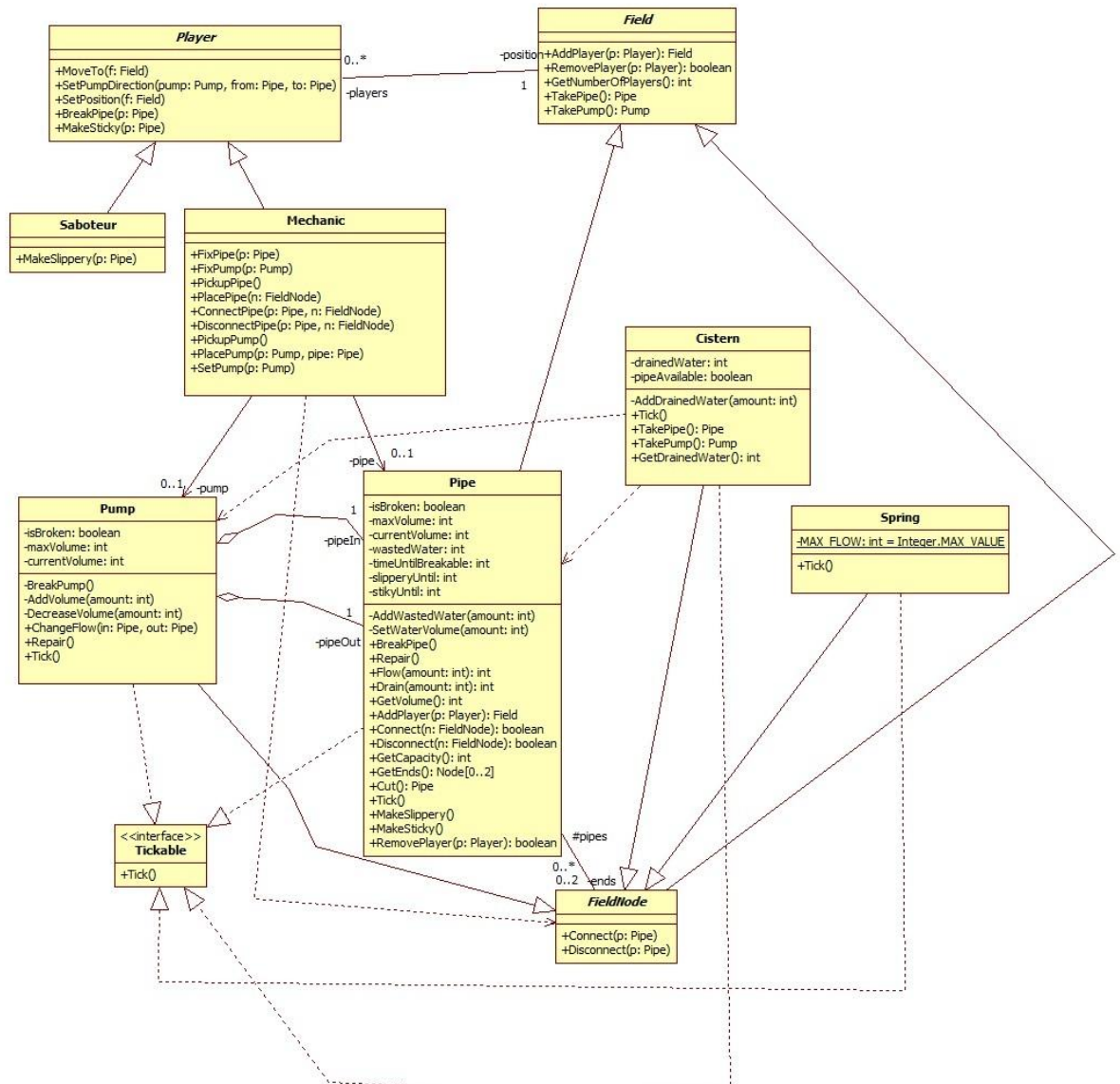


7. Prototípus koncepciója

7.0 Változás hatása a modellre

7.0.1 Módosult osztálydiagram



7.0.2 Új vagy megváltozó metódusok

7.0.2.1 Pipe osztály

- Új **timeUntilBreakable** tagváltozó. Amikor egy adott cső megjavításra kerül, akkor ez az érték egy pozitív értékre lesz beállítva, ami az időt jelképezi, hogy még mennyi van hátra, amíg újra ki lehet lyukasztani. Ha az érték 0, akkor ki lehet lyukasztani a csövet.

- Új **slipperyUntil** tagváltozó. Amikor a csőnek meghívódik a *MakeSlippery()* metódusa, akkor ez a változó beállítódik egy pozitív értékre, ami azt jelképezi, hogy mennyi idő van még hátra abból, hogy a cső csúszós legyen. Ha a változó 0, akkor a cső nem csúszós.
- Új **stickyUntil** tagváltozó. Amikor a csőnek meghívódik a *MakeSticky()* metódusa, akkor ez a változó beállítódik egy pozitív értékre, ami azt jelképezi, hogy mennyi idő van még hátra a cső ragadósságából. Ha a változó 0, akkor a cső nem ragadós.
- A *Pipe* osztály, újonnan implementálja a **Tickable** interfészt. Erre azért volt szükség, hogy minden tickre, a *Pipe*-nak a *Tick()* metódusában mindig csökkentjük egyel a következő változók értékét 0-ig: *timeUntilBreakable*, *slipperyUntil*, *stickyUntil*.
- Új **MakeSlippery()** metódus. Meghívásakor beállítja a *slipperyUntil* tagváltozót.
- Új **MakeSticky()** metódus. Meghívásakor beállítja a *stickyUntil* tagváltozót.
- A *Field* osztály **RemovePlayer(p: Player)** metódusának felüldefiniálása, annak érdekében, hogy a cső ellenőrizni tudja, hogy a játékos elléphet-e vagy sem a csőről.

7.0.2.2 Field osztály

- Az **AddPlayer(p: Player)** metódus többé nem absztrakt. Visszatérési értéke *boolean*-ról megváltozott *Field*-re. A függvény hozzáadja a *players* tagváltozójához a paraméterben kapott játékost, és visszatéríti saját magát.
- A **RemovePlayer(p: Player)** visszatérési értéke *boolean* lett. Erre azért volt szükség, mert, hogy ha egy cső ragadós, akkor nem biztos, hogy el lehet róla távolítani a játékost. A függvény, mindig *true*-t térít vissza, illetve a *Pipe* osztály majd felüldefiniálja.

7.0.2.3 Player osztály

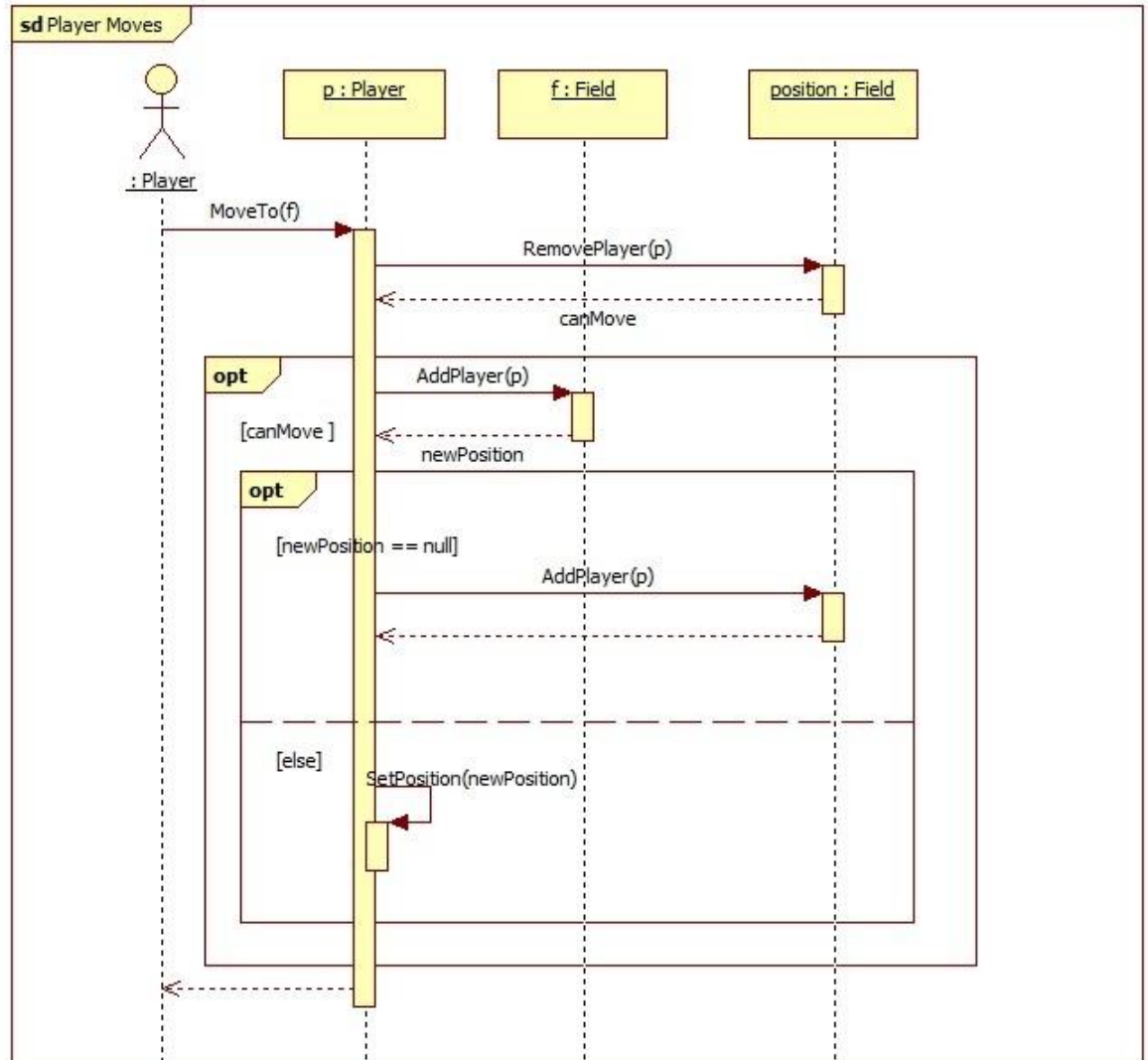
- A *Saboteur* osztály **BreakPipe(p: Pipe)** metódusa átkerült a *Player* osztályba, ezáltal megvalósítottuk azt az új követelményt, hogy a szerelő is ki tud lyukasztani egy csövet.
- Új **MakeSticky(p: Pipe)** metódus, amely a paraméterben kapott csövet ragadóssá teszi.

7.0.2.4 Saboteur osztály

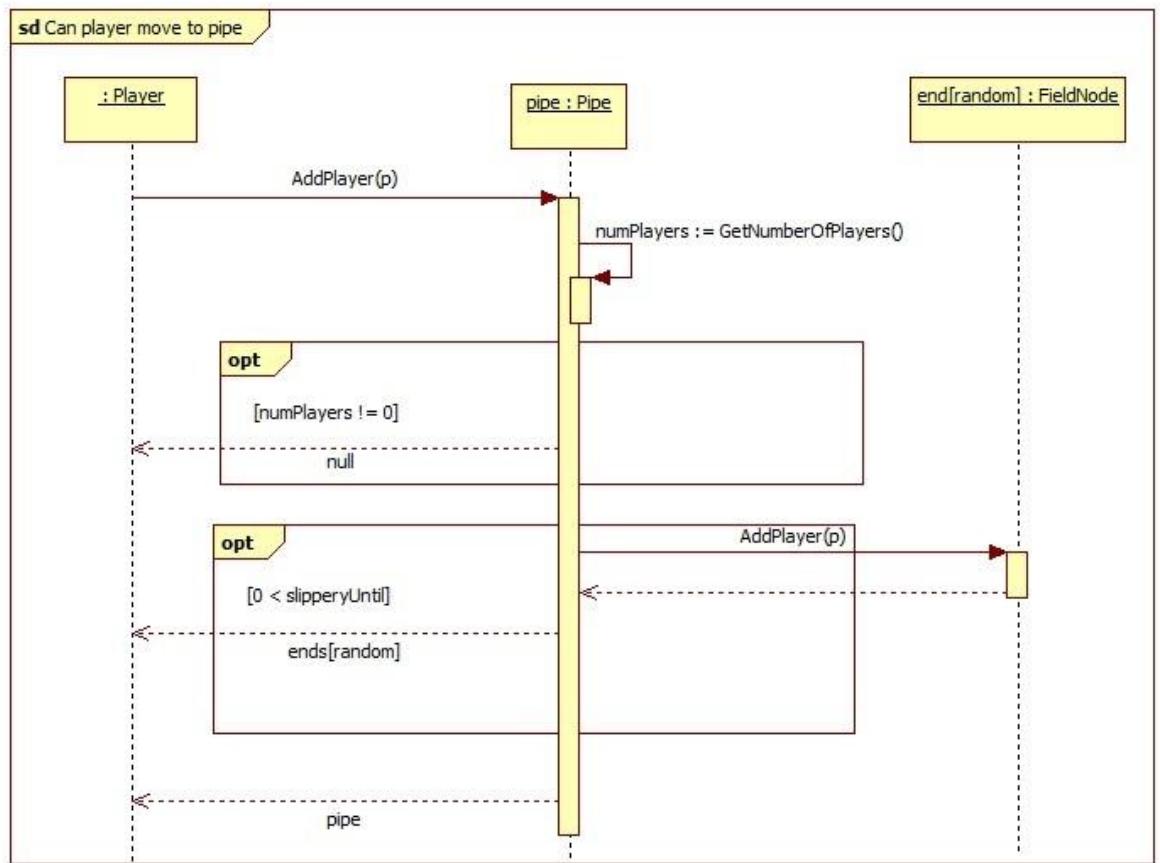
- Új **MakeSlippery(p: Pipe)** metódus, amely a paraméterben kapott csövet csúszóssá teszi.

7.0.3 Szekvencia-diagramok

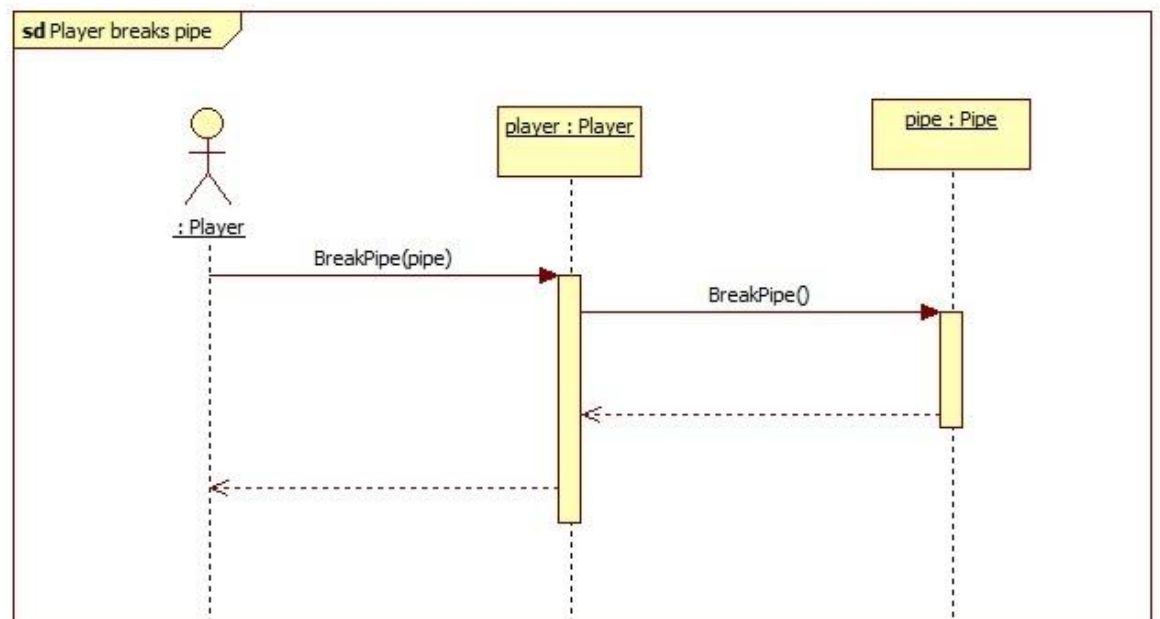
7.0.3.1 Player moves (megváltozott szekvenciadiagram)

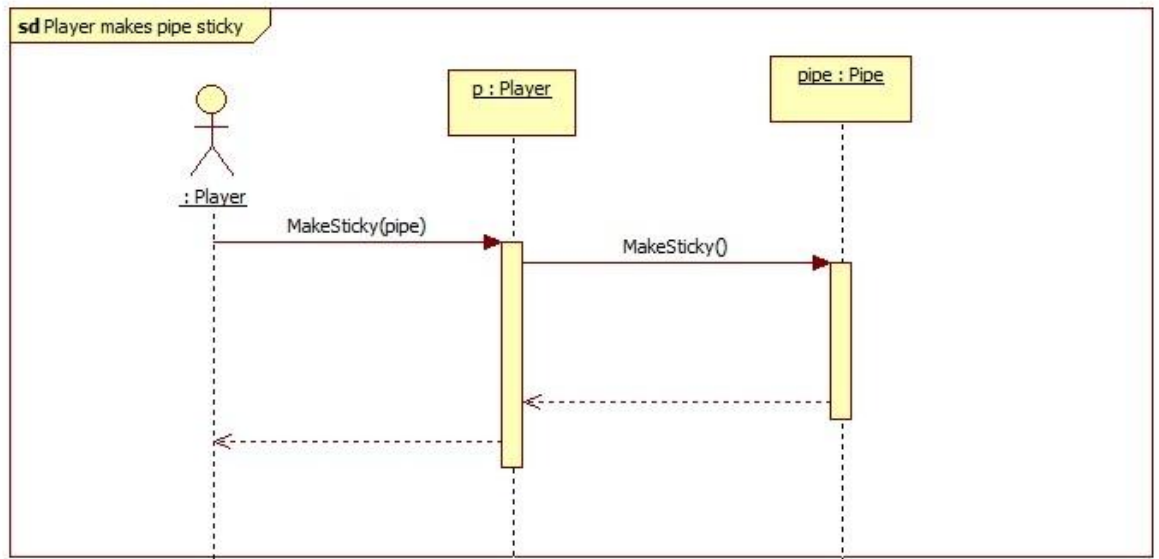
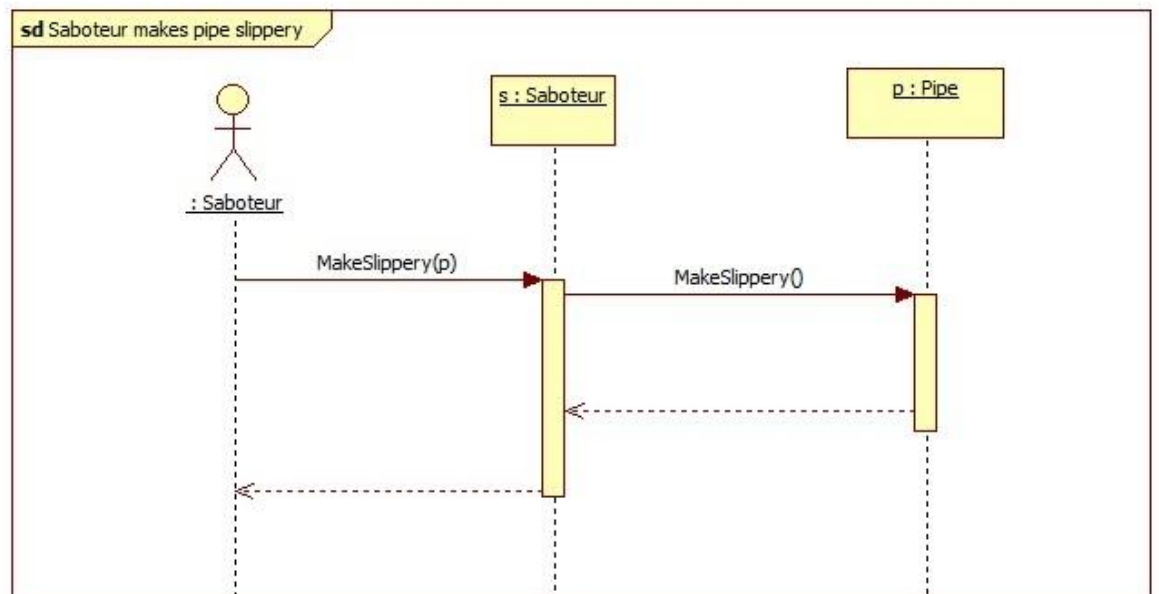


7.0.3.2 Can player move to pipe (megváltozott szekvenciadiagram)

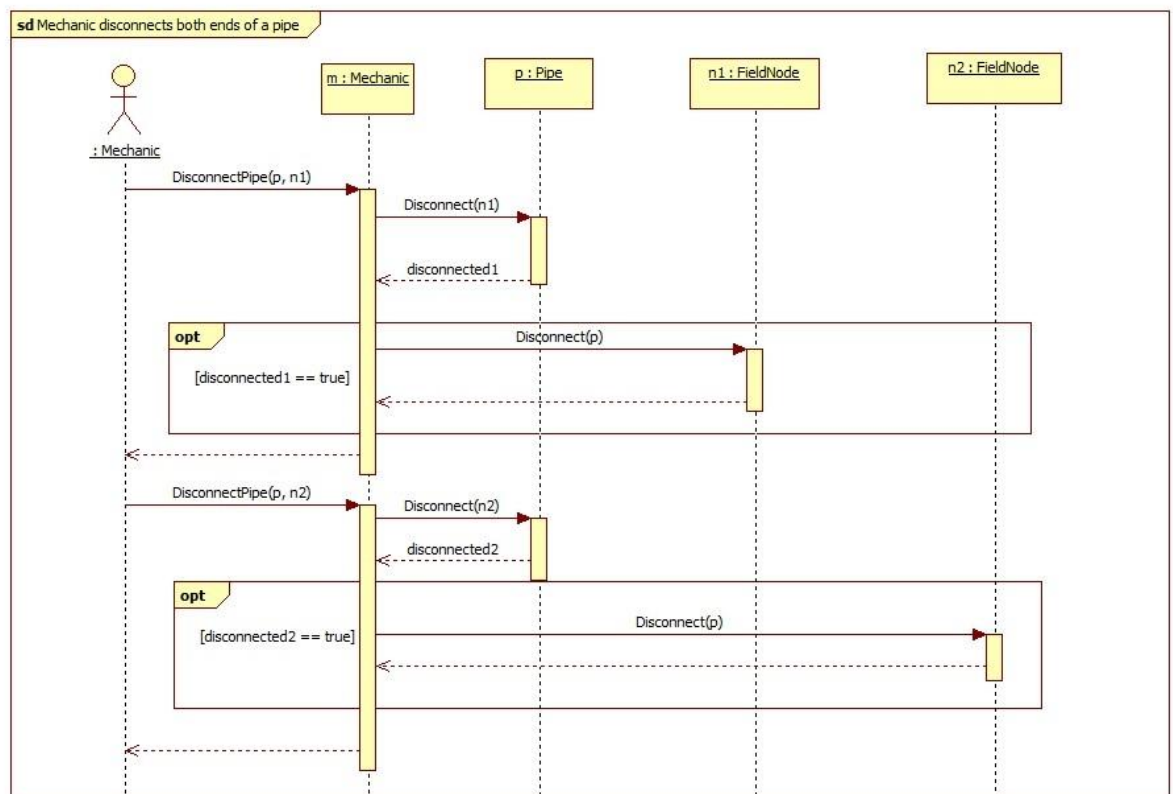


7.0.3.3 Player breaks pipe (eddig szabotőre volt csak, most már bármilyen játékosra igaz)



7.0.3.4 Player makes pipe sticky (új szekvenciadiagram)**7.0.3.5 Saboteur makes pipe slippery (új szekvenciadiagram)**

7.0.3.6 Mechanic disconnects both end of pipe (új szekvenciadiagram)



7.1 Prototípus interface-definíciója

7.1.1 Az interfész általános leírása

A protó program konzolról lesz irányítható, minden sor után azonnal alkalmazva a módosításokat. Az interfész csak a szabványos bemenetről fogad parancsokat, és a szabványos kimenetre ír. Ez lehetővé teszi hogy terminálon tesztelhető legyen, vagy átirányítható legyen ki és bemeneti fájlokra automatikus tesztelés céljából. A tesztet akkor sikeres, ha a parancsok sorozatára a kimenet azonos az elvárt kimenettel, ha eltér akkor pedig sikertelen.

7.1.2 Bemeneti nyelv

Alább definiálva vannak a bemeneti nyelv parancsai. Az egyes parancsokhoz tartozik egy leírás, amelyben röviden kifejtjük a parancs működését. Ezt követően a parancshoz tartozó további opciók, valamint azok jelentései szerepelnek. Amennyiben egy parancs esetén egy adott opció megadása kötelező, ezt "< >" jelekkel jelöltük. Az esetlegesen elhagyható opciók (pl. pályaépítés érdekében) szögletes zárójelekkel vannak jelölve ("["]").

load

Leírás:

Egy pálya betöltésére szolgál. A fájl amiből betölti a pályát olyan parancsokat kell hogy tartalmazzon, amelyek az interfészben definiáltak.

Opciók: load <filename>

- *filename* - A fájl elérési útvonala

Példa:

- load input.txt

save**Leírás:**

A *save start* parancs kiadása után a *save end* parancs kiadásáig, a konzolra írott parancsokat elmenti az *start* parancs beírásakor argumentumként megadott nevű fájlba.

Opciók: save <start> <filename> és save <end>

- *start* - A parancsok mentésének kezdete.
 - *filename* - A fájl elérési útvonala.
- *end* - A mentés vége.

Példa:

- save start output.txt
- save end

random**Leírás:**

A parancs segítségével ki- és bekapcsolható a program véletlent tartalmazó elemeinek véletlenszerűsége. (pl. pumpa véletlen eltörése)

Opciók: random <on / off>

- *on* - A véletlenszerűség bekapcsolása.
- *off* - A véletlenszerűség kikapcsolása.

Példa:

- random on

add**Leírás:**

Felvesz a játékba egy új csövet/pumpát/ciszternát/forrást/szabotőrt/szerelőt a megadott névvel.

Opciók: add <type> <name>

- *type* - a generálni kívánt elem típusa, ez lehet ('*pump*', '*cistern*', '*spring*', '*pipe*', '*saboteur*', '*mechanic*').
- *name* - a generált elem neve

Példa:

- add saboteur Saboteur1

move**Leírás:**

Egy játékost elmozgat a pozíciójáról egy másik pozícióra.

Opciók: move <player_name> <field_name>

- ***player_name*** - a generált Saboteur vagy Mechanic megadott neve
- ***field_name*** - a generált pálya elem neve amire rá szeretnénk lépni

Példa:

- move saboteur1 pipe1

connect**Leírás:**

A megadott nevű szerelő összeköt egy csövet (amelyik rendelkezik legalább egy szabad véggel) és egy csomópontot.

Opciók: connect <pipe_name> <fieldnode_name> [mechanic_name]

- ***pipe_name*** - A csatlakoztatni kívánt cső neve
- ***fieldnode_name*** - A csomópont neve, amelyhez a csövet csatlakoztatjuk.
- ***mechanic_name*** - A szerelő neve. Pályaépítés esetén elhagyható.

Példa:

- connect pipe1 pump1 mechanic1

setpump**Leírás:**

Átállítja a megadott pumpa folyási irányát (mely bemeneti csőből melyik kimeneti csőbe pumpáljon).

Opciók: setpump <pump_name> <pipe_in> <pipe_out> [player_name]

- ***pump_name*** - A pumpa neve, amelyet átállítunk
- ***pipe_in*** - A bemeneti cső neve
- ***pipe_out*** - A kimeneti cső neve
- ***player_name*** - A játékos, aki a pumpát átállítja. Pályaépítés során elhagyható.

Példa:

- setpump pump1 pipe1 pipe2 mechanic1

puncturepipe**Leírás:**

A játékos kilyukasztja a csövet amin áll.

Opciók: puncturepipe <pipe_name> [player_name]

- ***pipe_name*** - A cső neve.
- ***player_name*** - A játékos neve. Elhagyása esetén a cső magától kilyukad.

Példa:

- puncturepipe pipe1 saboteur1

pickuppipe**Leírás:**

A szerelő magához vesz egy csövet, amennyiben előzőleg nem volt nála cső, és a pozícióján van felvehető cső.

Opciók: pickuppipe <mechanic_name>

- *mechanic_name* - A szerelő neve.

Példa:

- pickupipe mechanic1

pickuppump

Leírás: A szerelő magához vesz egy pumpát, amennyiben előzőleg nem rendelkezett pumpával és a pozícióján van elérhető pumpa.

Opciók: pickuppump <mechanic_name>

- *mechanic_name* - A szerelő neve.

Példa:

- pickuppump mechanic1

placepump

Leírás: A szerelő lehelyez egy pumpát a pályára a megadott névvel ellátva.

Opciók: placepump <mechanic_name> <pump_name>

- *mechanic_name* - A szerelő neve
- *pump_name* - A lehelyezett pumpa ezt a nevet kapja.

Példa:

- placepump mechanic1 pump1

placepipe

Leírás: A szerelő lehelyezi a nála lévő csövet, amelynek egyik végét ahhoz a csomóponthoz köti, amelyiken éppen áll.

Opciók: placepipe <mechanic_name> <node_name>

- *mechanic_name* - A szerelő neve
- *node_name* - A csomópont neve

Példa:

- placepipe mechanic1 cistern1

fixpipe

Leírás: A szerelő megjavítja a megadott csövet.

Opciók: fixpipe <pipe_name> [mechanic_name]

- *pipe_name* - A cső neve
- *mechanic_name* - A szerelő neve. Elhagyása esetén a cső magától megjavul.

Példa:

- fixpipe pipe1 mechanic1

fixpump

Leírás: A szerelő megjavítja a megadott pumpát.

Opciók: fixpump <泵_name> [mechanic_name]

- *pump_name* - A pumpa neve
- *mechanic_name* - A szerelő neve.

Példa:

- fixpump pump1 mechanic1

breakpump

Leírás: A megadott pumpa eltörik.

Opciók: breakpump <泵_name>

- *pump_name* - Az eltörni kívánt pumpa neve

Példa:

- breakpump pump1

tick

Leírás: A megadott nevű *Tickable* interfészt implementáló metódust tickeli.

Opciók: tick <name>

- *name* - Az objektum neve, amelyet tickelni szeretnénk.

Példa:

- tick pump1

state

Leírás: Kiírja a megadott nevű pályaelem vagy játékos állapotát. Ha a paraméter nem kap értéket, a játékban szereplő összes objektum állapotát kilistázza.

Opciók: state [name]

- *name* - A pályaelem vagy játékos neve. (Megadása nem kötelező.)

Példa:

- state pipe1

makeslippy

Leírás: A megadott nevű szabotőr csúszóssá változtatja a megadott nevű csövet rövid időre.

Opciók: makeslippy <pipe_name> [saboteur_name]

- *pipe_name* - A cső neve.
- *saboteur_name* - A szabotőr neve. Elhagyása esetén a cső magától csúszóssá válik.

Példa:

- makeslippy pipe1 saboteur1

makesticky

Leírás: A megadott nevű játékos ragadóssá változtatja a megadott nevű csövet rövid időre.

Opciók: makesticky <pipe_name> [player_name]

- *pipe_name* - A cső neve.
- *player_name* - A játékos neve. Elhagyása esetén a cső magától ragadóssá válik.

Példa:

- makesticky pipe1 mechanic1

setcurrentvolume

Leírás: A megadott cső vagy pumpa aktuális víz mennyiségét a megadott értékre állítja.

Opciók: setcurrentvolume <name> <water_volume>

- *name* - A pályaelem neve.
- *water_volume* - Az új vízmennyiség.

Példa:

- setcurrentvolume pump1 33

setmaxvolume

Leírás: A megadott cső vagy pumpa maximális víz mennyiségét a megadott értékre állítja.

Opciók: setmaxvolume <name> <water_volume>

- *name* - A pályaelem neve.
- *water_volume* - Az új vízmennyiség.

Példa:

- setmaxvolume pipe1 125

7.1.3 Kimeneti nyelv

Az egyetlen parancs, amely kimenetet generál a futása során az a *state* parancs. Ennek segítségével a játékot alkotó objektumok teljes állapota ellenőrizhető. Tehát a többi parancs nem generál kimenetet, azok sikeres lefutását a *state* parancs használatával lehet ellenőrizni.

state

Attól függően, hogy milyen típusú objektumra adjuk ki a parancsot, különböző kimenetei lehetnek:

- *Mechanic*

Mechanic <name> on <position> pipe: <name of the pipe / null> pump: <name of the pump / null>

- *Saboteur*

Saboteur <name> on <position>

- *Pipe*

Pipe <name> with ends: <end1 name / null>, <end2 name / null> max volume: <max volume> current volume: <current volume> wasted water: <wasted water> broken: <true /

false> breakable: <remaining time> slippery: <remaining time> sticky: <remaining time>
standing player: <player name / null>

- *Pump*

Pump <name> with ends: <names of connected pipes delimited by “,”> input: <input pipe name> output: <output pipe name> broken: <true / false> max volume: <max volume> current volume: <current volume> standing players: <list of standing player names delimited by “,” / null>

- *Cistern*

Cistern <name> with ends: <names of connected pipes delimited by “,”> pipe available: <true / false> drained water: <drained water> standing players: <list of standing player names delimited by “,” / null>

- *Spring*

Spring <name> with ends: <names of connected pipes delimited by “,”> standing players: <list of standing player names delimited by “,” / null>

7.2 Összes részletes use-case

Use-case neve	1. load
Rövid leírás	A pálya betöltése fájlból.
Aktorok	Tester
Forgatókönyv	1. A pályát betölti az opcióként adott fájlból.

Use-case neve	2. save
Rövid leírás	Fájlba menti a konzolra kiírt parancsokat
Aktorok	Tester
Forgatókönyv	1. A <i>save start</i> parancs kiadása után a <i>save end</i> parancs kiadásáig, a konzolra írt parancsokat elmenti az <i>start</i> parancs beírásakor argumentumként megadott nevű fájlba.

Use-case neve	3. add
Rövid leírás	Hozzáad a megadott típusú elemből egyet a játékhoz.
Aktorok	Tester
Forgatókönyv	1. Hozzáadja a megadott típusú elemet a játékhoz a megadott névvel.

Use-case neve	4. move
Rövid leírás	A megadott játékost átlépteti a megadott mezőre.
Aktorok	Player
Forgatókönyv	1. A megadott játékos átlép a megadott mezőre.

Use-case neve	5. connect
Rövid leírás	Összeköti a csövet a csomóponttal.
Aktorok	Tester, Mechanic

Forgatókönyv	1. A megadott csövet rákötö a megadott csomópontra.
---------------------	---

Use-case neve	6. setpump
Rövid leírás	Átállítja a megadott pumpa folyási irányát.
Aktorok	Tester
Forgatókönyv	1. Beállítja a pumpa ki és bemeneti csövet.

Use-case neve	7. puncturepipe
Rövid leírás	Kilyukasztja a csövet.
Aktorok	Tester, Mechanic, Saboteur
Forgatókönyv	1. Eltöri a megadott csövet.

Use-case neve	8. pickuppipe
Rövid leírás	A szerelő elvesz egy csövet a ciszternától.
Aktorok	Mechanic
Forgatókönyv	1. A Mechanic elvesz egy csövet a ciszternától. 2. A Mechanic beteszi a csövet az eszköztárába.

Use-case neve	9. pickuppump
Rövid leírás	A szerelő elvesz egy pumpát a ciszternától.
Aktorok	Mechanic
Forgatókönyv	1. A Mechanic elvesz egy pumpát a ciszternától. 2. A Mechanic beteszi a pumpát az eszköztárába.

Use-case neve	10. placepump
Rövid leírás	A szerelő leteszi az eszköz tárában lévő pumpát.
Aktorok	Mechanic
Forgatókönyv	1. A szerelő kettévágja a csövet amin éppen áll. 2. Leteszi a pumpát. 3. A két szabad csövet összeköti vele.

Use-case neve	11. placepipe
Rövid leírás	A szerelő leteszi az eszköz tárában lévő csövet.
Aktorok	Mechanic
Forgatókönyv	1. A szerelő kiveszi az eszköztárából a csövet. 2. A cső két végét rákötö a megadott csomópontokra.

Use-case neve	12. fixpipe
Rövid leírás	A szerelő megjavítja a csövet.
Aktorok	Tester, Mechanic
Forgatókönyv	1. Megfoltozza a megadott csövet.

Use-case neve	13. fixpump
Rövid leírás	A szerelő megjavítja a megadott pumpát.
Aktorok	Tester, Mechanic

Forgatókönyv	1. Megjavítja a megadott pumpát.
---------------------	----------------------------------

Use-case neve	14. breakpump
Rövid leírás	Eltöri a pumpát.
Aktorok	Tester
Forgatókönyv	1. Eltöri a paraméterben megadott pumpát.

Use-case neve	15. tick
Rövid leírás	A megadott pályaelemen meghívja a tick metódust.
Aktorok	Tester
Forgatókönyv	1. Meghívja a tick metódust.

Use-case neve	16. state
Rövid leírás	Kiírja a megadott nevű pályaelem vagy játékos állapotát.
Aktorok	Tester
Forgatókönyv	1. Kiírja a megfelelő helyre a pályaelem vagy játékos állapotát.

Use-case neve	17. makeslipperry
Rövid leírás	A megadott nevű szabotőr csúszóssá változtatja a megadott nevű csövet rövid időre.
Aktorok	Tester, Saboteur
Forgatókönyv	1. A szabotőr csúszóssá teszi a csövet.

Use-case neve	18. makesticky
Rövid leírás	Cső ragadóssá változtatása.
Aktorok	Tester, Mechanic, Saboteur
Forgatókönyv	1. A megadott cső ragadóssá válik.

Use-case neve	19. setcurrentvolume
Rövid leírás	Aktuális vízmennyiség beállítása.
Aktorok	Tester
Forgatókönyv	1. A megadott cső vagy pumpa aktuális víz mennyiségét a megadott értékre állítja.

Use-case neve	20. setmaxvolume
Rövid leírás	Maximális vízmennyiség beállítása.
Aktorok	Tester
Forgatókönyv	1. A megadott cső vagy pumpa maximális víz mennyiségét a megadott értékre állítja.

7.3 Tesztelési terv

Teszt-eset neve	1. Pumpa meghibásodás
Rövid leírás	A víz folyása közben, egy pumpa meghibásodik.
Teszt célja	Ellenőrizzük, hogy a meghibásodás után a pumpa viz pumpálása leáll-e.

Teszt-eset neve	2. Lépés üres csőre
Rövid leírás	Egy játékos megpróbálkozik egy csőre lépni.
Teszt célja	Ellenőrizzük hogy a játékos rá tudott-e lépni a csőre.

Teszt-eset neve	3. Lépés foglalt csőre
Rövid leírás	Egy játékos megpróbálkozik egy csőre lépni.
Teszt célja	Ellenőrizzük hogy a játékos nem tudott rálépni a csőre.

Teszt-eset neve	4. Lépés üres pumpára
Rövid leírás	Egy játékos rálép egy pumpára
Teszt célja	Ellenőrizzük hogy a játékos rá tudott-e lépni az üres pumpára.

Teszt-eset neve	5. Lépés foglalt pumpára
Rövid leírás	Egy játékos rálép egy pumpára
Teszt célja	Ellenőrizzük hogy a játékos rá tud-e lépni foglalt pumpára is.

Teszt-eset neve	6. Lépés ciszternára
Rövid leírás	Egy játékos rálép egy ciszternára.
Teszt célja	Ellenőrizzük hogy a játékos rá tudott-e lépni a ciszternára.

Teszt-eset neve	7. Lépés csúszós csőre
Rövid leírás	Egy játékos rálép egy csúszós csőre.
Teszt célja	Ellenőrizzük hogy a játékos elcsúszik a csövön.

Teszt-eset neve	8. Ragadós csőről lelépés
Rövid leírás	Egy játékos lelép egy ragadós csőről.
Teszt célja	Ellenőrizzük hogy tényleg nem tudott lelépni a csőről a játékos.

Teszt-eset neve	9. Lépés forrásra
Rövid leírás	Egy játékos rálép egy forrásra.
Teszt célja	Ellenőrizzük hogy a játékos tényleg rá tud lépni a forrásra.

Teszt-eset neve	10. Cső egyik végének lecsatlakoztatása
Rövid leírás	Egy szerelő lecsatlakoztatja a cső egyik végét.
Teszt célja	Ellenőrizzük hogy lehet-e cső aminek egyik vége szabad.

Teszt-eset neve	11. Cső mindkét végének lecsatlakoztatása
Rövid leírás	Egy szerelő lecsatlakoztatja a cső mindkét végét.
Teszt célja	Ellenőrizzük hogy lehet-e cső aminek mindkét vége szabad.

Teszt-eset neve	12. Szerelő átcsatlakoztat egy csövet egyik pumpáról egy másikra.
Rövid leírás	Egy szerelő lecsatlakoztat egy csövet egy pumpáról, majd átcsatlakoztatja a szabad véget egy másik pumpára.
Teszt célja	Ellenőrizzük hogy a szerelő valóban átcsatlakoztathat-e egy csövet egy pumpáról egy másikra.

Teszt-eset neve	13. Szerelő csövet vesz fel a ciszternától sikeresen
Rövid leírás	A szerelő felvesz egy csövet a ciszternától.
Teszt célja	Ellenőrizzük hogy a felvett cső megjelent-e a szerelő eszköztárában.

Teszt-eset neve	14. Szerelő csövet vesz fel a ciszternától sikertelenül
Rövid leírás	A szerelő megpróbál felvenni egy csövet a ciszternától, de ott nincsen cső.
Teszt célja	Ellenőrizzük, hogy a szerelő ilyen esetben valóban nem tud új csövet magához venni.

Teszt-eset neve	15. Szerelő csövet próbál felvenni a forrásnál
Rövid leírás	A szerelő megpróbál felvenni egy csövet a forrásnál.
Teszt célja	Ellenőrizzük hogy valóban csak a ciszternánál lehet csövet felvenni.

Teszt-eset neve	16. Szerelő pumpát vesz fel a ciszternától sikeresen
Rövid leírás	A szerelő felvesz egy pumpát a ciszternától.
Teszt célja	Ellenőrizzük hogy a felvett pumpa megjelent-e a szerelő eszköztárában.

Teszt-eset neve	17. Szerelő pumpát vesz fel a ciszternától sikertelenül
Rövid leírás	A szerelő megpróbál felvenni a ciszternától, de már van nála pumpa.
Teszt célja	Ellenőrizzük hogy a szerelő valóban egyszerre csak egy pumpával rendelkezhet.

Teszt-eset neve	18. Szerelő pumpát próbál felvenni a forrásnál
Rövid leírás	A szerelő megpróbál a forráson állva pumpát felvenni.
Teszt célja	Ellenőrizzük, hogy a szerelő valóban csak a ciszternánál tud pumpát felvenni.

Teszt-eset neve	19. Szabotőr csövet lyukaszt
------------------------	------------------------------

Rövid leírás	Egy Szabotőr kilyukasztja-e a csövet amin áll.
Teszt célja	Ellenőrizzük, hogy a szabotőr sikeresen kilyukasztotta-e a csövet.

Teszt-eset neve	20. Szabotőr már megfoltozott csövet lyukaszt
Rövid leírás	Egy Szabotőr olyan csövet lyukaszt ami meg lett foltozva ezért egy ideig nem lehet lyukasztani.
Teszt célja	Ellenőrizzük, hogy a szabotőr nem tudta-e kilyukasztani a csövet.

Teszt-eset neve	21. Szabotőr már megfoltozott de lyukasztható csövet lyukaszt
Rövid leírás	Egy Szabotőr olyan csövet lyukaszt ami meg lett foltozva ezért egy ideig nem lehet lyukasztani de ez az idő már lejárt.
Teszt célja	Ellenőrizzük, hogy a szabotőr sikeresen ki tudta-e lyukasztani a csövet.

Teszt-eset neve	22. Szerelő csövet lyukaszt
Rövid leírás	Egy szerelő kilyukasztja a csövet amin áll.
Teszt célja	Ellenőrizzük hogy a szerelő sikeresen kilyukasztotta-e a csövet.

Teszt-eset neve	23. Szerelő megjavít egy pumpát
Rövid leírás	A szerelő megjavít egy törött pumpát.
Teszt célja	Ellenőrizzük hogy a javítás után a víz pumpálása sikeres-e.

Teszt-eset neve	24. Szerelő megfoltoz egy csövet
Rövid leírás	Egy szerelő megfoltozza a csövet amin áll.
Teszt célja	Ellenőrizzük, hogy a foltozás után a csőből kifolyik-e a víz.

Teszt-eset neve	25. Szerelő ragadóssá tesz egy csövet
Rövid leírás	A szerelő ragadóssá teszi a csövet amin áll.
Teszt célja	Ellenőrizzük, hogy a cső ragadós lett-e.

Teszt-eset neve	26. Szabotőr csúszóssá tesz egy csövet
Rövid leírás	A szabotőr csúszóssá teszi a csövet.
Teszt célja	Ellenőrizzük, hogy a csőre nem tud játékos rálépni úgy, hogy ne kerüljön (csússzon) át egy másik csomópontra azonnal.

Teszt-eset neve	27. Vízet pumpálunk a ciszternába
Rövid leírás	A ciszternába befolyik a víz.
Teszt célja	Ellenőrizzük, hogy a ciszternához tartozó csövekből a elfolyt a víz a ciszternába.

Teszt-eset neve	28. Forrásból víz folyik
Rövid leírás	A forrásból víz kerül a hozzá kötött csövekbe.
Teszt célja	Ellenőrizzük hogy víz folyik-e a forráshoz kötött csövekbe.

Teszt-eset neve	29. Forráshoz kötött cső eldugult
Rövid leírás	A forráshoz közvetlenül kötött cső tele van, tehát a forrásból a víz nem tud kifolyni.
Teszt célja	Ellenőrizzük hogy abban az esetben, ha a forráshoz kötött cső tele van, a forrásból a víz valóban nem képes a csőbe folyni.

Teszt-eset neve	30. Játékos megváltoztatja a pumpa folyási irányát
Rövid leírás	A játékos megváltoztatja, hogy a pumpa melyik csőből melyik másikba pumpálja a vizet.
Teszt célja	

Teszt-eset neve	31. Pumpa tickelődik
Rövid leírás	A pumpa továbbítja a bemeneti csővéről érkezett vizet a kimeneti csőbe.
Teszt célja	Ellenőrizzük hogy a pumpa pumpál-e.

Teszt-eset neve	32. Csúszós cső tickelődik
Rövid leírás	Egy csúszós cső egy idő után visszaáll eredeti állapotába.
Teszt célja	Ellenőrizzük hogy egy csúszós cső sikeresen visszaáll eredeti állapotába.

Teszt-eset neve	33. Ragadós cső tickelődik
Rövid leírás	Egy ragadós cső egy idő után visszaáll eredeti állapotába.
Teszt célja	Ellenőrizzük hogy egy ragadós cső sikeresen visszaáll eredeti állapotába.

Teszt-eset neve	34. Ciszterna tickelődik
Rövid leírás	Egy ciszterna beszívja a hozzá kötött csövekben lévő vizet.
Teszt célja	Ellenőrizzük hogy a ciszterna sikeresen beszívja-e a vizet.

Teszt-eset neve	35. A szerelő lehelyez egy pumpát egy csőre
Rövid leírás	A szerelő lehelyez egy pumpát, úgy, hogy egy csövet kettévág, és a cső végeire a pumpát csatlakoztatja.
Teszt célja	Ellenőrizzük hogy a pumpára sikeresen fel lettek csatlakoztatva az elvágott cső szabad végei.

Teszt-eset neve	36. A szerelő lehelyez egy csövet
------------------------	-----------------------------------

Rövid leírás	A szerelő lehelyez egy csövet, amelynek egyik végét rákötí a pumpára amelyiken áll.
Teszt célja	Ellenőrizzük, hogy a pumpára sikeresen felcsatlakoztattuk a cső egyik végét.

Teszt-eset neve	37. Telített pumpába vizet pumpálunk
Rövid leírás	Egy olyan pumpába próbálunk vizet pumpálni, amelyik megtelt, és nem képes tovább adni a vizet.
Teszt célja	Ellenőrizzük, hogy a pumpa képes-e megtelítődni, és ilyen esetben valóban nem tud a csőből kiszívni.

Teszt-eset neve	38. Üres csőből próbál vizet kiszívni a pumpa
Rövid leírás	A pumpa bemeneti csőve üres, a pumpa ilyenkor a tartályából továbbítja a vizet.
Teszt célja	Ellenőrizzük, hogy ha a bemeneti cső üres, akkor a pumpa nem tud kiszívni vizet onnan, ilyenkor pedig a tartályából továbbítja a vizet.

Teszt-eset neve	39. Szabad végű csőből víz folyik a sivatagba
Rövid leírás	A pumpa kimeneti csővének másik végén keresztül a víz a sivatagba folyik.
Teszt célja	Ellenőrizzük, hogy ha a cső egyik vége szabad, akkor a víz valóban a sivatagba folyik.

Teszt-eset neve	40. Lyukas csőből víz folyik a sivatagba
Rövid leírás	A pumpa kimeneti csőve lyukas, tehát az ott folyó víz a sivatagba folyik.
Teszt célja	Ellenőrizzük, hogy ha a cső lyukas, akkor a rajta keresztül folyó víz a sivatagba jut.

7.4 Tesztelést támogató segéd- és fordítóprogramok specifikálása

A tesztelést egy parancssori segédprogram fogja végezni. A felhasználó megadja a teszt sorszámát vagy egy sorszámokból álló intervallumot, amely alapján a program kiválasztja a futtatandó teszteket, illetve ha nem ad sorszámot, akkor az összes tesztet lefuttatja. Az elvárt és valós kimenetek összehasonlítása érdekében a tesztelő programunk beolvassa a teszthez tartozó bemeneti adatokat egy fájlból, majd az elvárt kimeneti értékeket egy másik fájlból. Ezután a tesztelő program a konzolra írja a teszt futtatásának eredményét, vagyis hogy a teszt sikeres volt-e vagy sem. Ha a teszt sikertelen, akkor a tesztelő program az eltérő valós és elvárt kimenet közti különbséget is kiírja a konzolra, hogy a felhasználó könnyen értelmezhesse a hibát.

7.5 Napló

Kezdet	Időtartam	Részvevők	Leírás
2023.04.18. 17:00	4 óra	Barabási Ganzer Kurcsi Nagy Palásti	<p>Értekezlet: Újrabeszéltük a feladatot, az új követelmények figyelembevételével.</p> <p>Döntés: Meghatároztuk az alapvető teszteseteket és elosztottuk, hogy ki mit fog megvalósítani: A tesztelési tervet együtt végezzük el. Emelett Barabási elkészíti az új osztálydiagramot, a 7.0.3.1. -es szekvenciadiagramot és specifikálja a tesztelés folyamatát; Ganzer elkészíti a 7.0.3.2., 7.0.3.3. szekvenciadiagramokat, leírja a megváltozott és új metódusok szerepeit; Kurcsi elkészíti a bemeneti nyelvet, valamint a 7.0.3.4. - es szekvenciadiagramot; Nagy elkészíti a kimeneti nyelvet és a 7.0.3.5. - ös szekvenciadiagramot; Palásti elkészíti a Use casek leírását, valamint a 7.0.3.6. - ös szekvenciadiagramot.</p>
2023.04.20. 16:20	2.5 óra	Barabási	Tevékenység: Az új osztálydiagram, és a 7.0.3.1. szekvenciadiagram elkészítése
2023.04.20. 19:45	3 óra	Nagy	Tevékenység: A kimeneti nyelv elkészítése.

2023.04.20. 09:50	2.5 óra	Kurcsi	Tevékenység: A bemeneti nyelv elkészítése.
2023.04.21. 11:40	2 óra	Ganzer	Tevékenység: A tesztesetek írásának elkezdése.
2023.04.21. 14:10	3 óra	Palásti	Tevékenység: Use-casek leírása.
2023.04.21. 13:15	1.5 óra	Nagy	Tevékenység: Kimeneti nyelv folytatása, valamint a 7.0.3.4. - es szekvenciadiagram elkészítése
2023.04.21. 10:15	2 óra	Barabási	Tevékenység: A tesztesetek folytatása, valamint a tesztelés folyamatát specifikálta.
2023.04.22. 10:20	2 óra	Kurcsi	Tevékenység: A bemeneti nyelv véglegesítése és a 7.0.3.4. - es szekvenciadiagram elkészítése.
2023.04.22. 11:20	1.5 óra	Palásti	Tevékenység: A 7.0.3.6. - os szekvenciadiagram elkészítése, és a use casek véglegesítése.
2023.04.23. 13:00	2.5 óra	Ganzer	Tevékenység: A 7.0.3.2., 7.0.3.3. szekvenciadiagramok elkészítése, tesztesetek folytatása, megváltozott metódusok leírása.
2023.04.23 18:00	2 óra	Barabási Ganzer Kurcsi Nagy Palásti	Értekezlet: Összesítettük a munkákat, megkerestük az esetleges hibákat. Döntés: A talált helyesírási hibákat Barabási javítja.
2023.04.23. 21:55	15 perc	Barabási	Tevékenység: Dokumentum véglegesítése.