egy aktív elemen áll. Átmozgatjuk egy, a tartózkodási helyéhez csatlakozó csőre. Ez a cső nem foglalt, viszont a másik vége le van csatlakoztatva. Ekkor azt várjuk, hogy a játékosunk az eredeti helyén marad.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy ha egy cső egyik vége le van csatolva, akkor nem tudunk rálépni.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, false, 0, 0, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump, null)

Move(Plumber, Pipe)

Info(Plumber)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, null

**AddPipe:** Pump aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump, carriedPipes: null.

**Move:** Plumbersikertelenül ment a Pipe elemre.

**Info:** Plumber  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump, carriedPipes: null.

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, null

### 8.2.7 Víz átjuttatása forrásból a hozzá csatlakozó nem lyukas, üres csőbe, amelynek másik vége is csatlakoztatva van

* **Leírás**

A forrás teletölti a hozzá csatlakozó csövet, amelynek vízszintje a maximumra emelkedik.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy mindkét végén csatlakoztatott, nem lyukas cső nem folyatja ki magából a vizet.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePump:Pump(null, null, 0, 10, true, null, 15)

CreatePipe:Pipe(0, true, 10, false, 0, 0, 0, Source, Pump)

AddPipe(Source, Pipe)

AddPipe(Pump, Pipe)

WaterFlow(Source)

Info(Pipe)

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**AddPipe:** Source aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**WaterFlow:** Input: null =(0)=>Aktív elem: Source =(10)=> Output: Pipe

**Info:** Pipe  
Adatai: waterLevel: 10, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

### 8.2.8 Víz átjuttatása forrásból a hozzá csatlakozó nem lyukas, üres csőbe, amelynek másik vége nincs csatlakoztatva

* **Leírás**

A forrás vizet tölt a hozzá csatlakozó csőbe, amelyből az összes víz a sivatagba folyik. Minden egység sivatagba ömlött víz után 1 pontot kapnak a szabotőrök.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy egyik végén nem csatlakoztatott, nem lyukas csőből a sivatagba folyik a víz.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePipe:Pipe(0, true, 10, false, 0, 0, 0, Source, null)

AddPipe(Source, Pipe)

WaterFlow(Source)

Info(Pipe)

Points()

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, null

**AddPipe:** Source aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**WaterFlow:** Input: null =(0)=>Aktív elem: Source =(10)=> Output: Pipe

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**Points:** Szabotőrök: 10, Szerelők: 0

### 8.2.9 Víz átjuttatása forrásból a hozzá csatlakozó lyukas, üres csőbe, amelynek másik vége is csatlakoztatva van

* **Leírás**

A forrás vizet tölt a hozzá csatlakozó csőbe, amelyből az összes víz a sivatagba folyik.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy mindkét végén csatlakoztatott, lyukas csőből a sivatagba folyik a víz. Minden egység sivatagba ömlött víz után 1 pontot kapnak a szabotőrök.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePump:Pump(null, null, 0, 10, true, null, 15)

CreatePipe:Pipe(0, true, 10, false, 0, 0, 0, Source, Pump)

AddPipe(Source, Pipe)

AddPipe(Pump, Pipe)

WaterFlow(Source)

Info(Pipe)

Points()

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: false, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**AddPipe:** Source aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**WaterFlow** Source aktív elemből 10 egység víz folyt át Pipe csőbe

**WaterFlow:** Input: null =(0)=>Aktív elem: Source =(10)=> Output: Pipe

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**Points:** Szabotőrök: 10, Szerelők: 0

### 8.2.10 Víz átjuttatása forrásból a hozzá csatlakozó nem lyukas, teli csőbe, amelynek másik vége is csatlakoztatva van

* **Leírás**

A forrás nem tölt több vizet a hozzá csatlakozó csőbe.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy mindkét végén csatlakoztatott, nem lyukas teli csőbe nem tudunk több vizet tölteni.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePump:Pump(null, null, 0, 10, true, null, 15)

CreatePipe:Pipe(10, true, 10, true, 0, 0, 0, Source, Pump)

AddPipe(Source, Pipe)

AddPipe(Pump, Pipe)

WaterFlow(Source)

Info(Pipe)

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 10, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**AddPipe:** Source aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**WaterFlow:** Input: null =(0)=>Aktív elem: Source =(0)=> Output: Pipe

**Info:** Pipe  
Adatai: waterLevel: 10, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

### 8.2.11 Víz átjuttatása ciszternába egy hozzá csatlakozó csőből.

* **Leírás**

A ciszterna az összes vizet kipumpálja a csőből.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy ciszterna ki tudja pumpálni a vizet egy csőből. Minden egység ciszternába juttatott víz után 1 pontot kapnak a szabotőrök.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 15)

CreatePump:Pump(null, null, 0, 10, true, null, 15)

CreatePipe:Pipe(10, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Pump, Pipe)

AddPipe(Cistern, Pipe)

WaterFlow(Cistern)

Info(Pipe)

Points()

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 10, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Cistern aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**WaterFlow:** Input: Pipe =(10)=>Aktív elem: Cistern =(0)=> Output: null

**Points:** Szabotőrök: 0, Szerelők: 10

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

### 8.2.12 Nem csúszós, nem ragadós cső lyukasztása

* **Leírás**

A cső nem csúszós és nem is ragadós, úgymond megfelelő állapotban van. Egy játékos rálép erre a csőre, majd kilyukasztja.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy megfelelő állapotú csövet ki tud lyukasztani egy szabotőr és egy szerelő is.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, false, 0, 0, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump1, null)

Move(Plumber, Pipe)

Sabotage(Plumber, Pipe)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Move:** Plumbersikeresen ment a Pipe elemre.

**Sabotage:** Plumbersikeresen kilyukasztotta Pipe-t

**Info:** Pipe  
Adatai: waterLevel: 0, working: false, capacity: 10, occupied: false, sabotageable: 5, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

### 8.2.13 Ragadós cső lyukasztása

* **Leírás**

A csövet beállítottuk ragadós állapotúra. Egy játékos rálép erre a csőre, majd kilyukasztja.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy szabotőr és egy szerelő akkor is ki tudja lyukasztani a csövet, ha az ragadós állapotú.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, false, 0, 1, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump1, null)

Move(Plumber, Pipe)

Sabotage(Plumber, Pipe)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 1, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Move:** Plumbersikeresen ment a Pipe elemre.

**Sabotage:** Plumbersikeresen kilyukasztotta Pipe-t

**Info:** Pipe  
Adatai: waterLevel: 0, working: false, capacity: 10, occupied: false, sabotageable: 5, sticky: 1, slippery: 0, endpoints: Pump1, Pump2

### 8.2.14 Lyukas cső lyukasztása

* **Leírás**

A csövet már korábban kilyukasztottuk. Egy játékos rálép erre a csőre, majd meg próbálja kilyukasztja.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy szabotőr és egy szerelő ha megpróbál kilyukasztani egy csövet, akkor nem történik semmilyen változás.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, false, 10, false, 0, 0, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump1, null)

Move(Plumber, Pipe)

Sabotage(Plumber, Pipe)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: false, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Move:** Plumbersikeresen ment a Pipe elemre.

**Sabotage:** Plumbersikeresen kilyukasztotta Pipe-t

**Info:** Pipe  
Adatai: waterLevel: 0, working: false, capacity: 10, occupied: false, sabotageable: 5, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

### 8.2.15 Foltozott cső lyukasztása a megadott időn belül

* **Leírás**

A csövet már korábban kilyukasztottuk és azóta meg is javítottuk. Ámbár az az idő még nem járt le, ami után újra lyukaszthatunk. Egy játékos rálép erre a csőre, majd megpróbálja kilyukasztani.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy ha egy szabotőr vagy egy szerelő ha megpróbál kilyukasztani egy csövet, azon az időn belül, amikor még nem lehetséges, akkor az nem sikerül neki.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, false, 10, 0, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump1, null)

Move(Plumber, Pipe)

Sabotage(Plumber, Pipe)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 10, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Move:** Plumbersikeresen ment a Pipe elemre.

**Sabotage:** Plumbersikertelenül kilyukasztotta Pipe-t

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 10, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

### 8.2.16 Megfelelő állapotban lévő pumpa elrontása

* **Leírás**

Egy megfelelően működő pumpát elrontunk, szabotálunk.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy az elromlott pumpa nem működik tovább.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump(null, null, 0, 0, true, null, 15)

CreateSaboteur:Saboteur(1, Pump, null)

Sabotage(Saboteur, Pump)

Info(Pump)

* **Elvárt kimenet**

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai:,ID: 1, location: Pump, carriedPipes: null

**Sabotage:** Saboteursikeresen kilyukasztotta Pump-t

**Info:** Pump  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: false, pipes: null, maxPipes: 15

### 8.2.17 Hibás pumpa elrontása

* **Leírás**

Egy működő pumpa már el van romolva és újra megpróbáljuk elrontani.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

0Ellenőrizzük, hogy ha egy pumpa már el van romolva, akkor ha még egyszer el akarjuk rontani, akkor ne történjen változás.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump(null, null, 0, 0, false, null, 15)

CreateSaboteur:Saboteur(1, Pump, null)

Sabotage(Saboteur, Pump)

Info(Pump)

* **Elvárt kimenet**

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: false, pipes: null, maxPipes: 15

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai:,ID: 1, location: Pump, carriedPipes: null

**Sabotage:** Saboteursikeresen kilyukasztotta Pump-t

**Info:** Pump  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: false, pipes: null, maxPipes: 15

### 8.2.18 Lyukas cső megjavítása

* **Leírás**

Egy csövet kiválasztunk, amit kilyukasztunk. Majd ezt a csövet megjavítjuk.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy megjavított cső helyesen működik.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, false, 10, false, 10, 0, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump1, null)

Move(Plumber, Pipe)

FixGameElement(Plumber,Pipe)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: false, capacity: 10, occupied: false, sabotageable: 5, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Move:** Plumbersikeresen ment a Pipe elemre.

**FixGameElement :**Plumbersikeresen javította Pipet-t

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 5, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

### 8.2.19 Jó állapotban lévő cső megjavítása

* **Leírás**

Egy csövet kiválasztunk, ami tökéletesen működik. Majd ezt a csövet megpróbáljuk megjavítani.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy ha egy jó állapotban lévő csövet akarunk megjavítani, akkor az továbbra is jól működik.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, false, 0, 0, 0, Pump1, Pump2)

AddPipe(Pump1, Pipe)

AddPipe(Pump2, Pipe)

CreatePlumber:Plumber(false, 1, false, null, Pump1, null)

Move(Plumber, Pipe)

FixGameElement(Plumber,Pipe)

Info(Pipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**Move:** Plumbersikeresen ment a Pipe elemre.

**FixGameElement :**Plumbersikeresen javította Pipet-t

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

### 8.2.20 Hibás pumpa megjavítása

* **Leírás**

Egy pumpát kiválasztunk, amit elrontunk. Majd ezt a pumpát megjavítjuk.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy megjavított pumpa megfelelően működjön.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump(null, null, 0, 0, false, null, 15)

CreatePlumber:Plumber(false, 1, false, null, Pump, null)

FixGameElement(Plumber,Pump)

Info(Pump)

* **Elvárt kimenet**

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: false, pipes: null, maxPipes: 15

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**FixGameElement :**Plumbersikeresen javította Pump-t

**Info:** Pump  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

### 8.2.21 Jó állapotban lévő pumpa megjavítása

* **Leírás**

Egy pumpát kiválasztunk, ami tökéletesen működik. Majd ezt a pumpát megpróbáljuk megjavítani.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy ha egy jó állapotban lévő pumpát akarunk megjavítani, akkor az továbbra is jól működik.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump(null, null, 0, 0, true, null, 15)

CreatePlumber:Plumber(false, 1, false, null, Pump, null)

FixGameElement(Plumber,Pump)

Info(Pump)

* **Elvárt kimenet**

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump1, carriedPipes: null.

**FixGameElement :**Plumbersikeresen javította Pump-t

**Info:** Pump  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

### 8.2.22 Megadott játék betöltése

* **Leírás**

Fájlokban elmente tárolva vannak a játéknak különböző állapotai. Ezeket töltjük be.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a megadott fájlt sikeresen be tudjuk tölteni.

* **Bemenet**

Load(“Proba.txt”)

* **Elvárt kimenet**

**Load:** A pályát sikeresen betöltöttük.

### 8.2.23 Jelenlegi pálya elmentése

* **Leírás**

Elmenti a pálya jelenlegi állását a megadott fájlba.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a pálya jelenlegi állapota elmenthető-e.

* **Bemenet**

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:Pipe(0, true, 10, false, 10, 0, 0, Pump1, Pump2)

Save(“Proba.txt”)

* **Elvárt kimenet**

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 10, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**Save:** A pályát sikeresen elmentettük.

### 8.2.24 Működő pumpa vizet pumpál, a bemeneti cső nem üres. A pumpa több vizet tartalmaz, mint a kimeneti cső kapacitása

* **Leírás**

A pumpa a bementi csőből a tartályába pumpálja a vizet, onnan pedig a kimeneti csőbe. A kimeneti csövet a maximum szintig tölti.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, pumpa csak annyi vizet pumpál-e a kimeneti csőbe, mint amennyi belefér.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePump:Pump(null, null, 11, 15, true, null, 15)

CreateCistern:Cistern(0, null, null, 15)

CreatePipe:InPipe(3, true, 10, false, 0, 0, 0, Source, Pump)

CreatePipe:OutPipe(0, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Source, InPipe)

AddPipe(Pump, InPipe)

AddPipe(Pump, OutPipe)

AddPipe(Cistern, OutPipe)

ChangePumpInput(Pump, InPipe)

ChangePumpOutput(Pump, OutPipe)

WaterFlow(Pump)

Info(InPipe)

Info(Pump)

Info(OutPipe)

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 11, maxStorage: 15, working: true, pipes: null, maxPipes: 15

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePipe:** InPipe cső létrejött  
Adatai: waterLevel: 3, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**CreatePipe:** OutPipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Source aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : InPipe, OutPipe

**AddPipe:** Cistern aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : OutPipe

**ChangePumpInput:** Pump bemeneti csöve InPipe csőre változott.

**ChangePumpOutput:** Pump kimeneti csöve OutPipe csőre változott.

**WaterFlow** Input: InPipe =(3)=>Aktív elem: Pump =(10)=> Output: OutPipe

**Info:** InPipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**Info:**  Pump  
Adatai: output: OutPipe, input: InPipr, storedWater: 4, maxStorage: 15, working: true, pipes: InPipe, OutPipe, maxPipes: 15

**Info:** OutPipe  
Adatai: waterLevel: 10, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

### 8.2.25 c

* **Leírás**

A pumpa a saját tartályából pumpál vizet a kimenetbe, amíg ki nem ürül.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a pumpa akkor is képes vizet továbbítani a kimeneti csőbe, ha a bemeneti cső üres.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePump:Pump(null, null, 10, 15, true, null, 15)

CreateCistern:Cistern(0, null, null, 15)

CreatePipe:InPipe(0, true, 10, false, 0, 0, 0, Source, Pump)

CreatePipe:OutPipe(0, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Source, InPipe)

AddPipe(Pump, InPipe)

AddPipe(Pump, OutPipe)

AddPipe(Cistern, OutPipe)

ChangePumpInput(Pump, InPipe)

ChangePumpOutput(Pump, OutPipe)

WaterFlow(Pump)

Info(InPipe)

Info(Pump)

Info(OutPipe)

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 10, maxStorage: 15, working: true, pipes: null, maxPipes: 15

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePipe:** InPipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**CreatePipe:** OutPipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Source aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : InPipe, OutPipe

**AddPipe:** Cistern aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : OutPipe

**ChangePumpInput:** Pump bemeneti csöve InPipe csőre változott.

**ChangePumpOutput:** Pump kimeneti csöve OutPipe csőre változott.

**WaterFlow** Input: InPipe =(0)=>Aktív elem: Pump =(10)=> Output: OutPipe

**Info:**  Pump  
Adatai: output: OutPipe, input: InPipe, storedWater: 0, maxStorage: 15, working: true, pipes: InPipe, OutPipe, maxPipes: 15

**Info:** OutPipe  
Adatai: waterLevel: 10, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

### 8.2.26 Működő pumpa vizet pumpál, a bemeneti cső nem üres. A pumpa kevesebb vizet tartalmaz, mint a kimeneti cső kapacitása.

* **Leírás**

A pumpa a bemeneti csőből a tartályába pumpálja a vizet, a tartályban még így is kevesebb víz van, mint amit a kimenet fogadni tudna. Ekkor a pumpa a tartályából az összes vizet a kimenetbe juttatja.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a pumpa mindig a lehető legtöbb vizet juttatja a kimeneti csőbe.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePump:Pump(null, null, 5, 15, true, null, 15)

CreateCistern:Cistern(0, null, null, 15)

CreatePipe:InPipe(3, true, 10, false, 0, 0, 0, Source, Pump)

CreatePipe:OutPipe(0, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Source, InPipe)

AddPipe(Pump, InPipe)

AddPipe(Pump, OutPipe)

AddPipe(Cistern, OutPipe)

ChangePumpInput(Pump, InPipe)

ChangePumpOutput(Pump, OutPipe)

WaterFlow(Pump)

Info(InPipe)

Info(Pump)

Info(OutPipe)

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 5, maxStorage: 15, working: true, pipes: null, maxPipes: 15

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePipe:** InPipe cső létrejött  
Adatai: waterLevel: 3, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**CreatePipe:** OutPipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Source aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : InPipe, OutPipe

**AddPipe:** Cistern aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : OutPipe

**ChangePumpInput:** Pump bemeneti csöve InPipe csőre változott.

**ChangePumpOutput:** Pump kimeneti csöve OutPipe csőre változott.

**WaterFlow** Input: InPipe =(3)=>Aktív elem: Pump =(8)=> Output: OutPipe

**Info:** InPipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**Info:**  Pump  
Adatai: output: OutPipe, input: InPipe, storedWater: 0, maxStorage: 15, working: true, pipes: InPipe, OutPipe, maxPipes: 15

**Info:** OutPipe  
Adatai: waterLevel: 8, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

### 8.2.27 Működő pumpa vizet pumpál, a bemeneti cső és a pumpa tartálya is üres.

* **Leírás**

A pumpa a saját tartályából próbál vizet pumpálni a kimenetbe, azonban ez üres. A bemeneti csőből sem tudja feltölteni a tartályát ugyanis ez is üres, ekkor nem kerül víz a kimeneti csőbe.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy ha pumpa tartálya és a bemeneti cső is üres, akkor nem kerül víz a kimenetbe.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePump:Pump(null, null, 0, 15, true, null, 15)

CreateCistern:Cistern(0, null, null, 15)

CreatePipe:InPipe(0, true, 10, false, 0, 0, 0, Source, Pump)

CreatePipe:OutPipe(0, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Source, InPipe)

AddPipe(Pump, InPipe)

AddPipe(Pump, OutPipe)

AddPipe(Cistern, OutPipe)

ChangePumpInput(Pump, InPipe)

ChangePumpOutput(Pump, OutPipe)

WaterFlow(Pump)

Info(InPipe)

Info(Pump)

Info(OutPipe)

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 15, working: true, pipes: null, maxPipes: 15

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePipe:** InPipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**CreatePipe:** OutPipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Source aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : InPipe, OutPipe

**AddPipe:** Cistern aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : OutPipe

**ChangePumpInput:** Pump bemeneti csöve InPipe csőre változott.

**ChangePumpOutput:** Pump kimeneti csöve OutPipe csőre változott.

**WaterFlow** Input: InPipe =(0)=>Aktív elem: Pump =(0)=> Output: OutPipe

**Info:** InPipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**Info:**  Pump  
Adatai: output: OutPipe, input: InPipe, storedWater: 0, maxStorage: 15, working: true, pipes: InPipe, OutPipe, maxPipes: 15

**Info:** OutPipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

### 8.2.28 Pumpa vizet pumpál, a bemeneti csőben több víz van, mint ami a pumpa tartályába fér.

* **Leírás**

A pumpa csak annyi vizet vesz fel a bemeneti csőből, amennyi a tartályába fér.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a pumpa csak annyi vizet vesz fel a bemeneti csőből, mint amennyit képes eltárolni.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePump:Pump(null, null, 8, 15, true, null, 15)

CreateCistern:Cistern(0, null, null, 15)

CreatePipe:InPipe(10, true, 10, false, 0, 0, 0, Source, Pump)

CreatePipe:OutPipe(0, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Source, InPipe)

AddPipe(Pump, InPipe)

AddPipe(Pump, OutPipe)

AddPipe(Cistern, OutPipe)

ChangePumpInput(Pump, InPipe)

ChangePumpOutput(Pump, OutPipe)

WaterFlow(Pump)

Info(InPipe)

Info(Pump)

Info(OutPipe)

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 8, maxStorage: 15, working: true, pipes: null, maxPipes: 15

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePipe:** InPipe cső létrejött  
Adatai: waterLevel: 10, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**CreatePipe:** OutPipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Source aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : InPipe, OutPipe

**AddPipe:** Cistern aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : OutPipe

**ChangePumpInput:** Pump bemeneti csöve InPipe csőre változott.

**ChangePumpOutput:** Pump kimeneti csöve OutPipe csőre változott.

**WaterFlow** Input: InPipe =(7)=>Aktív elem: Pump =(10)=> Output: OutPipe

**Info:** InPipe  
Adatai: waterLevel: 3, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**Info:**  Pump  
Adatai: output: OutPipe, input: InPipe, storedWater: 5, maxStorage: 15, working: true, pipes: InPipe, OutPipe, maxPipes: 15

**Info:** OutPipe  
Adatai: waterLevel: 10, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

### 8.2.29 Pumpa vizet pumpál, a kimeneti cső tele van.

* **Leírás**

A pumpa nem továbbít vizet a kimeneti csőbe.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a pumpa nem továbbít vizet olyan csőbe, amelyik nem képes több vizet fogadni.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePump:Pump(null, null, 8, 15, true, null, 15)

CreateCistern:Cistern(0, null, null, 15)

CreatePipe:InPipe(5, true, 10, false, 0, 0, 0, Source, Pump)

CreatePipe:OutPipe(10, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Source, InPipe)

AddPipe(Pump, InPipe)

AddPipe(Pump, OutPipe)

AddPipe(Cistern, OutPipe)

ChangePumpInput(Pump, InPipe)

ChangePumpOutput(Pump, OutPipe)

WaterFlow(Pump)

Info(InPipe)

Info(Pump)

Info(OutPipe)

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 8, maxStorage: 15, working: true, pipes: null, maxPipes: 15

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePipe:** InPipe cső létrejött  
Adatai: waterLevel: 5, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**CreatePipe:** OutPipe cső létrejött  
Adatai: waterLevel: 10, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Source aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : InPipe, OutPipe

**AddPipe:** Cistern aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : OutPipe

**ChangePumpInput:** Pump bemeneti csöve InPipe csőre változott.

**ChangePumpOutput:** Pump kimeneti csöve OutPipe csőre változott.

**WaterFlow** Input: InPipe =(5)=>Aktív elem: Pump =(0)=> Output: OutPipe

**Info:** InPipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**Info:**  Pump  
Adatai: output: OutPipe, input: InPipe, storedWater: 13, maxStorage: 15, working: true, pipes: InPipe, OutPipe, maxPipes: 15

**Info:** OutPipe  
Adatai: waterLevel: 10, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

### 8.2.28 Pumpa nem működik.

* **Leírás**

A pumpa elromlott ekkor, nem tölt vizet a tartályába a bementi csőből és nem tölt a kimeneti csőbe se vizet.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy a nem működő pumpa nem továbbít vizet.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateSource:Source(null, 15)

CreatePump:Pump(null, null, 8, 15, false, null, 15)

CreateCistern:Cistern(0, null, null, 15)

CreatePipe:InPipe(5, true, 10, false, 0, 0, 0, Source, Pump)

CreatePipe:OutPipe(4, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Source, InPipe)

AddPipe(Pump, InPipe)

AddPipe(Pump, OutPipe)

AddPipe(Cistern, OutPipe)

ChangePumpInput(Pump, InPipe)

ChangePumpOutput(Pump, OutPipe)

WaterFlow(Pump)

Info(InPipe)

Info(Pump)

Info(OutPipe)

* **Elvárt kimenet**

**CreateSource:** Source forrás létrejött  
Adatai:pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 8, maxStorage: 15, working: false, pipes: null, maxPipes: 15

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePipe:** InPipe cső létrejött  
Adatai: waterLevel: 5, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**CreatePipe:** OutPipe cső létrejött  
Adatai: waterLevel: 4, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Source aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez InPipe cső sikeresen csatlakoztatva, pipes : InPipe

**AddPipe:** Pump aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : InPipe, OutPipe

**AddPipe:** Cistern aktív elemhez OutPipe cső sikeresen csatlakoztatva, pipes : OutPipe

**ChangePumpInput:** Pump bemeneti csöve InPipe csőre változott.

**ChangePumpOutput:** Pump kimeneti csöve OutPipe csőre változott.

**WaterFlow** Input: InPipe =(0)=>Aktív elem: Pump =(0)=> Output: OutPipe

**Info:** InPipe  
Adatai: waterLevel: 5, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Source, Pump

**Info:**  Pump  
Adatai: output: OutPipe, input: InPipe, storedWater: 8, maxStorage: 15, working: true, pipes: InPipe, OutPipe, maxPipes: 15

**Info:** OutPipe  
Adatai: waterLevel: 4, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

### 8.2.29 Cső lecsatlakoztatása, a cső nem be- vagy kimeneti cső

* **Leírás**

A játékos egy pumpán áll és a pumpához csatlakozó csövek közül lecsatol egyet. A lecsatolandó cső nem be- vagy kimenete a pumpának ekkor a játékoshoz kerül a cső vége és ezt magával tudja cipelni.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy le tudunk-e csatolni egy csövet.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 15)

CreatePump:Pump(null, null, 0, 10, true, null, 15)

CreatePipe:Pipe(10, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Pump, Pipe)

AddPipe(Cistern, Pipe)

CreateSaboteur:Saboteur(1, Pump, null)

PickUpPipeEnd(Saboteur, Pipe)

Info(Pump)

Info(Pipe)

Info(Saboteur)

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Cistern aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai:,ID: 1, location: Pump, carriedPipes: null

**PickUpPipeEnd:** Saboteursikeresen felvette Pipe cső végét Pump játékelemről.

**Info:**  Pump  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: null, Cistern

**Info:** Saboteur  
Adatai:,ID: 1, location: Pump, carriedPipes: Pipe

### 8.2.30 Cső lecsatlakoztatása, a cső be- vagy kimeneti cső

* **Leírás**

A játékos egy pumpán áll és a pumpához csatlakozó csövek közül lecsatol egyet. A lecsatolandó cső be- vagy kimenete a pumpának ekkor a játékos nem tudja lecsatolni a csövet.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy be- vagy kimeneti csövet nem tudunk lecsatolni.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 15)

CreatePump:Pump(null, null, 0, 10, true, null, 15)

CreatePipe:Pipe(10, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Pump, Pipe)

AddPipe(Cistern, Pipe)

CreateSaboteur:Saboteur(1, Pump, null)

ChangePumpInput(Pump, Pipe)

PickUpPipeEnd(Saboteur, Pipe)

Info(Pump)

Info(Pipe)

Info(Saboteur)

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Cistern aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai:,ID: 1, location: Pump, carriedPipes: null

**ChangePumpInput:** Pump bemeneti csöve Pipe csőre változott.

**PickUpPipeEnd:** Saboteur nem tudta felvenni Pipe cső végét Pump játékelemről.

**Info:**  Pump  
Adatai: output: null, input: Pipe, storedWater: 0, maxStorage: 10, working: true, pipes: Pipe, maxPipes: 15.

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**Info:** Saboteur  
Adatai:,ID: 1, location: Pump, carriedPipes: null

### 8.2.31 Pumpa lehelyezése csőre, a cső nem lyukas, a játékosnál van pumpa.

* **Leírás**

A játékos lehelyez egy pumpát arra a csőre, amin éppen áll. Ekkor keletkezik egy új cső amely a lehelyezett pumpa és a cső egyik végpontját köti össze, a régi cső pedig a lehelyezett pumpát és a másik végpontot köti össze.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy le tudunk-e helyezni egy pumpát a nem lyukas csőre és lehelyezés után a lehelyezett és a régi pumpák megfelelően vannak összekötve.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:oldPipe(0, true, 10, false, 0, 0, 0, Pump1, Pump2)

CreatePump:Pump3(null, null, 0, 0, true, null, 15)

AddPipe(Pump1, oldPipe)

AddPipe(Pump2, oldPipe)

CreatePlumber:Plumber(true, 1, Pump3, oldPipe, null)

PutDownPump(Plumber, Pipe)

Info(Plumber)

Info(Pump3)

Info(Pump1)

Info(Pump2)

Info(oldPipe)

Info(newPipe) //?? ez maradhat így?

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: - , maxPipes: 15

**CreatePipe:** oldPipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 5, slippery: 0, endpoints: Pump1, Pump2

**CreatePump:**  Pump3 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : oldPipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : oldPipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: true,ID: 1, carriedPump: Pump3, location: oldPipe, carriedPipes: null.

**PutDownPump:** Plumbersikeresen tette le Pump3-at oldPipa-ra.

**Info:** Plumber  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Pump3, carriedPipes: null.

**Info:**  Pump3  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: oldPipe, newPipe, maxPipes: 15

**Info:**  Pump1  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: oldPipe , maxPipes: 15

**Info:**  Pump2  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: newPipe , maxPipes: 15

**Info:** OldPipe   
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 5, slippery: 0, endpoints: Pump1, Pump3

**Info:** newPipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 5, slippery: 0, endpoints: Pump3, Pump2

**8.2.32**  **Pumpa lehelyezése csőre, a cső lyukas, a szerelőnél van pumpa.**

* **Leírás**

A szerelő megpróbál lehelyezni egy pumpát arra a csőre, amelyen éppen áll, azonban a cső lyukas. Ekkor a szerelő nem tudja lehelyezni a pumpát.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy lyukas csőre nem tudunk pumpát lehelyezni.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:oldPipe(0, false, 10, false, 0, 0, 0, Pump1, Pump2)

CreatePump:Pump3(null, null, 0, 0, true, null, 15)

AddPipe(Pump1, oldPipe)

AddPipe(Pump2, oldPipe)

CreatePlumber:Plumber(true, 1, Pump3, oldPipe, null)

PutDownPump(Plumber, Pipe)

Info(Plumber)

Info(Pump3)

Info(Pump1)

Info(Pump2)

Info(oldPipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: - , maxPipes: 15

**CreatePipe:** oldPipe cső létrejött  
Adatai: waterLevel: 0, working: false, capacity: 10, occupied: false, sabotageable: 0, sticky: 5, slippery: 0, endpoints: Pump1, Pump2

**CreatePump:**  Pump3 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : oldPipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : oldPipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: true,ID: 1, carriedPump: Pump3, location: oldPipe, carriedPipes: null.

**PutDownPump:** Plumbersikertelenül tette le Pump3-at oldPipa-ra.

**Info:** Plumber  
Adatai: hasPump: true,ID: 1, carriedPump: Pump3, location: oldPipe, carriedPipes: null.

**Info:**  Pump3  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**Info:**  Pump1  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: oldPipe , maxPipes: 15

**Info:**  Pump2  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: old , maxPipes: 15

**Info:** OldPipe   
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 5, slippery: 0, endpoints: Pump1, Pump2

### 8.2.33 Pumpa lehelyezése csőre, a cső nem lyukas, a szerelőnél nincs pumpa.

* **Leírás**

A szerelő megpróbál lehelyezni egy pumpát arra a csőre, amelyen éppen áll, azonban a nincs nála pumpa. Ekkor a szerelő nem tud pumpát lehelyezni.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy csak akkor tud egy szerelő pumpát lehelyezni, ha van nála pumpa.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 0, true, null, 15)

CreatePump:Pump2(null, null, 0, 0, true, null, 15)

CreatePipe:oldPipe(0, false, 10, false, 0, 0, 0, Pump1, Pump2)

AddPipe(Pump1, oldPipe)

AddPipe(Pump2, oldPipe)

CreatePlumber:Plumber(false, 1, null, oldPipe, null)

PutDownPump(Plumber, Pipe)

Info(Plumber)

Info(Pump1)

Info(Pump2)

Info(oldPipe)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null , maxPipes: 15

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: - , maxPipes: 15

**CreatePipe:** oldPipe cső létrejött  
Adatai: waterLevel: 0, working: false, capacity: 10, occupied: false, sabotageable: 0, sticky: 5, slippery: 0, endpoints: Pump1, Pump2

**AddPipe:** Pump1 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : oldPipe

**AddPipe:** Pump2 aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : oldPipe

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: oldPipe, carriedPipes: null.

**PutDownPump:** Plumbersikertelenül tette le carriedPump-at oldPipa-ra.

**Info:** Plumber  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: oldPipe, carriedPipes: null.

**Info:**  Pump1  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: oldPipe , maxPipes: 15

**Info:**  Pump2  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: old , maxPipes: 15

**Info:** OldPipe   
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 5, slippery: 0, endpoints: Pump1, Pump2

### 8.2.34 Pumpa felvétele ciszternánál, a szerelőnél nincsen pumpa

* **Leírás**

A szerelő a ciszternán áll és nincs nála pumpa. Megpróbál felvenni egy pumpát és sikeres lesz a művelet.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, játékos tud-e felvenni pumpát.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 15)

CreatePlumber:Plumber(false, 1, null, Cistern, null)

PickUpPump(Plumber, Cistern)

Info(Plumber)

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 0, working: true, pipes: null, maxPipes: 15

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: true,ID: 1, carriedPump: Pump, location: Cistern, carriedPipes: null

**PickUpPump:** Plumbersikertelenül vette fel newPump-t Cistern-ból

**Info:** Plumber  
Adatai: hasPump: true,ID: 1, carriedPump: Pump, location: Cistern, carriedPipes: null

### 8.2.35 Pumpa felvétele ciszternánál, a szerelőnél már van egy pumpa

* **Leírás**

A szerelő a ciszternán áll és van nála pumpa. Megpróbál felvenni egy pumpát és sikertelen lesz a művelet.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük,szerelő egyszerre csak egy pumpát tud magánál tartani.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 15)

CreatePump:Pump(null, null, 0, 0, true, null, 15)

CreatePlumber:Plumber(true, 1, Pump, Cistern, null)

PickUpPump(Plumber, Cistern)

Info(Plumber)

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePlumber:** Plumber szerelő létrejött  
Adatai: hasPump: false,ID: 1, carriedPump: null, location: Cistern, carriedPipes: null

**PickUpPump:** Plumbersikeresen vette fel newPump-t Cistern-ból

**Info:** Plumber  
Adatai: hasPump: true,ID: 1, carriedPump: newPump, location: Cistern, carriedPipes: null

### 8.2.36 Szabad végű cső létrehozása ciszternánál.

* **Leírás**

A ciszternán keletkezik egy új cső, amelynek egyik vége a ciszternához csatlakozik, a másik vége szabad. A szabad véget a játékosok fel tudják venni ha a ciszternára lépnek.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy létrejön-e a cső a ciszternán és az egyik vége szabad-e, továbbá a másik tényleg a ciszternához csatlakozik-e.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 15)

AddFreePipe:(newPipe, Cistern)

Info(Cistern)

Info(newPipe)

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**AddFreePipe:** Cistern ciszternához newPipe cső sikeresen hozzáadva

**Info:** Cistern  
Adatai: numberOfFreePipes: 1, createdPipes: newPipe, pipes: null, maxPipes: 15

**Info:** newPipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Cistern, null

### 8.2.37 Csővég felvétele ciszternánál, a játékosnál nincsen csővég

* **Leírás**

A játékos a ciszternán áll és nincs nála csővég. Megpróbál felvenni egy pumpát és sikeres lesz a művelet.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, játékos tud-e felvenni felvenni szabad csővéget a ciszternáról.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 15)

AddFreePipe:(freePipe, Cistern)

Info(Cistern)

CreateSaboteur:Saboteur(1, Cistern, null)

PickUpFreePipe(Saboteur, Cistern)

Info(Cistern)

Info(Saboteur)

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**AddFreePipe:** Cistern ciszternához freePipe cső sikeresen hozzáadva

**Info:** Cistern  
Adatai: numberOfFreePipes: 1, createdPipes: freePipe, pipes: null, maxPipes: 15

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai: ID: 1 ,location: Cistern, carriedPipes: null

**PickUpFreePipe:** Cistern ciszternáról freePipe cső sikeresen felvéve Saboteur által

**Info:** Cistern  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**Info:** Saboteur  
Adatai: ID: 1 ,location: Cistern, carriedPipes: freePipe

### 8.2.38 Szabad csővég felvétele ciszternánál, a játékosnál már van egy csővég

* **Leírás**

A játékos a ciszternán áll és van nála csővég. Megpróbál felvenni egy szabad csővéget és sikertelen lesz a művelet.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, játékos egyszerre csak egy csővéget tud magánál tartani.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 15)

AddFreePipe:(freePipe, Cistern)

Info(Cistern)

CreatePipe:Pipe(0, true, 10, false, 0, 0, 0, Cistern, null)

CreateSaboteur:Saboteur(1, Cistern, Pipe)

PickUpFreePipe(Saboteur, Cistern)

Info(Cistern)

Info(Saboteur)

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**AddFreePipe:** Cistern ciszternához freePipe1 cső sikeresen hozzáadva

CreatePipe()

**Info:** Cistern  
Adatai: numberOfFreePipes: 1, createdPipes: freePipe, pipes: null, maxPipes: 15

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai: ID: 1 ,location: Cistern, carriedPipes: Pipe

**PickUpFreePipe:** Cistern ciszternáról freePipe cső sikertelenül felvéve Saboteur által

**Info:** Cistern  
Adatai: numberOfFreePipes: 1, createdPipes: freePipel, pipes: null, maxPipes: 15

**Info:** Saboteur  
Adatai: ID: 1 ,location: Cistern, carriedPipes: Pipe

### 8.2.39 Pumpa bemenetének állítása

* **Leírás**

A játékos egy pumpán áll. A csatlakozó csövek közül egyet beállít bemenetként. Ezt követően a pumpa az új bemenetből szív ki vizet.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük,hogy át tudjuk-e állítani a pumpa bemenetét.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 10, true, null, 15)

CreatePump:Pump2(null, null, 0, 10, true, null, 15)

CreatePump:Pump3(null, null, 0, 10, true, null, 15)

CreatePipe:Pipe1(0, true, 10, false, 0, 0, 0, Pump1, Pump2)

CreatePipe:Pipe2(0, true, 10, false, 0, 0, 0, Pump1, Pump3)

AddPipe(Pump1, Pipe1)

AddPipe(Pump1, Pipe2)

AddPipe(Pump2, Pipe1)

AddPipe(Pump3, Pipe2)

ChangePumpInput(Pump1, Pipe1)

CreateSaboteur:Saboteur(1, Pump1, null)

Info(Pump1)

ChangePumpInputByPlayer(Pump1, Pipe2, Saboteur)

Info(Pump1)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePump:**  Pump3 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePipe:** Pipe1 cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 0, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**CreatePipe:** Pipe2 cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 0, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump3

**AddPipe:** Pump1 aktív elemhez Pipe1 cső sikeresen csatlakoztatva, pipes : Pipe1

**AddPipe:** Pump1 aktív elemhez Pipe2 cső sikeresen csatlakoztatva, pipes : Pipe1, Pipe2l

**AddPipe:** Pump2 aktív elemhez Pipe1 cső sikeresen csatlakoztatva, pipes : Pipe1

**AddPipe:** Pump3 aktív elemhez Pipe2 cső sikeresen csatlakoztatva, pipes : Pipe2

**ChangePumpInput:** Pump1 bemeneti csöve Pipe1 csőre változott.

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai:ID: 1, location: Pump1, carriedPipes: null

**Info:**  Pump1  
Adatai: output: null, input: Pipe1, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**ChangePumpInputByPlayer:** Saboteursikeresen megváltoztatta Pump1 bemeneti csövét Pipe2 csőre.

**Info:**  Pump1  
Adatai: output: null, input: Pipe2, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

### 8.2.40 Pumpa kimenetének állítása

* **Leírás**

A játékos egy pumpán áll. A csatlakozó csövek közül egyet beállít kimenetként. Ezt követően a pumpa az új kimenetbe továbbítja a vizet.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük,hogy át tudjuk-e állítani a pumpa kimenetét.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreatePump:Pump1(null, null, 0, 10, true, null, 15)

CreatePump:Pump2(null, null, 0, 10, true, null, 15)

CreatePump:Pump3(null, null, 0, 10, true, null, 15)

CreatePipe:Pipe1(0, true, 10, false, 0, 0, 0, Pump1, Pump2)

CreatePipe:Pipe2(0, true, 10, false, 0, 0, 0, Pump1, Pump3)

AddPipe(Pump1, Pipe1)

AddPipe(Pump1, Pipe2)

AddPipe(Pump2, Pipe1)

AddPipe(Pump3, Pipe2)

ChangePumpOutput(Pump1, Pipe1)

CreateSaboteur:Saboteur(1, Pump1, null)

Info(Pump1)

ChangePumpOutputByPlayer(Pump1, Pipe2, Saboteur)

Info(Pump1)

* **Elvárt kimenet**

**CreatePump:**  Pump1 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePump:**  Pump2 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePump:**  Pump3 pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePipe:** Pipe1 cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 0, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump2

**CreatePipe:** Pipe2 cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 0, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump1, Pump3

**AddPipe:** Pump1 aktív elemhez Pipe1 cső sikeresen csatlakoztatva, pipes : Pipe1

**AddPipe:** Pump1 aktív elemhez Pipe2 cső sikeresen csatlakoztatva, pipes : Pipe1, Pipe2l

**AddPipe:** Pump2 aktív elemhez Pipe1 cső sikeresen csatlakoztatva, pipes : Pipe1

**AddPipe:** Pump3 aktív elemhez Pipe2 cső sikeresen csatlakoztatva, pipes : Pipe2

**ChangePumpOutput:** Pump1 kimeneti csöve Pipe1 csőre változott.

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai:ID: 1, location: Pump1, carriedPipes: null

**Info:**  Pump1  
Adatai: output: Pipe1, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**ChangePumpInputByPlayer:** Saboteursikeresen megváltoztatta Pump1 kimeneti csövét Pipe2 csőre.

**Info:**  Pump1  
Adatai: output: Pipe2, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

### 8.2.41 Cső mindkét végének lecsatolása

* **Leírás**

Egy cső mindkét végét lecsatoljuk.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy cső mindkét végét le tudjuk-e csatolni.l

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 15)

CreatePump:Pump(null, null, 0, 10, true, null, 15)

CreatePipe:Pipe(10, true, 10, false, 0, 0, 0, Pump, Cistern)

AddPipe(Pump, Pipe)

AddPipe(Cistern, Pipe)

CreateSaboteur:Saboteur1(1, Pump, null)

CreateSaboteur:Saboteur2(2, Cistern, null)

PickUpPipeEnd(Saboteur1, Pipe)

PickUpPipeEnd(Saboteur2, Cistern)

Info(Pump)

Info(Cistern)

Info(Pipe)

Info(Saboteur1)

Info(Saboteur2)

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

**AddPipe:** Cistern aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**AddPipe:** Pump aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreateSaboteur:** Saboteur1 szabotőr létrejött  
Adatai:,ID: 1, location: Pump, carriedPipes: null

**CreateSaboteur:** Saboteur2 szabotőr létrejött  
Adatai:,ID: 2, location: Cistern, carriedPipes: null

**PickUpPipeEnd:** Saboteur1sikeresen felvette Pipe cső végét Pump játékelemről.

**PickUpPipeEnd:** Saboteur2sikeresen felvette Pipe cső végét Cistern játékelemről.

**Info:**  Pump  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 15.

**Info:** Cisternt  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 15

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: null, null

**Info:** Saboteur1  
Adatai:,ID: 1, location: Pump, carriedPipes: Pipe

**Info:** Saboteur2  
Adatai:,ID: 2, location: Cistern, carriedPipes: Pipe

### 8.2.42 Csővég letétele

* **Leírás**

Egy aktív elemre áll egy játékos, akinél van egy csővég. A játékos hozzácsatolja a csövet az aktív elemhez.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy csővéget cipelő játékos csatlakoztatni tudja-e a csövet egy aktív elemhez.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 2)

CreatePump:Pump(null, null, 0, 10, true, null, 2)

CreatePipe:Pipe(0, true, 10, false, 0, 0, 0, Pump, null)

AddPipe(Pump, Pipe)

CreateSaboteur:Saboteur(1,Cistern, Pipe)

PutDownPipeEnd(Saboteur, Cistern)

Info(Cistern)

Info(Saboteur)

Info(Pipe)

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 2

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 2.

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, null

**AddPipe:** Pump aktív elemhez Pipe cső sikeresen csatlakoztatva, pipes : Pipe

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai:,ID: 1, location: Cistern, carriedPipes: Pipe

**PutDownPipeEnd:** Saboteursikeresen letette Pipe cső végét Cistern játékelemre

**Info:** Cistern  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: Pipe, maxPipes: 2

**Info:** Saboteur  
Adatai:,ID: 1, location: Cistern, carriedPipes: null

**Info:** Pipe  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, Cistern

### 8.2.43 Maximálisnál több cső csatlakoztatása

* **Leírás**

Egy aktív elemhez a maximális számú cső csatlakozik. megpróbálunk egy újabb csövet csatlakoztatni de a művelet sikertelen.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük, hogy egy aktív elemhez csak adott számú cső csatlakoztatható.

* **Előfeltétel:** pályagenerálás működik
* **Bemenet**

CreateCistern:Cistern(0, null, null, 1)

CreatePump:Pump(null, null, 0, 10, true, null, 2)

CreatePipe:Pipe1(0, true, 10, false, 0, 0, 0, Pump, null)

CreatePipe:Pipe2(0, true, 10, false, 0, 0, 0, Cistern, null)

AddPipe(Pump, Pipe1)

AddPipe(Cistern, Pipe2)

CreateSaboteur:Saboteur(1,Cistern, Pipe1)

PutDownPipeEnd(Saboteur, Cistern)

Info(Cistern)

Info(Saboteur)

Info(Pipe1)

* **Elvárt kimenet**

**CreateCistern:** Cistern ciszterna létrejött  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: null, maxPipes: 1

**CreatePump:**  Pump pumpa létrejött  
Adatai: output: null, input: null, storedWater: 0, maxStorage: 10, working: true, pipes: null, maxPipes: 2.

**CreatePipe:** Pipe cső létrejött  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, null

**AddPipe:** Pump aktív elemhez Pipe1 cső sikeresen csatlakoztatva, pipes : Pipe1

**AddPipe:** Cistern aktív elemhez Pipe2 cső sikeresen csatlakoztatva, pipes : Pipe2

**CreateSaboteur:** Saboteur szabotőr létrejött  
Adatai:,ID: 1, location: Cistern, carriedPipes: Pipe1

**PutDownPipeEnd:** Saboteursikertelenül tette le Pipe1 cső végét Cistern játékelemre

**Info:** Cistern  
Adatai: numberOfFreePipes: 0, createdPipes: null, pipes: Pipe2, maxPipes: 1

**Info:** Saboteur  
Adatai:,ID: 1, location: Cistern, carriedPipes: Pipe1

**Info:** Pipe1  
Adatai: waterLevel: 0, working: true, capacity: 10, occupied: false, sabotageable: 0, sticky: 0, slippery: 0, endpoints: Pump, null

## 8.3 A tesztelést támogató programok tervei

Egy teszt letutattásához alapvetően csak a standard bemenetre lesz szükség. A bemeneti nyelv által definiált parancsokat soronként kell bevinni, az adott tesztesetekben megadott formátumban / sorrendben. Fontos a betűről betűre egyezés az egyes sorokban. Az egyes tesztesetek végét EOF-el kell jelezni. Az egyes tesztesetekhez a pályagenerálás történhet fájlból is, ehhez a bemeneti nyelvben definiált módon kell felépíteni, ezek után pedig az adott további parancsokat a megszokott módon, soronként kell megadni, majd EOF-el zárni.

Az egyes tesztek futtatásához nincs szükség külön programra, a TestOracle (továbbiakban szimplán tesztorákulumként hivatkozunk rá) osztály fogja az egyes teszteseteket értékelni. Az elvárt és tényleges kimenetet a tesztorákulum fogja ellenőrizni. Minden tesztesetnél a parancsok végrehajtás után a összehasonlítja a modell által módosított objektumok tulajdonságait a tesztorákulum által egy listában tárolt objektumokkal, amelyeken az egyes parancsok hatása végig lesz követve. Ily módon nem kell minden tesztesethez egy külön “hard-coded” elvárt kimenetet készíteni. A teszteset végén a tesztorákulum tájékoztatja a felhasználót a teszt sikerességéről. Ha az hibás, akkor a hibás részeket objektumonként kilistázza.

## 8.4 Napló

| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| --- | --- | --- | --- |
| 2023.04.26. 18:00 | 4 óra | Sütheö  Gerencsér  Vörös  Chu  Bognár | Értekezlet.  Döntés: Sütheö elkészíti a javított cső állapotgráfot 04.27 10:00-ig. Bognár és Chu elkészíti a metódusok részletes tervét, azok leírását 04.28. 20:00-ig. Sütheö elkészíti az aktivitásdiagramoakt 04.28. 20:00-ig. Gerencsér, Vörös elkészíti a tesztek részletes tervét 05.01. 18:00-ig. Sütheö elkészíti a tesztelést támogató programokat tervét 05.02. 14:00-ig. |
| 2023.04.26. 23:00 | 1,5 óra | Sütheö | Tevékenység: Sütheö elkészíti a javított cső állapotgráfot. (8.0) |
| 2023.04.27. 12:00 | 3 óra | Bognár | Tevékenység: Bognár elkészíti 8 darab osztály részletes leírását (metódusok, attribútumok). (8.1.1-8.1.8) |
| 2023.04.27. 18:00 | 2 óra | Chu | Tevékenység: Chu elkészíti 5 osztály részletes leírást (metódusok, attribútumok). (8.1.9-8.1.13) |
| 2023.04.28. 11:00 | 1,5 óra | Chu | Tevékenység: Chu elkészíti a maradék 3 osztály részletes leírását (metódusok, attribútumok). (8.1.14-8.1.16) |
| 2023.04.28. 15:00 | 1 óra | Sütheö | Tevékenység: Sütheö elkészíti az aktivitásdiagramokat. (8.1.1, 8.1.2, 8.1.7, 8.1.8, 8.1.9, 8.1.11) |
| 2023.04.29. 16:00 | 3,5 óra | Gerencsér | Tevékenység: Gerencsér elkészíti 23 darab tesztesetet leírását (bemenet, elvárt kimenetek, előfeltételek) (8.2.1-8.2.23) |
| 2023.05.01. 14:00 | 3 óra | Vörös | Tevékenység: Vörös elkészíti 20 darab tesztesetet (bemenetek, kimenet, előfeltételek (8.2.24-8.2.43) |
| 2023.05.02. 11:00 | 0,5 óra | Sütheö | Tevékenység: Sütheö elkészíti a tesztelést támogató programok tervei részt. A futtatás menetét, a tesztorákulum müködését. (8.1.3) |

**10. Prototípus beadása**

* 1. ***Fordítási és futtatási útmutató***
     1. **Fájllista**

| **Fájl neve** | **Méret** | **Keletkezés ideje** | **Tartalom** |
| --- | --- | --- | --- |
| Cistern.java | 3772 bytes | 2023.04.15. 12:15 | A ciszterna működéséhez szükséges attribútumok, függvények megírása. |
| Controller.java | 18 379 bytes | 2023.04.15. 18:12 | Teszteléshez volt szükséges, most már kevésbé releváns. |
| Desert.java | 1191 bytes | 2023.04.15. 12:21 | Az elfolyt víz mennyiségét nézi és növeli a szabotőrök pontjait. |
| Pipe.java | 9188 bytes | 2023.04.15 12:24 | A csövek működéséhez szükséges attribútumok, függvények megírása. |
| Player.java | 4958 bytes | 2023.04.15 12:27 | A játékosok lehetséges lépéseinek, műveleteinek megírása. |
| TestController.java | 26 857 bytes | 2023.05.12 14:14 | Teszteléshez szükséges függvények. |
| Element.java | 2893 bytes | 2023.04.15 12:31 | Az elemek működéséhez, állapotához szükséges függvények megírása. |
| Fixable.java | 264 bytes | 2023.04.15 12:34 | Az elemek működéséhez, állapotához szükséges függvények deklarálása. |
| Moveable.java | 376 bytes | 2023.04.15 12:36 | A mozgáshoz szükséges függvények deklarálása. |
| Plumber.java | 2903 bytes | 2023.04.15 12:40 | A szerelők lehetséges lépéseinek, műveleteinek megírása. |
| Pump.java | 4429 bytes | 2023.04.15 12:43 | A csövek működéséhez szükséges attribútumok, függvények megírása. |
| Saboteur.java | 510 bytes | 2023.04.15. 12:41 | A szabotőrök lehetséges lépéseinek, műveleteinek megírása. |
| Source.java | 909 bytes | 2023.04.15 12:45 | A forrás működéséhez szükséges attribútumok, függvények megírása. |
| Functioncall.java | 2366 bytes | 2023.04.15. 18:17 | Teszteléshez kiírásához volt szükséges, most már kevésbé releváns. |
| Game.java | 1103 bytes | 2023.04.15 11:50 | Játék indítása, befejezése, játékosok csatlakozása. |
| PointCounter | 210 bytes | 2023.04.15 13:22 | Pontokat számoló függvények deklarálása. |
| PumpController.java | 149 bytes | 2023.04.15. 13:26 | Pumpa elrontása függvény deklarálása. |
| Stepable.java | 157 bytes | 2023.04.15. 12:50 | Lépés függvény deklarálása. |
| WaterFlow.java | 107 bytes | 2023.04.15. 12:54 | Vízfolyás függvény deklarálása. |
| NoFreePipeCisternException.java | 238 bytes | 2023.05.13. 00:00 | Kivételdobás hibás működés esetén. |
| NoPickedupPipeException.java | 219 bytes | 2023.05.13. 00:05 | Kivételdobás hibás működés esetén. |
| NoPumpPickedupException.java | 234 bytes | 2023.05.13. 00:08 | Kivételdobás hibás működés esetén. |
| PipeAlreadyPickedupException.java | 214 bytes | 2023.05.13. 00:12 | Kivételdobás hibás működés esetén. |
| PipeGetEndpointException.java | 242 bytes | 2023.05.13. 00:14 | Kivételdobás hibás működés esetén. |
| PipePlaceException.java | 230 bytes | 2023.05.13. 00:17 | Kivételdobás hibás működés esetén. |
| PipeRemoveEndpointException.java | 269 bytes | 2023.05.13. 00:20 | Kivételdobás hibás működés esetén. |
| PipeSabotageableException.java | 259 bytes | 2023.05.13. 00:22 | Kivételdobás hibás működés esetén. |
| PipeSetEndpointException.java | 255 bytes | 2023.05.13. 00:25 | Kivételdobás hibás működés esetén. |
| PipeSlipperyException.java | 216 bytes | 2023.05.13. 00:29 | Kivételdobás hibás működés esetén. |
| PipeStickyException.java | 211 bytes | 2023.05.13. 00:33 | Kivételdobás hibás működés esetén. |
| PumpAlreadyPickedupException.java | 249 bytes | 2023.05.13. 00:36 | Kivételdobás hibás működés esetén. |
| PumpInputException.java | 236 bytes | 2023.05.13. 00:40 | Kivételdobás hibás működés esetén. |
| PumpOutputException.java | 242 bytes | 2023.05.13. 00:43 | Kivételdobás hibás működés esetén. |
| teszt0\_in.txt -> teszt43\_in.txt | 19 197 bytes | 2023.05.14. 8:43 | Tesztesetek bemenetei a tesztelés futattásához. |
| teszt0\_out.txt -> teszt43\_out.txt | 15 494 bytes | 2023.05.14. 8:50 | Tesztesetek kimentei a tesztesetek futásának ellenőrzéséhez, ezt a tesztorákulum használja fel. |

* + 1. **Fordítás**

Először is csomagoljuk ki a zip fájlban található .java fájlokat és a tesztbemeneteket tartozó mappát egy **Test** nevű mappába az **Asztalra** (rakhatjuk bármilyen más mappába is, csak a **cd** paranccsal adjuk meg a mappa elérési útját). Ezután indítsunk egy Command Prompt -ot (cmd.exe) és írjuk be ezt a két sort:

*cd C:\Users\cloud\Desktop\Test*

*javac Cistern.java Controller.java Desert.java Element.java Fixable.java FunctionCall.java Game.java Moveable.java MoveException.java NoFreePipeCisternException.java NoPickedupPipeException.java NoPumpPickedupException.java Pipe.java PipeAlreadyPickedupException.java PipeGetEndpointException.java PipePiercedPlacePumpException.java PipePlaceException.java PipeRemoveEndpointException.java PipeSabotageableException.java PipeSetEndpointException.java PipeSlipperyException.java PipeStickyException.java Player.java Plumber.java PointCounter.java Pump.java PumpAlreadyPickedupException.java PumpController.java PumpInputException.java PumpOutputException.java Saboteur.java Source.java Stepable.java TestController.java WaterFlow.java*

* + 1. **Futtatás**

Ha nem zártuk be a Command Prompt -t (cmd.exe) akkor elég ezt az egy sort beírni:

*java TestController*

És a listából a megfelelő számot beírva lefut a megfelelő teszteset, ennek a függvényhívás sorát tudjuk végigkövetni.

Ha bezártuk a Command Prompt-ot (cmd.exe) akkor indítsunk egy újat és írjuk be a következő parancsot két parancsot:

*cd C:\Users\cloud\Desktop\Test*

*java TestController*

Ezek után úgy mint az előző esetben, meg tudjuk hívni a megfelelő teszteseteket a sorszámok beírásával.

* 1. ***Tesztek jegyzőkönyvei***
     1. **Pálya létrehozás teszt**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:00 |
| **Teszt eredménye** |  |

**10.2.2 Játékos mozgatása aktív elemről csőre, ha a cső nem foglalt**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:05 |
| **Teszt eredménye** |  |

**10.2.3 Játékos mozgatása aktív elemről csőre, ha a cső foglalt.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:08 |
| **Teszt eredménye** |  |

**10.2.4 Játékos mozgatása aktív elemről csőre, ha a cső nem foglalt és csúszós**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:01 |
| **Teszt eredménye** |  |

**10.2.5 Játékos mozgatása csőről aktív elemre, úgy hogy az aktív elemen áll egy másik játékos**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:03 |
| **Teszt eredménye** |  |

**10.2.6 Játékos mozgatása csőről aktív elemre, ha cső ragadós**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:05 |
| **Teszt eredménye** |  |

**10.2.7 Játékos mozgatása aktív elemről olyan csőre, amelynek egyik vége le van csatolva.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:08 |
| **Teszt eredménye** |  |

**10.2.8 Víz átjuttatása forrásból a hozzá csatlakozó nem lyukas, üres csőbe, amelynek másik vége is csatlakoztatva van**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:10 |
| **Teszt eredménye** |  |

**10.2.9 Víz átjuttatása forrásból a hozzá csatlakozó nem lyukas, üres csőbe, amelynek másik vége nincs csatlakoztatva**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:14 |
| **Teszt eredménye** |  |

**10.2.10 Víz átjuttatása forrásból a hozzá csatlakozó lyukas, üres csőbe, amelynek másik vége is csatlakoztatva van**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:17 |
| **Teszt eredménye** |  |

**10.2.11 Víz átjuttatása forrásból a hozzá csatlakozó nem lyukas, teli csőbe, amelynek másik vége is csatlakoztatva van**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:18 |
| **Teszt eredménye** |  |

**10.2.12 Víz átjuttatása ciszternába egy hozzá csatlakozó csőből.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:19 |
| **Teszt eredménye** |  |

**10.2.13 Nem csúszós, nem ragadós cső lyukasztása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:19 |
| **Teszt eredménye** |  |

**10.2.14 Ragadós cső lyukasztása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:20 |
| **Teszt eredménye** |  |

**10.2.15 Lyukas cső lyukasztása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:21 |
| **Teszt eredménye** |  |

**10.2.16 Foltozott cső lyukasztása a megadott időn belül**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:23 |
| **Teszt eredménye** |  |

**10.2.17 Lyukas cső megjavítása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:25 |
| **Teszt eredménye** |  |

**10.2.18 Jó állapotban lévő cső megjavítása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:27 |
| **Teszt eredménye** |  |

**10.2.19 Hibás pumpa megjavítása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:30 |
| **Teszt eredménye** |  |

**10.2.20 Jó állapotban lévő pumpa megjavítása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:33 |
| **Teszt eredménye** |  |

**10.2.21 Működő pumpa vizet pumpál, a bemeneti cső nem üres. A pumpa több vizet tartalmaz, mint a kimeneti cső kapacitása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:35 |
| **Teszt eredménye** |  |

**10.2.22 Működő pumpa vizet pumpál, a bemeneti cső üres**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:39 |
| **Teszt eredménye** |  |

**10.2.23 Működő pumpa vizet pumpál, a bemeneti cső nem üres. A pumpa kevesebb vizet tartalmaz, mint a kimeneti cső kapacitása.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:43 |
| **Teszt eredménye** |  |

**10.2.24 Működő pumpa vizet pumpál, a bemeneti cső és a pumpa tartálya is üres.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:46 |
| **Teszt eredménye** |  |

**10.2.25**  **Pumpa vizet pumpál, a bemeneti csőben több víz van, mint ami a pumpa tartályába fér.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:52 |
| **Teszt eredménye** |  |

**10.2.26 Pumpa vizet pumpál, a kimeneti cső tele van.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 13:56 |
| **Teszt eredménye** |  |

**10.2.26 Cső lecsatlakoztatása, a cső be- vagy kimeneti cső**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:05 |
| **Teszt eredménye** |  |

**10.2.27 Cső lecsatlakoztatása, a cső be- vagy kimeneti cső**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:05 |
| **Teszt eredménye** |  |

**10.2.28 Pumpa lehelyezése csőre, a cső nem lyukas, a játékosnál van pumpa.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:10 |
| **Teszt eredménye** |  |

**10.2.29 Pumpa lehelyezése csőre, a cső lyukas, a szerelőnél van pumpa.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:14 |
| **Teszt eredménye** |  |

**10.2.30 Pumpa lehelyezése csőre, a cső nem lyukas, a szerelőnél nincs pumpa.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:18 |
| **Teszt eredménye** |  |

**10.2.31 Pumpa felvétele ciszternánál, a szerelőnél nincsen pumpa**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:22 |
| **Teszt eredménye** |  |

**10.2.32 Pumpa felvétele ciszternánál, a szerelőnél már van egy pumpa**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:25 |
| **Teszt eredménye** |  |

**10.2.33 Szabad végű cső létrehozása ciszternánál.**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:30 |
| **Teszt eredménye** |  |

**10.2.34 Csővég felvétele ciszternánál, a játékosnál nincsen csővég**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:34 |
| **Teszt eredménye** |  |

**10.2.35 Szabad csővég felvétele ciszternánál, a játékosnál már van egy csővég**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:38 |
| **Teszt eredménye** |  |

**10.2.36 Pumpa bemenetének állítása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:46 |
| **Teszt eredménye** |  |

**10.2.37 Pumpa kimenetének állítása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:52 |
| **Teszt eredménye** |  |

**10.2.38 Cső mindkét végének lecsatolása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 14:56 |
| **Teszt eredménye** |  |

**10.2.39 Csővég letétele**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 15:00 |
| **Teszt eredménye** |  |

**10.2.40 Maximálisnál több cső csatlakoztatása**

| **Tesztelő neve** | Vörös László |
| --- | --- |
| **Teszt időpontja** | 2023.05.14. 15:05 |
| **Teszt eredménye** |  |

* 1. ***Értékelés***

| **Tag neve** | **Tag neptun** | **Munka százalékban** |
| --- | --- | --- |
| Sütheö István | XOBJYX | 20 |
| Vörös László | SLLV0S | 20 |
| Gerencsér Máté | YVSPMN | 20 |
| Bognár Zsombor | BNO3XE | 20 |
| Duy Anh Chu | G4H99J | 20 |

* 1. ***Napló***

| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| --- | --- | --- | --- |
| 2023.05.08. 18:00 | 3,5 óra | Sütheö  Gerencsér  Vörös  Chu  Bognár | Értekezlet.  Döntés: Chu és Bognár elkészítik a program eddigi osztályainak, interfészeinek megfelelő metódusait 05.09. 18:00-ig. Gerencsér felveszi a TestController (tesztorákulum) osztályt, valamint implementálja a bemeneti nyelv függvényeit 05.10. 20:00-ig. Sütheö megírja a tesztesetek bemetetét, valamint a várható kimenteket, valamint az ehhez kapcsolodó tesztorákulum logikát, és a fordítási, futtatási útmutatót 05.12. 20:00-ig. Vörös elvégzi a tesztesetek dokumentálását 05.15. 10:00-ig. Chu elkészíti a fájlista felsorolásást 05.15. 12:00-ig. |
| 2023.05.09. 10:00 | 2 óra | Chu | Tevékenység: Chu implementálja a metódusok játékosokat érintő metódusait. |
| 2023.05.09. 11:30 | 3,5 óra | Bognár | Tevékenység: Bognár implementálja az aktív elemekkel kapcsolatos metódusokat, valamint a játék logikai metódusait. |
| 2023.05.10. 12:00 | 4,5 óra | Gerencsér | Tevékenység: Gerencsér implementálja TestController osztályt, valamint a bemeneti nyelv függvényeit. |
| 2023.05.11. 14:00 | 4 óra | Sütheö | Tevékenység: Elkészíti a be-, és kimeneti txt-ket, valamint implementálja a tesztorákulumot. |
| 2023.05.11. 22:00 | 0,5 óra | Sütheö | Tevékenyésg  Elkészíti a futtatási, fordítási útmutatót. (10.1.2, 10.1.3) |
| 2023.05.14. 13:00 | 3 óra | Vörös | Tevékenység: Elkészíti a tesztek jegyzőkönyveit, dokumentálja azokat. (10.2) |
| 2023.05.15. 10:30 | 1,5 óra | Chu | Tevékenység:  Elkészíti a fájlok felsorolását, valamint azokhoz tartozó táblázat megfelelő részeit kitölti. (10.1.1) |

1. **Grafikus felület specifikációja**
   1. ***A grafikus interfész***

A játék elindítása után a következő menü fogadja a felhasználót:

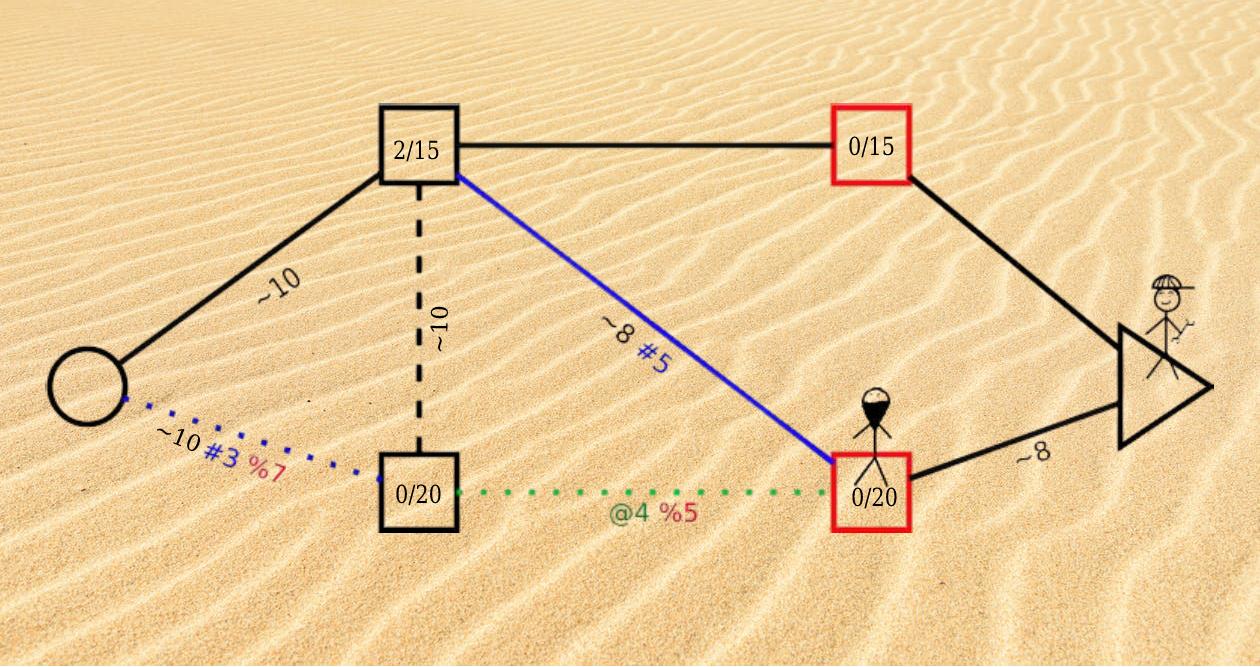


A **Help** gomb megnyomásával a játékról, valamint annak irányításáról, funkcióiról kapunk információkat. Mind a szabotőrök, mind a szerelők lehetséges tevékenységéről is itt kapunk bővebb leírást.

Az **Exit** gomb megnyomása alapvetően a játékból való kilépésre szolgál.

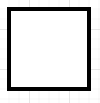
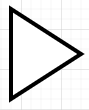
A **Play** gomb lenyomásával tudunk egy új játékot indítani. Ezen belül lesz lehetőségünk különböző előre generált pályákat betölteni, valamint kiválasztani, hogy melyik csapatot szeretnénk erősíteni.

Alapvetően egy pálya a következőképpen fog kinézni:



Az ikonok, valamint a képen megjelenő számok jelentése:

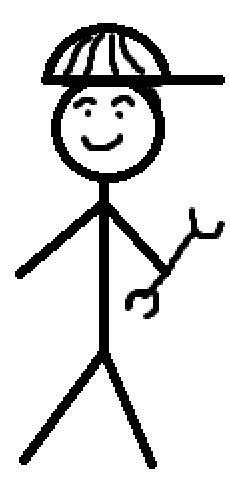
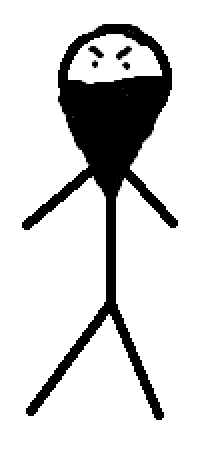
Az egyenes játékelemek ikonjainak jelentése:

 - forrás - pumpa - ciszterna

A vonalak a csöveket jelölik, a különböző típusú (folyamatos, szaggatott, pontozott) vonalak a csövek különböző állapotait tükrözik. A folyamatos a normál, a pontozott a foltozott (ami egy adott ideig nem lyukasztható ki), a szaggatott pedig a kilyukasztott állapotú csövet jelölik.   
A vonalak színe a csövek különböző effektusait jelölik. A fekete szín a normál, a kék szín a csúszós, a zöld szín pedig a ragadós csövet jelölik.

A pumpáknál a fekete szín a normál állapotot, a piros szín pedig az elrontott állapotot jelöli. A pumpákon lévő számértékek közül az első a pumpa által tárolt aktuális vízmennyiséget, a második pedig a pumpa kapacitását jelöli.  
A csövek mellett lévő számok pedig az egyes effektusok időtartalmát, valamint a rajtuk átfolyó vízmennyiséget jelölik. A **~ jel** utáni szám a csövön átfolyó aktuális vízmennyiséget jelöli, a **# jel** a hátralévő fagyasztást mutatja másodpercben, a **@ jel** a hátralévő ragadósságot mutatja másodpercben, a **% jel** a hátralévő “lyukaszthatlanságot” jelöli másodpercben. Ezek a számértékek alapvetően, csak akkor jelennek meg, ha van értelmük, azaz értékük nem 0.

A játékosok ikonjainak jelentése:

- szerelő - szabotőr

A megfelelő játékelemen fognak megjelenni, ezzel jelölve, hogy a játéktéren éppen hol tartózkodnak.

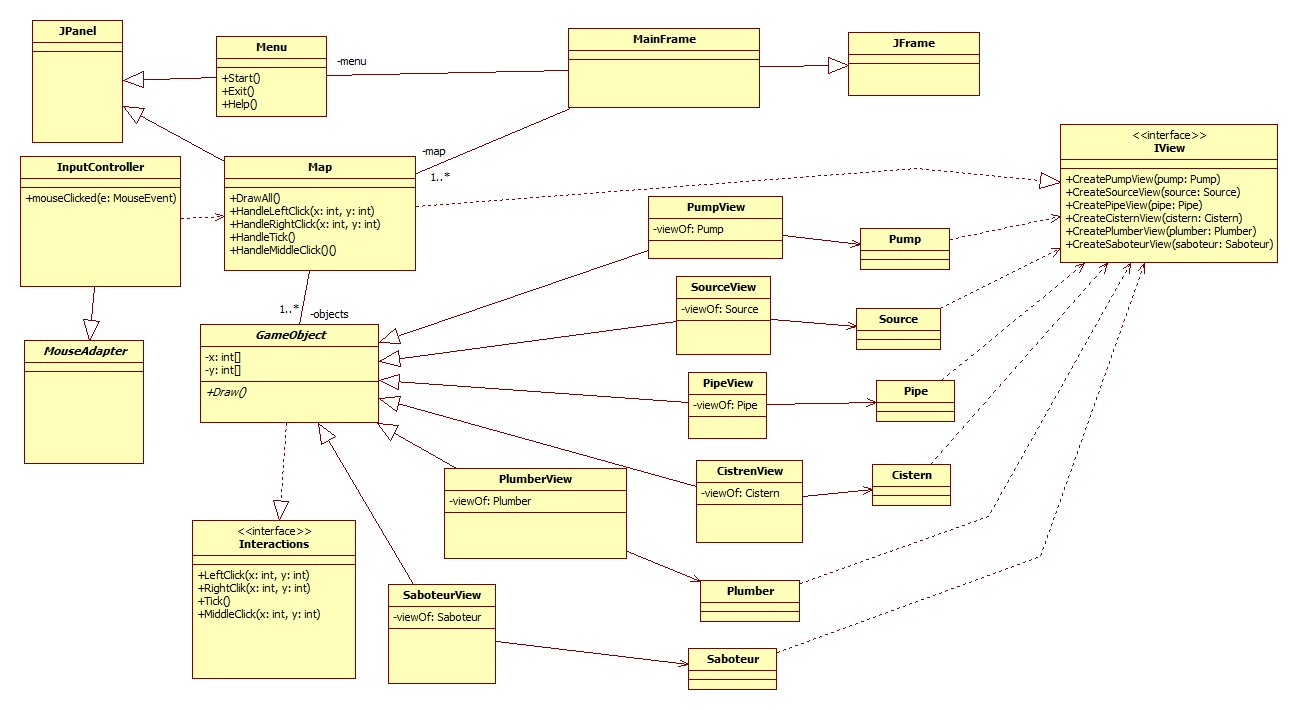
* 1. ***A grafikus rendszer architektúrája***
     1. **A felület működési elve**

A megjelenítés során törekedtünk az MVC megvalósításra. Az alapelv **push** alapú lett, azaz a modell értesíti **Map** osztályt, hogy változott, ekkor újrarajzolja a megfelelő objektumokat.

A kirajzolódó objektumoknak lesz egy grafikus osztálybeli **GameObject** megfelelője, amiből a különböző játékelemek, játékosok grafikus képei fognak leszármazni. Ezeket a **Map** grafikus kezelő osztály eltárolja, és ezeken fogja meghívni a kirajzoló, valamint eseménykezelő műveleteket. Az **InputController** egy MouseListener interfészt megvalósító osztály, ami a beérkező egérműveletek (inputok) elkapásáért felelős. Ha input érkezik, akkor a **Map** osztály megfelelő függvényét meghívva fog az inputok kezelése megtörténni.

A **MainFrame** alapvetően a menü, valamint a játék megjelenítésére szolgál, a **Menu** osztály pedig a menü kirajzolásáért.

* + 1. **A felület osztály-struktúrája**

**A Pump, Source, Pipe, Cistern, Saboteur, Plumber osztályok a prototípus azon osztályai, amelyekhez a grafikus felületet létrehozó osztályok kapcsolódnak. Ezeknek az attribútumait, metódusait külön nem tüntetjük fel a jobb átláthatóság érdekében. A prototípus osztálydiagramján szerepelnek ezek a többi, a grafikus felülethez nem kapcsolodó osztályokkal együtt.

* 1. ***A grafikus objektumok felsorolása***
     1. **MainFrame**
        + **Felelősség**

**Megjeleníti a menüt vagy a pályát.**

* + - * **Ősosztályok**

**JFrame**

* + - * **Interfészek**

**-**

* + - * **Attribútumok**
* **- map: Map**: **a pálya, amint játék zajlik**
* **- menu: Menu** : **a játék menüje**
  + - * **Metódusok**
* **-**
  + 1. **Menu**
       - **Felelősség**

**A menü megjelenítése és a hozzá tartozó akciók(kilépés, játék indítása, stb…) kezelése.**

* + - * **Ősosztályok**

**JPanel**

* + - * **Interfészek**

**-**

* + - * **Attribútumok**

**-**

* + - * **Metódusok**
* **+void Start()**: **elindítja a játékot**
* **+void Exit()**: **bezárja a programot**
* **+void Help()**: **információkat jelenít meg a játékról, valamint a játékmenetről**
  + 1. **Map**
       - **Felelősség**

**A pálya megjelenítése.**

* + - * **Ősosztályok**

**JPanel**

* + - * **Interfészek**

**-**

* + - * **Attribútumok**

**-objects[]: GameObject: a pályán elhelyezkedő objektumok(pumpák, csövek, játékosok, stb…)**

* + - * **Metódusok**
* **+void DrawAll()**: **kirajzolja a pálya objektumait**
* **+void HandelLeftClick()**: **meghívja a tárolt objektumok LeftClick() függvényét**
* **+void HandelRightClick()**: **meghívja a tárolt objektumok RightClick() függvényét**
* **+void Tick()**: **meghívja a tárolt objektumok Tick() függvényét**
  + 1. **GameObject**
       - **Felelősség**

**Egy adott objektum megjelenítése a pályán**

* + - * **Ősosztályok**

**-**

* + - * **Interfészek**

**Drawable**

**Interact**

* + - * **Attribútumok**

**-x: int[]: az objektum sarkainak x koordinátái**

**-y: int[]: az objektum sarkainak y koordinátái**

* + - * **Metódusok**

**-**

* + 1. **Drawable**
       - **Felelősség**

**Az objektumok kirajzolhatósága.**

* + - * **Ősosztályok**

**-**

* + - * **Interfészek**

**-**

* + - * **Attribútumok**

**-**

* + - * **Metódusok**

**+void Draw(): kirajzolja az objektumot**

* + 1. **Interactions**
* **Felelősség**

**Az objektumokon végrehajtott akciók lekezelése.**

* **Ősosztályok**

**-**

* **Interfészek**

**-**

* **Attribútumok**

**-**

* + - * **Metódusok**
* **+void LeftClick(int x, int y): Ha a paraméterként x, y koordinátájú pont az objektumon belül helyezkedik el, akkor végrehajtja az objektum megfelelő műveleteit.**
* **+void RightClick(int x, int y): Ha a paraméterként x, y koordinátájú pont az objektumon belül helyezkedik el, akkor végrehajtja az objektum megfelelő műveleteit.**
* **+void Tick(): Végrehajtja az objektum megfelelő műveleteit.**
  + 1. **SaboteurView**
* **Felelősség**

**Szabotőr megjelenítése.**

* **Ősosztályok**

**GameObject**

* **Interfészek**

**-**

* **Attribútumok**

**-viewOf: Saboteur: az a Saboteur példány, akit megjelenít**

* + - * **Metódusok**

**-**

* + 1. **PlumberView**
* **Felelősség**

**Plumber megjelenítése.**

* **Ősosztályok**

**GameObject**

* **Interfészek**

**-**

* **Attribútumok**

**-viewOf: Plumber: az a Plumber példány, akit megjelenít**

* + - * **Metódusok**

**-**

* + 1. **CisternView**
* **Felelősség**

**Ciszterna megjelenítése.**

* **Ősosztályok**

**GameObject**

* **Interfészek**

**-**

* **Attribútumok**

**-viewOf: Cistern: az a Cistern példány, akit megjelenít**

* + - * **Metódusok**

**-**

* + 1. **SourceView**
* **Felelősség**

**Forrás megjelenítése.**

* **Ősosztályok**

**GameObject**

* **Interfészek**

**-**

* **Attribútumok**

**-viewOf: Source: az a Source példány, akit megjelenít**

* + - * **Metódusok**

**-**

* + 1. **PumpView**
* **Felelősség**

**Pumpa megjelenítése.**

* **Ősosztályok**

**GameObject**

* **Interfészek**

**-**

* **Attribútumok**

**-viewOf: Pump: az a Pump példány, akit megjelenít**

* + - * **Metódusok**

**-**

* + 1. **PipeView**
* **Felelősség**

**Cső a megjelenítése.**

* **Ősosztályok**

**GameObject**

* **Interfészek**

**-**

* **Attribútumok**

**-viewOf: Pipe: az a Pipe példány, akit megjelenít**

* + - * **Metódusok**

**-**

* + 1. **InputController**
* **Felelősség**

**A egérkattintások kezelése.**

* **Ősosztályok**

**-**

* **Interfészek**

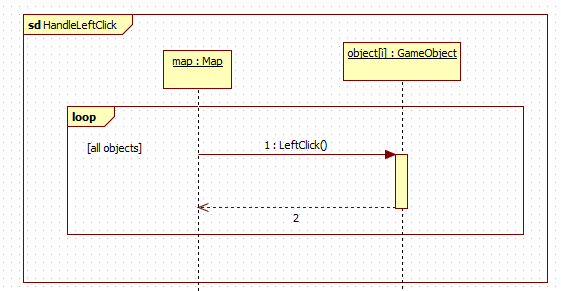
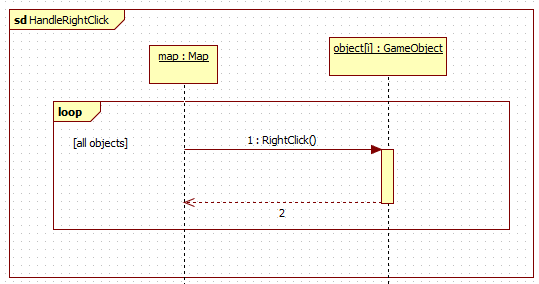
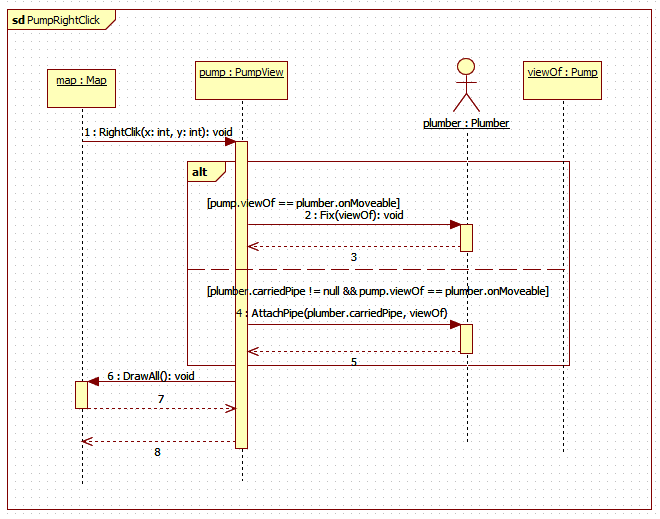
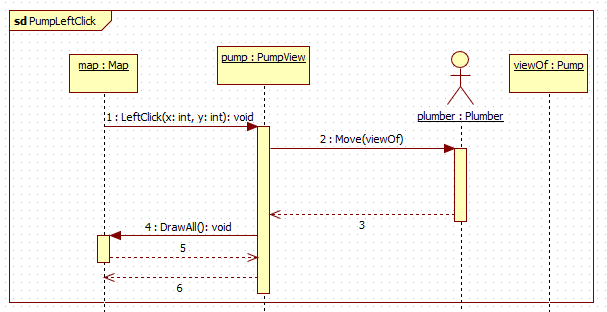
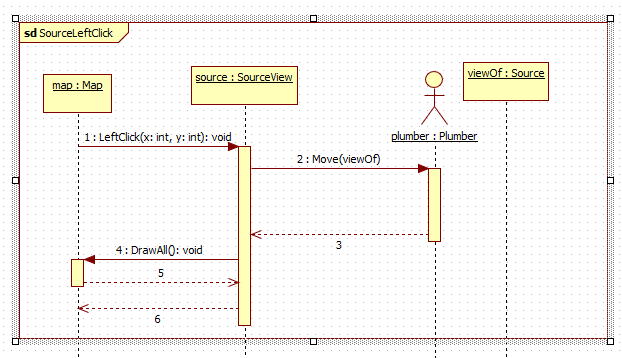
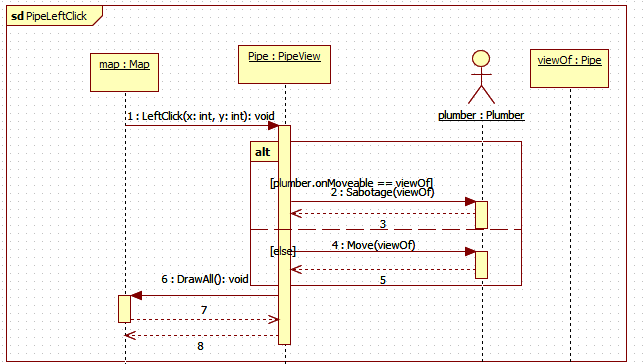
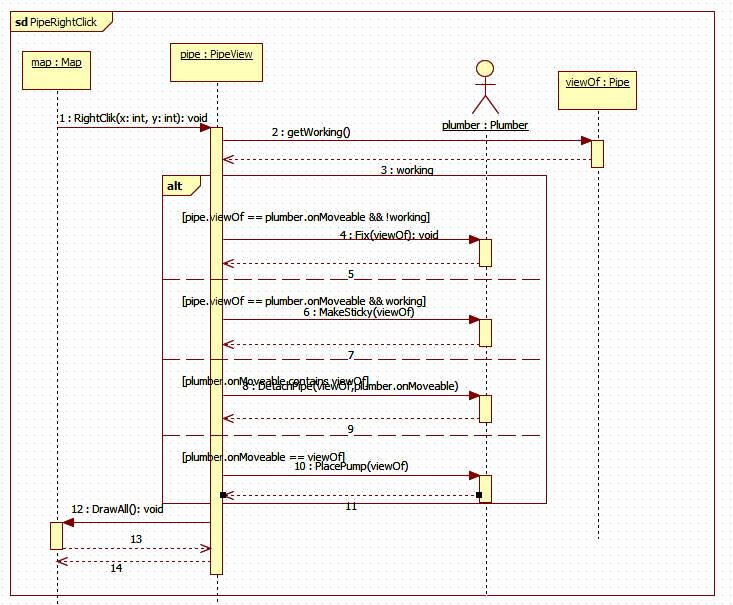
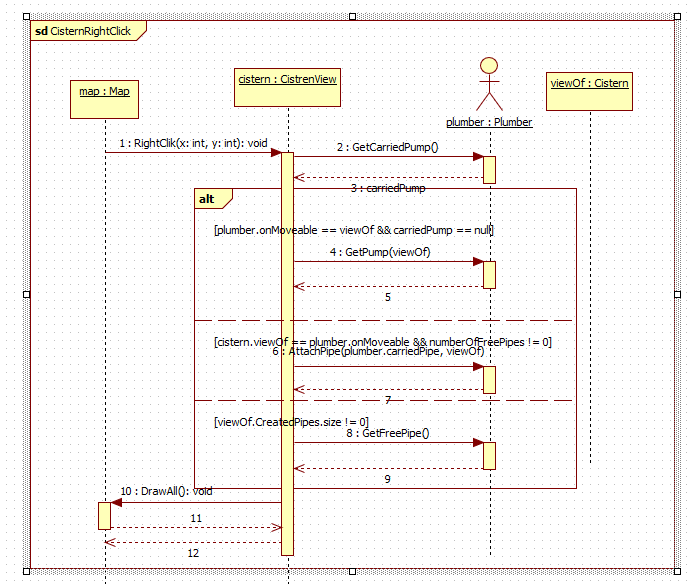
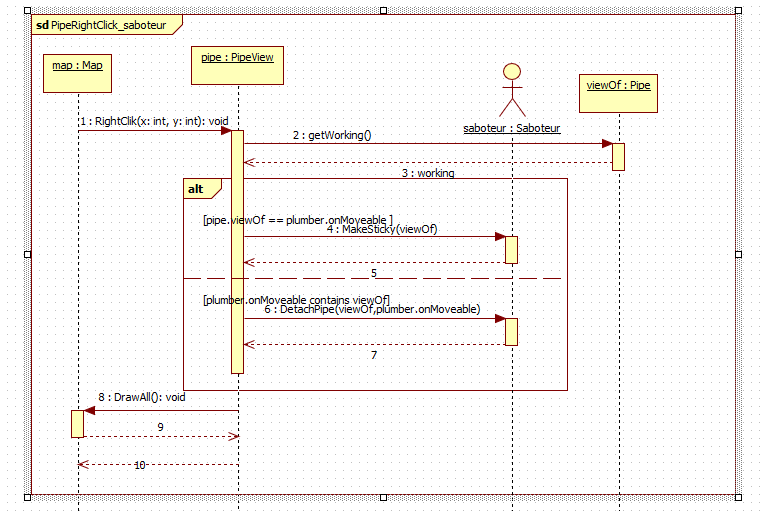
**MouseListener**

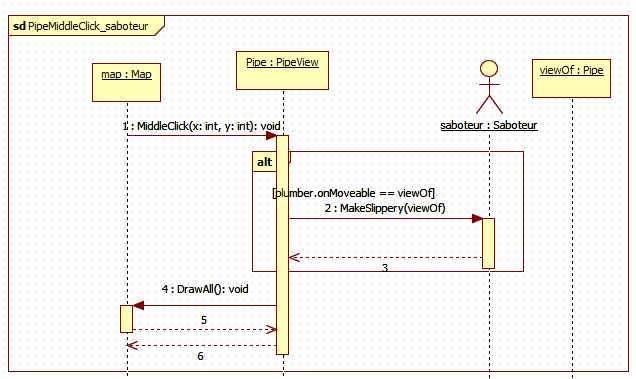
* **Attribútumok**

**-**

* + - * **Metódusok**

**+**[**mouseClicked**](https://docs.oracle.com/javase/8/docs/api/java/awt/event/MouseListener.html#mouseClicked-java.awt.event.MouseEvent-)**(**[**MouseEvent**](https://docs.oracle.com/javase/8/docs/api/java/awt/event/MouseEvent.html) **e): lenyomott egérgombtól függően meghívja az Map osztály megfelelő metódusát**

* 1. ***Kapcsolat az alkalmazói rendszerrel***
     1. ***DrawAll  
        ***
     2. ***HandleLeftClick  
        ***
     3. ***HandleRightClick  
        ***
     4. ***PumpRightClick  
        ***
     5. ***PumpLeftClick  
        ***
     6. ***SourceLeftClick  
        ***
     7. ***PipeLeftClick  
        ***
     8. ***PipeRightClick  
        ***
     9. ***CisternLeftClick***
     10. **CisternRightClick  
         **
     11. **PipeRightClick Saboteur  
         **
     12. **PipeMiddleClick Saboteur**

****

* 1. ***Napló***

| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| --- | --- | --- | --- |
| 2023.05.17. 19:00 | 4 óra | Sütheö  Vörös  Gerencsér  Chu  Bognár | Értekezlet.  Döntés:  Sütheö elkészíti a grafikus interfész általános leírását, a menü-, valamint a pályaképet, valamint, hogy mi mit jelent az adott képeken 05.18. 16:00-ig. Gerencsér elkészíti az osztálydiagramot, valamint az osztályok leírását 05.19 16:00-ig. Vörös elkészíti a felület működési elvéről szóló leírást. 05.20 12:00-ig. Chu és Bognár elkészítik a szekvenciadigramokat a grafikus interfészhez kapcsolódóan. |
| 2023.05.18. 10:00 | 2 óra | Sütheö | Tevékenység:  Sütheö elkészíti a menü, valamint a játékmenet képét.  (11.1) |
| 2023.05.18. 13:30 | 1,5 óra | Sütheö | Tevékenység:  Sütheö elkészíti az ikonok leírását a játékelemekhez, valamint a játékosokhoz. (11.1) |
| 2023.05.18. 19:45 | 1,5 óra | Gerencsér | Tevékenység: Gerencsér elkészíti az osztálydiagramot a grafikus felülethez.  (11.2.2) |
| 2023.05.18. 23:00 | 2 óra | Gerencsér | Tevékenység:  Gerencsér elkészíti az osztályok, attribútumaik, metódusaik leírását. (11.3) |
| 2023.05.19. 18:00 | 3 óra | Vörös | Tevékenység: Vörös leírja a felület müködési elvét, a grafikus osztályok felelősségeit, valamint, hogy általánosan hogyan müködik a grafikus felület. Ezenfelül kijavítja az elírt metódus neveket az osztálydiagramon, valamint az osztályleírásban. (11.2.1) |
| 2023.05.20. 14:00 | 3,5 óra | Chu | Tevékenység: Chu elkészíti az általános kirajzoló, bemenet kezelő, pumpa bemenetét kezelő szekvenciadiagramokat.  (11.4.1-11.4.6) |
| 2023.05.20. 21:30 | 3,5 óra | Bognár | Tevékenység:  Bognár elkészíti a cső, ciszterna, forrás bemenetét kezelő szekvenciadiagramokat. (11.4.7-11.4.12) |

**13. Grafikus változat elkészítése**

* 1. ***Fordítási és futtatási útmutató***
     1. **Fájllista**

| **Fájl neve** | **Méret** | **Keletkezés ideje** | **Tartalom** |
| --- | --- | --- | --- |
| Cistern.java | 3516 bytes | 2023.04.15. 12:15 | A ciszterna működéséhez szükséges attribútumok, függvények megírása. |
| CisternView.java | 2491 bytes | 2023.04.24. 16:15 | A ciszterna grafikus megjelenitéséhez szükséges. |
| Controller.java | 18 513 bytes | 2023.04.15. 18:12 | Teszteléshez volt szükséges, most már kevésbé releváns. |
| Desert.java | 1191 bytes | 2023.04.15. 12:21 | Az elfolyt víz mennyiségét nézi és növeli a szabotőrök pontjait. |
| Drawable.java | 89 bytes | 2023.04.24. 16:20 | A megjelenítéshez szükséges. |
| Pipe.java | 9188 bytes | 2023.04.15 12:24 | A csövek működéséhez szükséges attribútumok, függvények megírása. |
| Player.java | 4958 bytes | 2023.04.15 12:27 | A játékosok lehetséges lépéseinek, műveleteinek megírása. |
| TestController.java | 26 857 bytes | 2023.05.12 14:14 | Teszteléshez szükséges függvények. |
| Element.java | 2869 bytes | 2023.04.15 12:31 | Az elemek működéséhez, állapotához szükséges függvények megírása. |
| Fixable.java | 264 bytes | 2023.04.15 12:34 | Az elemek működéséhez, állapotához szükséges függvények deklarálása. |
| Moveable.java | 376 bytes | 2023.04.15 12:36 | A mozgáshoz szükséges függvények deklarálása. |
| Plumber.java | 2903 bytes | 2023.04.15 12:40 | A szerelők lehetséges lépéseinek, műveleteinek megírása. |
| Pump.java | 4429 bytes | 2023.04.15 12:43 | A csövek működéséhez szükséges attribútumok, függvények megírása. |
| Saboteur.java | 510 bytes | 2023.04.15. 12:41 | A szabotőrök lehetséges lépéseinek, műveleteinek megírása. |
| Source.java | 909 bytes | 2023.04.15 12:45 | A forrás működéséhez szükséges attribútumok, függvények megírása. |
| Functioncall.java | 2366 bytes | 2023.04.15. 18:17 | Teszteléshez kiírásához volt szükséges, most már kevésbé releváns. |
| Game.java | 1103 bytes | 2023.04.15 11:50 | Játék indítása, befejezése, játékosok csatlakozása. |
| PointCounter | 210 bytes | 2023.04.15 13:22 | Pontokat számoló függvények deklarálása. |
| PumpController.java | 149 bytes | 2023.04.15. 13:26 | Pumpa elrontása függvény deklarálása. |
| Stepable.java | 157 bytes | 2023.04.15. 12:50 | Lépés függvény deklarálása. |
| WaterFlow.java | 107 bytes | 2023.04.15. 12:54 | Vízfolyás függvény deklarálása. |
| NoFreePipeCisternException.java | 238 bytes | 2023.05.13. 00:00 | Kivételdobás hibás működés esetén. |
| NoPickedupPipeException.java | 219 bytes | 2023.05.13. 00:05 | Kivételdobás hibás működés esetén. |
| NoPumpPickedupException.java | 234 bytes | 2023.05.13. 00:08 | Kivételdobás hibás működés esetén. |
| PipeAlreadyPickedupException.java | 214 bytes | 2023.05.13. 00:12 | Kivételdobás hibás működés esetén. |
| PipeGetEndpointException.java | 242 bytes | 2023.05.13. 00:14 | Kivételdobás hibás működés esetén. |
| PipePlaceException.java | 230 bytes | 2023.05.13. 00:17 | Kivételdobás hibás működés esetén. |
| PipeRemoveEndpointException.java | 269 bytes | 2023.05.13. 00:20 | Kivételdobás hibás működés esetén. |
| PipeSabotageableException.java | 259 bytes | 2023.05.13. 00:22 | Kivételdobás hibás működés esetén. |
| PipeSetEndpointException.java | 255 bytes | 2023.05.13. 00:25 | Kivételdobás hibás működés esetén. |
| PipeSlipperyException.java | 216 bytes | 2023.05.13. 00:29 | Kivételdobás hibás működés esetén. |
| PipeStickyException.java | 211 bytes | 2023.05.13. 00:33 | Kivételdobás hibás működés esetén. |
| PumpAlreadyPickedupException.java | 249 bytes | 2023.05.13. 00:36 | Kivételdobás hibás működés esetén. |
| PumpInputException.java | 236 bytes | 2023.05.13. 00:40 | Kivételdobás hibás működés esetén. |
| PumpOutputException.java | 242 bytes | 2023.05.13. 00:43 | Kivételdobás hibás működés esetén. |
| GameMap.java | 5201 bytes | 2023.04.24. 16:31 | Játék grafikus megjelenítéshez és a pálya felépítéséhez szükséges. |
| GameObject.java | 1138 bytes | 2023.04.24. 16:38 | Játék grafikus megjelenítéshez szükséges. |
| HelpPanel.java | 2645 bytes | 2023.04.26. 12:40 | Játék rövid műkődését leíró panel. |
| InputController.java | 1091 bytes | 2023.04.24. 16:55 | Bemenetek, inputok vizsgálata. |
| Interactions.java | 725 bytes | 2023.04.24. 17:02 | Bemenetek, inputok absztarkt osztálya. |
| IView.java | 309 bytes | 2023.04.24. 17:02 | Játék grafikus megjelenítéshez szükséges. |
| MainFrame.java | 603 bytes | 2023.04.24. 15:55 | Játék grafikus megjelenítéshez szükséges, itt található a main, innen indul a program. |
| Menu.java | 6068 bytes | 2023.04.25. 8:02 | A játék menüje belépés után. |
| PipeView.java | 8656 bytes | 2023.04.24. 17:17 | A csövek grafikus megjelenitéséhez szükséges. |
| PlumberView.java | 1376 bytes | 2023.04.24. 17:22 | A szerelők grafikus megjelenitéséhez szükséges. |
| PumpView.java | 2312 bytes | 2023.04.24. 17:31 | A pumpa grafikus megjelenitéséhez szükséges. |
| SaboteurView.java | 1650 bytes | 2023.04.24. 17:27 | A szabotor grafikus megjelenitéséhez szükséges. |
| SourceView.java | 1667 bytes | 2023.04.24. 17:39 | A forrás grafikus megjelenitéséhez szükséges. |

* + 1. **Fordítás és telepítés**

Először is csomagoljuk ki a zip fájlban található .java fájlokat tartozó mappát egy **Graf** nevű mappába az **Asztalra** (rakhatjuk bármilyen más mappába is, csak a **cd** paranccsal adjuk meg a mappa elérési útját). Ezután indítsunk egy Command Prompt -ot (cmd.exe) és írjuk be ezt a két sort:

*cd C:\Users\cloud\Desktop\Graf*

Cistern.java CisternView.java Controller.java Desert.java Drawable.java Pipe.java Player.java TestController.java Element.java Fixable.java Moveable.java Plumber.java Pump.java Saboteur.java Source.java Functioncall.java Game.java PointCounter.java PumpController.java Stepable.java WaterFlow.java NoFreePipeCisternException.java NoPickedupPipeException.java NoPumpPickedupException.java PipeAlreadyPickedupException.java PipeGetEndpointException.java PipePlaceException.java PipeRemoveEndpointException.java PipeSabotageableException.java PipeSetEndpointException.java PipeSlipperyException.java PipeStickyException.java PumpAlreadyPickedupException.java PumpInputException.java PumpOutputException.java GameMap.java GameObject.java HelpPanel.java InputController.java Interactions.java IView.java MainFrame.java Menu.java PipeView.java PlumberView.java PumpView.java SaboteurView.java SourceView.java

* + 1. **Futtatás**

Ha nem zártuk be a Command Prompt -t (cmd.exe) akkor elég ezt az egy sort beírni:

*java MainFrame*

A parancs kiadása után elindul a játék.

* 1. ***Értékelés***

| **Tag neve** | **Tag neptun** | **Munka százalékban** |
| --- | --- | --- |
| Sütheö István | XOBJYX | 20% |
| Vörös László | SLLV0S | 20% |
| Gerencsér Máté | YVSPMN | 20% |
| Bognár Zsombor | BNO3XE | 20% |
| Duy Anh Chu | G4H99J | 20% |

* 1. ***Napló***

| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| --- | --- | --- | --- |
| 2023.05.24. 18:00 | 1,5 óra | Sütheö  Vörös  Gerencsér  Bognár  Chu | Értekezlet.  Döntés: Sütheö elkészíti a program skeleton vázát az osztálydiagram alapján 05.25 12:00-ig  Vörös elkészíti a menüt 05.25. 18:00ig  Bognár elkészíti a pálya ablakát és létrehozza az objektumokhoz tartozó nézeteket 05.25. 15:00ig  Chu megtervezi és elkészíti a pályát 05.26. 16:00ig  Gerencsér implementálja a játékosok mozgásához szükséges függvényeket 05.26.21:00ig |
| 2023.05.25. 9:00 | 2,5 óra | Sütheö | Tevékenység: Sütheö elkészíti a program skeleton vázát |
| 2023.05.25. 14:00 | 4 óra | Vörös | Tevékenység: Vörös elkészíti a menüt. |
| 2023.05.25. 15:00 | 3 óra | Bognár | Tevékenység: Bognár az objektumok kirajzolásához szükséges függvényeket. |
| 2023.05.25. 19:00 | 2,5 óra | Chu | Tevékenység: Chu megtervezi és elkészíti a pályát. |
| 2023.05.26. 12:00 | 3 óra | Gerencsér | Tevékenység: Gerencsér implementálja a játékosok mozgatásához szükséges függvéyneket. |
| 2023.05.27. 10:30 | 1,5 óra | Vörös | Tevékenység: Vörös kisebb módosításokat végez a pálya implementációján. |
| 2023.05.27. 15:00 | 1,5 óra | Sütheö | Tevékenység: Sütheö elvégzi program dokumentálását. |
| 2023.05.28. 16:00 | 1 óra | Gerencsér | Tevékenység: Gerencsér elkészíti a jegyzőkönyvet |

1. **Összefoglalás**
   1. ***A projektre fordított összes munkaidő***

| **Tag neve** | **Munkaidő (óra)** |
| --- | --- |
| *Sütheö István* | 90,5 |
| *Vörös László* | 85 |
| *Gerencsér Máté* | 92 |
| *Bognár Zsombor* | 87,5 |
| *Duy Ahn Chu* | 86 |
| **Összesen** | 438 |

***• A feltöltött programok forrássorainak száma***

| **Fázis** | **Kódsorok száma** |
| --- | --- |
| Szkeleton | 1235 |
| Prototípus | 2250 |
| Grafikus változat | 1476 |
| **Összesen** | 4961 |

* 1. ***• Projekt összegzés***
     1. **Mit tanultak a projektből konkrétan és általában?**

A legnagyobb tanulság az az volt, hogy megfontoltan kell kiválasztani a csapattársakat egy ilyen nagyobb volumenű (féléves) projekt esetében. Figyelembe kell tartani a tagok erősségeit, preferenciáit az egyes feladatrészek kiosztásában, valamint megvalósításában. Fontos, hogy a csapatbeli hierarchia a projekt végighaladása során megmaradjon, hogy a tervezett ütemtervet tartani tudjuk.

* + 1. **Mi volt a legnehezebb és a legkönnyebb?**

A legnehezebb része talán a modell megtervezése volt, nehéz volt egy olyan elméleti tervet létrehozni, amely aztán a gyakorlatban is megfelelően működik.  
A legkönnyebb része az implementálása volt a programnak, jó volt látni, hogy a megtervezett program tényleg az elvártak szerint funkcionál.

* + 1. **Összhangban állt-e az idő és a pontszám az elvégzendő feladatokkal?**

Általánosságban igen, viszont a Prototípus elkészítése rész sokkal több időt vett igénybe, mint bármely másik rész, és ezt a pontszám nem suggalta.

* + 1. **Ha nem, akkor hol okozott ez nehézséget?**

Összességében egyedül a Prototípus elkészítésénél volt ez gond.

* + 1. **Milyen változtatási javaslatuk van?**

Az egyes részek leírása, valamint a sablonok már hétfőn kikerülhetnének, mert hasznos lenne néha, ha az ember abban a két napban előre dolgozhatna, hiszen ígyis a tárgyon kívül sok más beadandónk is volt.

* + 1. **Milyen feladatot ajánlanának a projektre?**

Minecraft egyszerűsített változata.

* + 1. **Egyéb kritika és javaslat**

Az elején hozzá kellett szoknunk a precízségedhez, de ezek után teljesen korrektül, megfelelő módon közölted velünk a követelményeket az egyes részeknél. Összességében azt kérted számon tölünk amit elmondtál, valamint sok jó tippet adtál a program elkészítésében. Köszönjük szépen neked is a féléves munkád :).