

## 4. Analízis modell kidolgozása

### 12 – Külföldiek és András

Konzulens:  
Goldschmidt Balázs

#### Csapattagok

**Kurcsi Norbert**

Barabási Zsolt Botond

Ganzer Attila

Nagy Szabolcs

Palásti András

**Y3ZTEI**

FWHHHB

Z44FF6

Z73X7L

IDNGIS

**kurcsi.n@gmail.com**

barabasizalan@gmail.com

ganzer.attila@gmail.com

n.szaby4@gmail.com

andraspalasti29@gmail.com

2023. 03. 25.

## 4. Analízis modell kidolgozása

### 4.1 Objektum katalógus

#### 4.1.1 Saboteur

Ez egy játékos által felvehető szerep. Ezzel a szereppel az a dolga a játékosnak, hogy különböző műveleteket végrehajtva (pumpa átállítása, csövek kilyukasztása) elérje, hogy minél több víz kifolyjon a homokba. A szabotőr a pumpákon és csöveken léphet, azzal a megkötéssel, hogy ha valaki áll már egy csövön, legyen az szabotőr vagy szerelő, akkor nem léphet arra a mezőre.

#### 4.1.2 Mechanic

Ez egy játékos által felvehető szerep. Ezzel a szereppel az a dolga a játékosnak, hogy a játék során minél több vizet juttasson a forrásból a ciszternákba. Ezt a következő műveletekkel valósíthatja meg: pumpák átállítása, pumpák megjavítása, csövek megfoltozása, új pumpák elhelyezése egy-egy csővezeték közepén, illetve csövek végének az elmozdítása. A szerelő a pumpákon és csöveken léphet, azzal a megkötéssel, hogy egy csőre csak akkor léphet rá, ha más játékos nem áll éppen rajta.

#### 4.1.3 Pipe

Az a pályaelem amelyen keresztül a játékosok a vizet szállíthatják a pumpák között, és ezáltal a forrásból a ciszternákba. A szabotőrök kilyukaszthatják, a szerelők pedig megjavíthatják, valamint áthelyezhetik. Egy cső mindig két játékelemet (ciszterna, forrás, pompa) köt össze. A csőben csak akkor folyhat víz, ha az egyik vége egy pumpához van csatlakoztatva. Egy csőben mindig csak egy irányba folyhat a víz, tehát van egy bemeneti illetve egy kimeneti pumpája. Emellett minden cső rendelkezik egy kapacitással, ami korlátozza annak a víznek a mennyiségét, amit átereszt. A ciszternáknál keletkező szabad végű csöveket a szerelők felvehetik, majd egy kiválasztott helyen letehetik, amiután letették, már nem vehető fel a cső, csak a cső egyik vége köthető át.

#### 4.1.4 Pump

Az a pályaelem, amelyen keresztül a víz pumpálódik egyik csőből a másikba. A szerelők, valamint a szabotőrök egyaránt állíthatják, hogy honnan hová pumpáljon. Egyszerre több játékos is állhat rajta. A pompa rendelkezik egy tartállyal, amelybe akkor kerül víz ha a csövön érkező víz kapacitása meghaladja a kimeneti cső kapacitását. A pumpák véletlen időközönként elromolhatnak, amely esetben nem képesek vizet továbbítani.

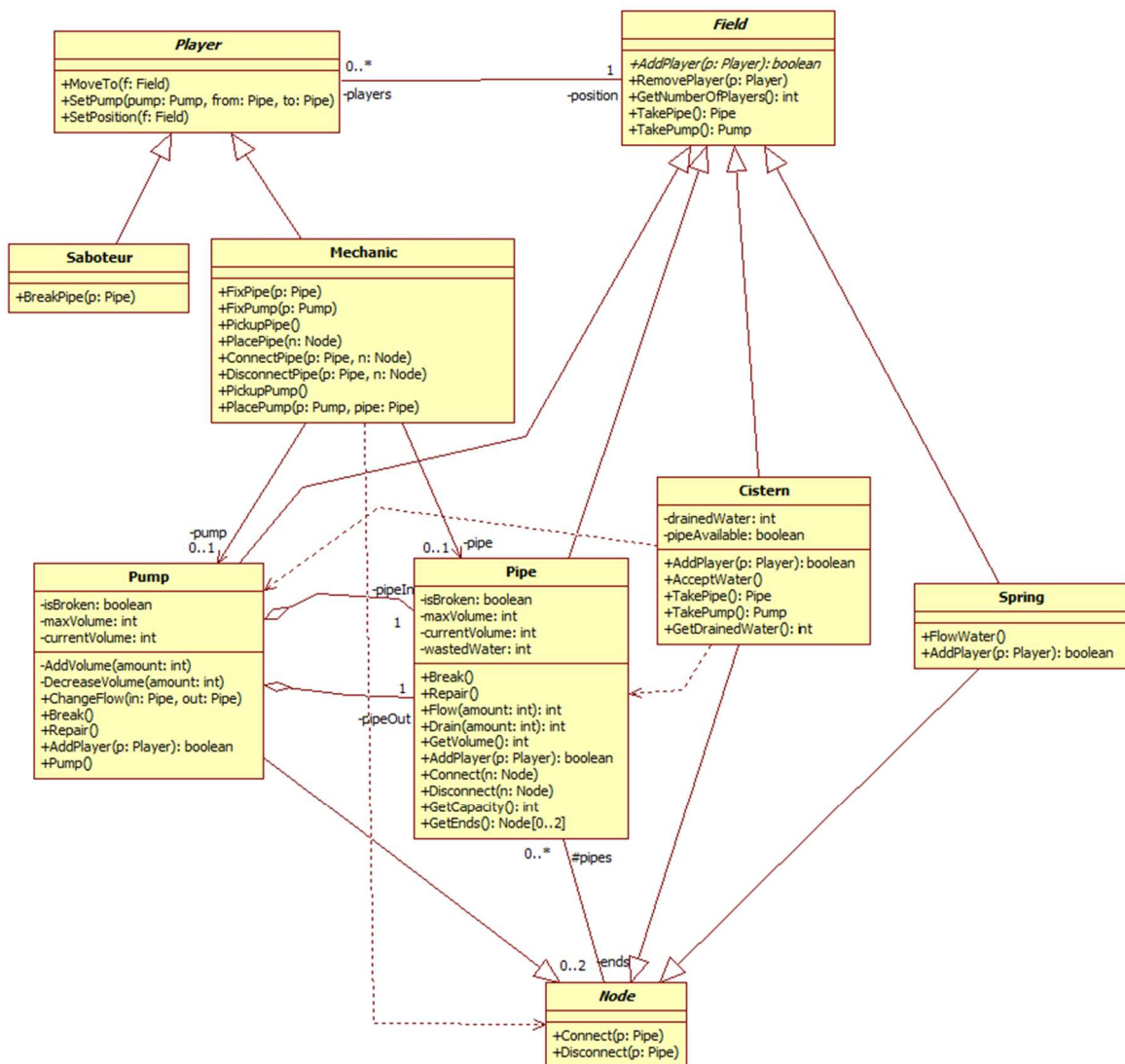
#### 4.1.5 Cistern

Az a játékelem, ahová a szerelők a vizet eljuttatva pontokat szerezhetnek. A ciszternát a játékosok nem képesek elmozdítani a játék során. A játék során folyamatosan termelődnek ezen a helyen a szabad csövek, valamint pumpák, amelyeket a szerelők felhasználhatnak a pálya építéséhez.

#### 4.1.6 Spring

Az a játékelem, ahonnan a pályán lévő víz származik, amelyet a játékosok a szerepüknek megfelelően adott helyre próbálnak eljuttatni. A szabotőrök, és a szerelők sem képesek a pályaelem elmozdítására, annak bármilyen megváltoztatása nem lehetséges.

## 4.2 Statikus struktúra diagram



## 4.3 Osztályok leírása

### 4.3.1 Cistern

- Felelősség**

A ciszternákért felelős osztály. A szerelők célja, hogy minél több vizet juttassanak a forrásoktól a ciszternákig.

- Össztályok**

Field, Node.

- Interfészek**

- Attribútumok**

- drainedWater:** A vízmennyiség ami befolyt a ciszternákba. Ennek függvényében számolódnak a szerelő csapat pontjai.

- pipeAvailable:** Van-e elérhető cső, amelyet a szerelők felvehetnek az eszköztárukba.

- **Metódusok**
- **boolean AddPlayer(Player p):** Ha egy játékos rálép a pályaelemre, ezt hozzáadja a játékosok listájához. True-t térít vissza, ha sikerült játékost hozzáadni, False-t ha nem.
- **void AcceptWater():** Befogadja a vizet.
- **Pipe TakePipe():** Ha még ebben a körben nem vettek el csövet róla akkor visszatér egy pumpával különben pedig null-al.
- **Pump TakePump():** Mivel bármikor felvehető pumpa a ciszternából ezért mindig egy pumpával tér vissza.
- **int GetDrainedWater():** A ciszternába belefolyt víz mennyiségét téríti vissza.

#### 4.3.2 Field

- **Felelősség**  
A pályaelemekért felelős absztrakt osztály.
- **Össztályok**
- **Interfészek**
- **Attribútumok**
- **players:** A pályaelemen álló játékosok listája.
- **Metódusok**
- **boolean AddPlayer(Player p):** Ha egy játékos rálép a pályaelemre, ezt hozzáadja a játékosok listájához. True-t térít vissza, ha sikerült játékost hozzáadni, False-t ha nem.
- **void RemovePlayer():** Ha egy játékos ellép a pályaelemről, ezt kitörli a játékosok listájából.
- **int GetNumberOfPlayers():** Visszatéríti a pályaelemen álló játékosok számát. A cső esetében, egyszerre több mint egy játékos nem állhat rajta.
- **Pipe TakePipe():** Megpróbál elvenni egy csövet az adott mezőről, ha nem megy akkor null értékkel tér vissza.
- **Pump TakePump():** Megpróbál elvenni egy pumpát az adott mezőről, ha nem megy akkor null értékkel tér vissza.

#### 4.3.3 4.3.3 Mechanic

- **Felelősség**  
A lyukas csövek és az elromlott pumpák megjavításáért felelős, valamint a csőrendszert úgy kell alakítsa, hogy minél több víz jusson a forrástól a ciszternákba.
- **Össztályok**  
Player.
- **Interfészek**
- **Attribútumok**
- **pump:** Az eszköztárban a játékosnak lehet egy vagy nulla pumpája, amelyet majd lehelyezhet.
- **pipe:** Az eszköztárban a játékosnak lehet egy vagy nulla csöve, amelyet majd lehelyezhet.

- **Metódusok**
- **void FixPipe(Pipe p):** A szerelő megjavít egy kilyukasztott csövet. Ezután a csövön keresztül, a víz átjut egyik pumpától a másikig, anélkül, hogy elfolyjon a homokba.
- **void FixPump(Pump p):** A szerelő megjavít egy meghibásodott pumpát. A pumpa ezután képes lesz a csöveken tovább pumpálni a vizet más csövek felé.
- **void PickupPipe():** A szerelő felvesz egy csövet a ciszternáktól. Ezután az eszköztárába elérhető lesz egy cső, amelyet ezt lehelyezhet egy pumpánál.
- **void PlacePipe(Node n):** A szerelő letesz egy csövet, amelyet a ciszternáktól vett fel. Egyik végét arra a pumpára köti amelyiken áll, a másik vége szabadon lóg. Ezután az eszköztára kiürül, és ha a ciszternáknál jár az elkövetkezendő körökben, akkor újabb csövet vehet fel.
- **void PickupPump():** A szerelő felvesz egy pumpát ha van a mezőn. Ez a pumpa a szerelő eszköztárába kerül, majd ezt a pályán bárhol lehelyezheti, ahol ez megengedett.
- **void PlacePump(Pump p, Pipe pipe):** A szerelő letesz egy pumpát, amely az eszköztárában volt eddig. A *pipera* letett pumpa a csövet kettéválasztja, egy új cső létrejön és a megfelelő végeit a régi és új csőnek csatlakoztatja a pumpához. Ezután az eszköztára kiürül és felvehet egy más pumpát az elkövetkezendő körökben.
- **void ConnectPipe(Pipe p, Node n):** A paraméterként kapott csomópontra csatlakoztatjuk fel a paraméterként kapott csövet.
- **void DisconnectPipe(Pipe p, Node n):** A paraméterként kapott csomópontra csatlakoztatott *p* csövet lecsatlakoztatja.

#### 4.3.4 4.3.4 Node

- **Felelősség**  
Egy absztrakt osztály ami egy csomópontot jelöl mindenhez amihez csöveket lehet csatlakoztatni.

- **Ősosztályok**  
Nincsenek.

- **Interfészek**
- **Attribútumok**
- **pipes:** a csatlakoztatható csövek halmaza
- **Metódusok**
- **void Connect(p: Pipe):** A csomóponthoz csatlakoztatja az adott csövet.
- **void Disconnect(p: Pipe):** Lecsatlakoztatja a csomóponttól az adott csövet.

#### 4.3.5 4.3.5 Pipe

- **Felelősség**  
A víz szállításáért felelős, vagy a többi játékelemből szállít el vizet vagy azokba szállítja. valamint ezen képesek a játékosok mozogni.
- **Ősosztályok**  
Field, Node.

- **Interfészek**
- **Attribútumok**
  - **maxVolume:** a cső kapacitása hogy maximum mennyi víz fér bele
  - **currentVolume:** a csővön érkező aktuális vízmennyiség
  - **isBroken:** a cső állapotát jelzi, midőn ki van lyukasztva vagy sem
  - **wastedWater:** a lyukakon elfolyt vízmennyiség
- **Metódusok**
  - **void Brake():** A cső kilyukadt állapotba kerül. Ezentúl a csővön érkező víz mind elfolyik, ezzel növelve a szabotőrök pontszámát.
  - **void Repair():** A cső megjavítása, amiután képes lesz vizet szállítani a pumpák között, valamint a ciszternák irányába.
  - **int Flow(int amount):** Egy pumpa vagy forrás hívja meg a csővön amikor betölti a vizet az adott csőbe. Visszatéríti azt a mennyiséget, ami befolyt a csőbe.
  - **int Drain(int amount):** Egy pumpa vagy ciszterna hívja meg a csővön amikor kiszívja annak tartalmát. Visszatéríti azt a mennyiséget, ami kifolyt a csőből.
  - **boolean AddPlayer(Player p):** Ha egy játékos rálép a pályaelemre, ezt hozzáadja a játékosok listájához. True-t térít vissza, ha sikerült játékost hozzáadni, False-t ha nem.
  - **int GetVolume():** Visszatéríti a csőben lévő víz mennyiségét.
  - **int GetCapacity():** Visszatéríti a cső kapacitását.
  - **void Connect(Node n):** A cső szabad végét rácsatlakoztatja az  $n$  csomóponttra.
  - **void Disconnect(Node n):** Lecsatlakoztatja a cső azon végét, ami az  $n$  csomóponthoz van csatlakoztatva.
  - **Node[] GetEnds():** A cső végeit, azaz a végeire csatlakoztatott játék elemeket téríti vissza.

### 4.3.6 4.3.6 Player

- **Felelősség**  
A játékosokért felelős és azokat megvalósító absztrakt osztály, amiből két további osztály származik: Mechanic, Saboteur.
- **Ősosztályok**  
Nincsenek.
- **Interfészek**
- **Attribútumok**
  - **position:** a játékos pozíciójának tárolására használt attribútum.
- **Metódusok**
  - **void MoveTo(Field f):** A játékos pozíciót vált.
  - **void SetPump(Pump pump, Pipe from, Pipe to):** A játékos a paraméterként kapott pumpát átállítja, hogy melyik csőből melyikbe pumpálja a vizet.
  - **void SetPosition(Field f):** Beállítja a játékos aktuális pozícióját.

### 4.3.7 4.3.7 Pump

- **Felelősség**

A víz pumpálásáért felelős osztály.

- **Ősosztályok**

Field, Node.

- **Interfészek**

- **Attribútumok**

- **maxVolume:** A pumpa tartályába tölthető maximális vízmennyiség.

- **currentVolume:** A pumpa tartályában lévő vízmennyiség.

- **isBroken:** A pumpa állapotát jelzi, ha működőképes vagy meghibásodott

- **pipeIn:** Az a cső, amelyről érkezik a víz.

- **pipeOut:** Az a cső, amelyiken keresztül továbbítódik a víz.

- **Metódusok**

- **void ChangeFlow(Pipe in, Pipe out):** Megváltoztatja, hogy melyik csőből melyikbe pumpálja a vizet.

- **void Break():** A pumpa meghibásodik, ezután ez használhatatlanná válik, nem képes vizet pumpálni, addig amíg egy szerelő meg nem javítja.

- **void Repair():** A pumpa megjavítása, ezután már használható - újra pumpálhat vizet.

- **void Pump():** Egyik csőből a másikba pumpálja a vizet.

- **boolean AddPlayer(Player p):** Ha egy játékos rálép a pályaelemre, ezt hozzáadja a játékosok listájához. True-t térít vissza, ha sikerült játékost hozzáadni, False-t ha nem.

### 4.3.8 4.3.8 Saboteur

- **Felelősség**

A szabotőrök a csövek kilyukasztásáért, valamint a csövek átállításáért felelősek, ezzel minél több vizet a homokba folytatnak, hogy ők legyenek a győztesek.

- **Ősosztályok**

Player.

- **Interfészek**

- **Attribútumok**

- **Metódusok**

- **void BreakPipe(Pipe p):** A szabotőr egy paraméterként kapott csövet kilyukaszt. Ezután a kilyukasztott csőből a víz a homokba folyik, ezzel növelve a csapat pontszámát.

### 4.3.9 4.3.9 Spring

- **Felelősség**

A rendszert, pontosabban a belőle kivezető csöveket látja el vízzel. Ezekbe minden kör végén folytatja a megfelelő mennyiségű vizet.

- **Ősosztályok**

Field, Node.

- **Interfészek**

- **Attribútumok**

- **Metódusok**

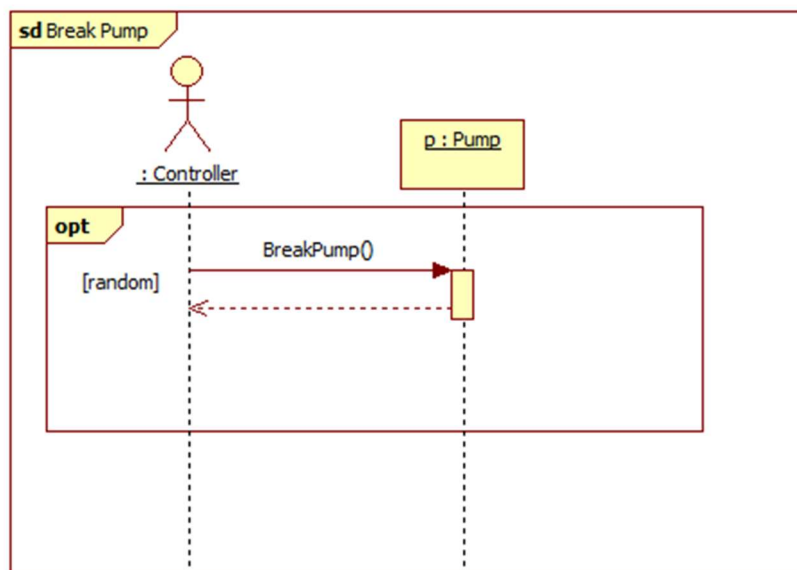
- **void FlowWater():** A víz áramlik a forrásokból.

- **boolean AddPlayer(Player p):** Ha egy játékos rálép a pályaelemre, ezt hozzáadja a játékosok listájához. True-t térít vissza, ha sikerült játékost hozzáadni, False-t ha nem.

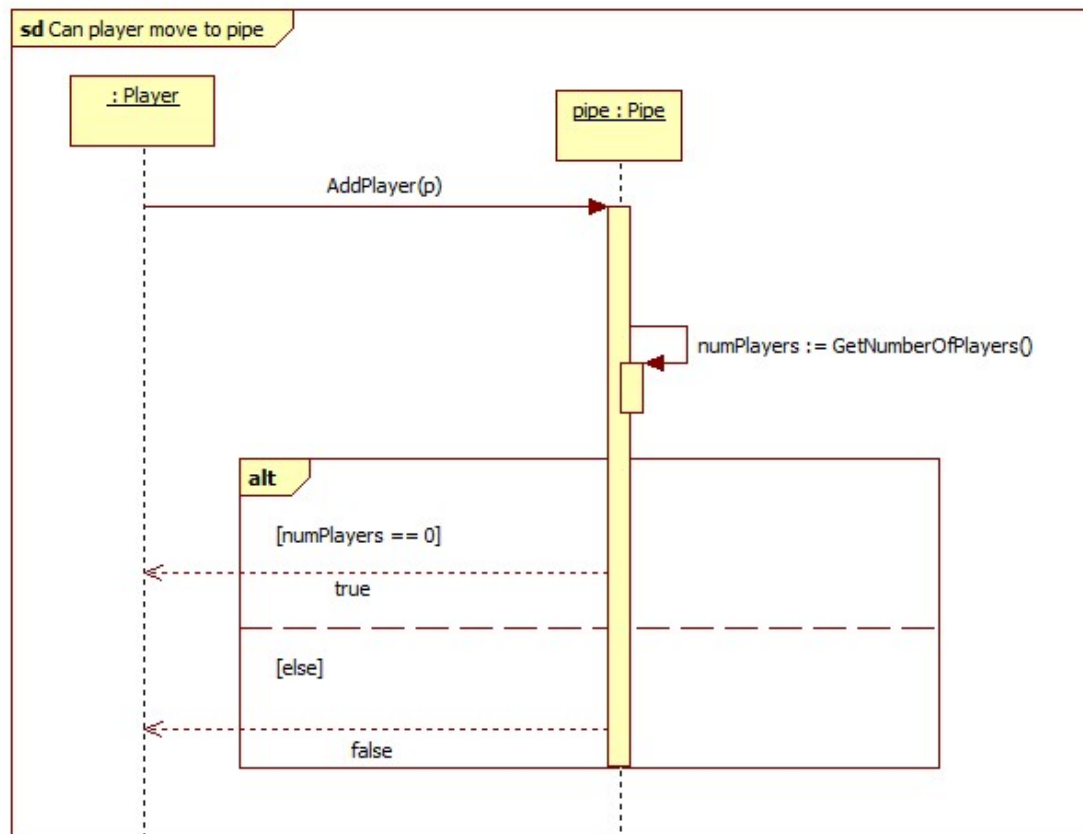


## 4.4 Szekvencia diagramok

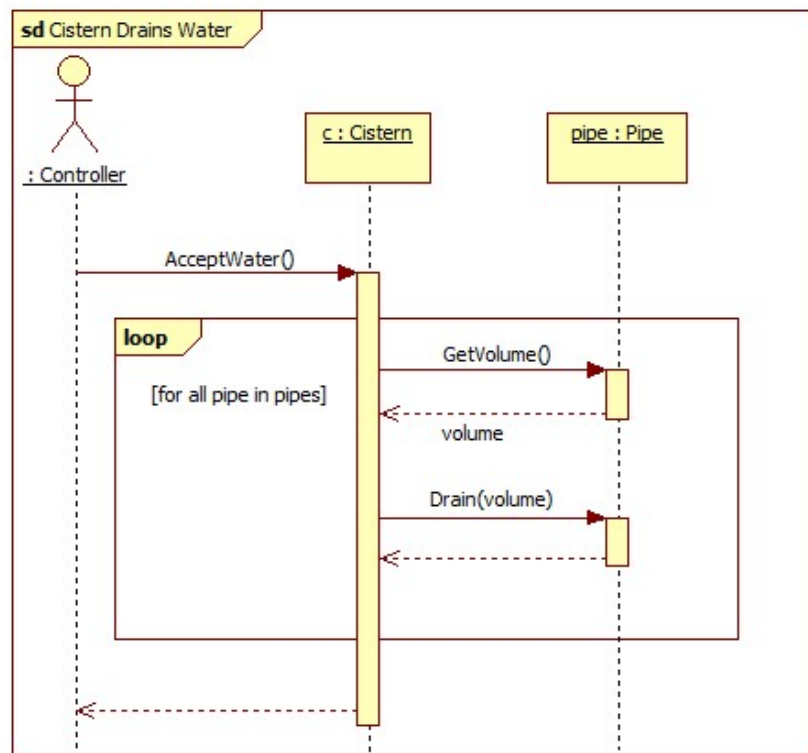
### 4.4.1 Break Pump



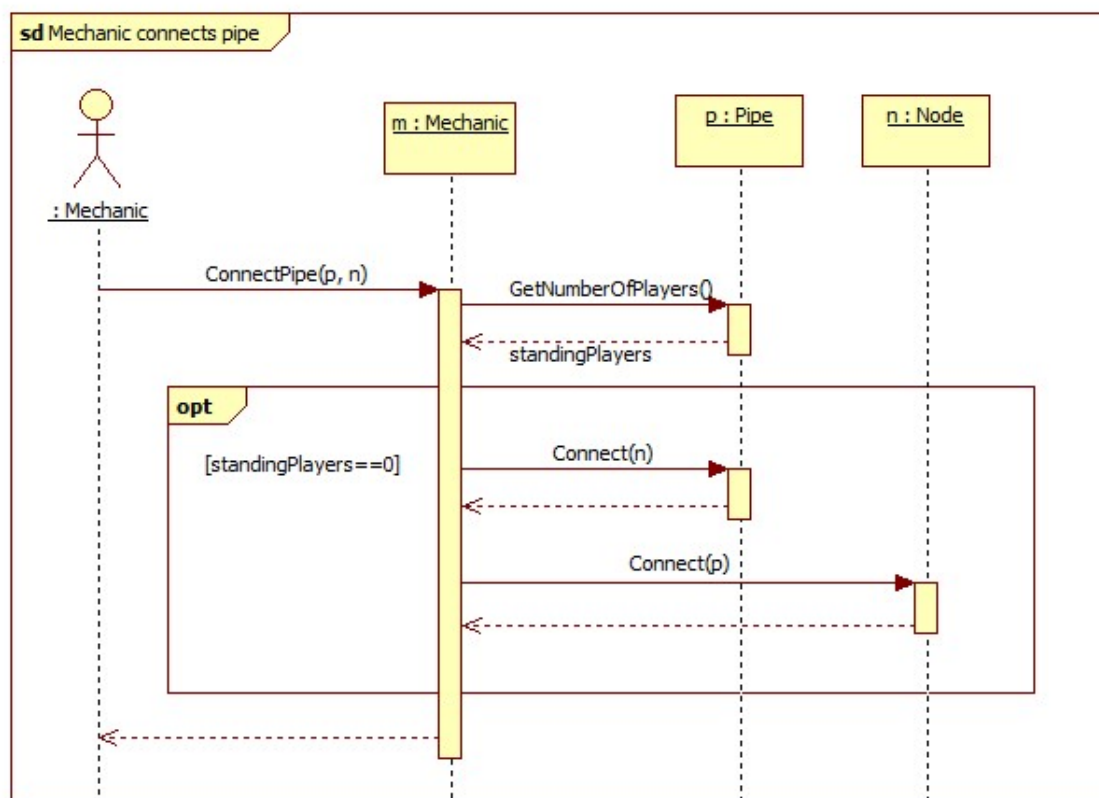
### 4.4.2 Can Player Move To Pipe



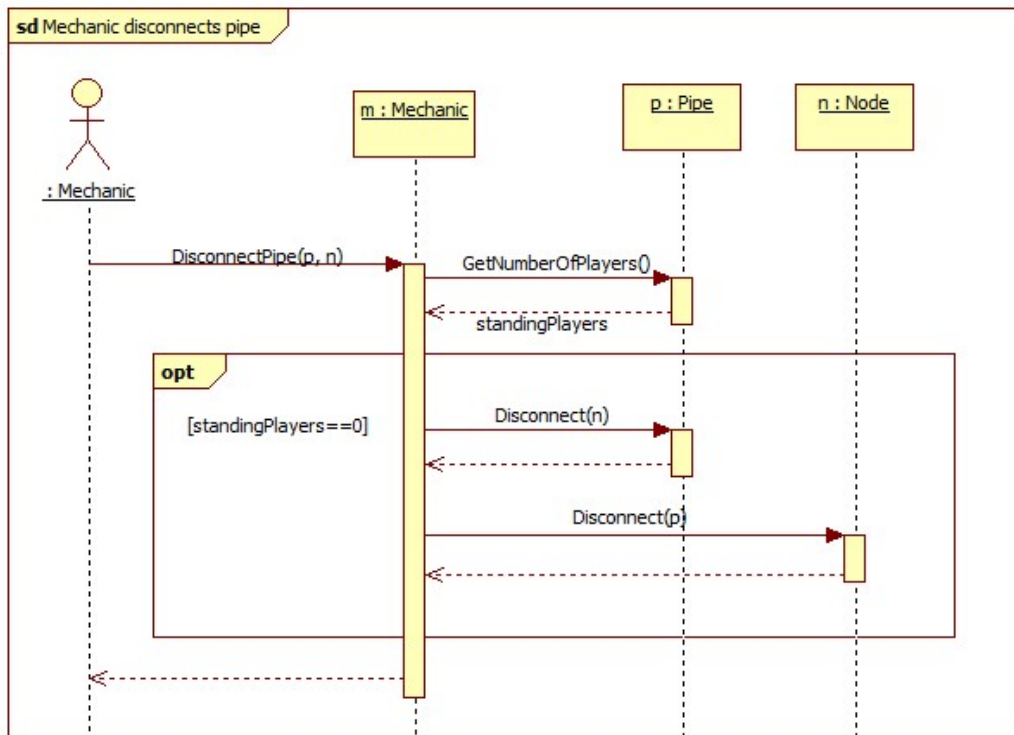
### 4.4.3 Cistern Drains Water



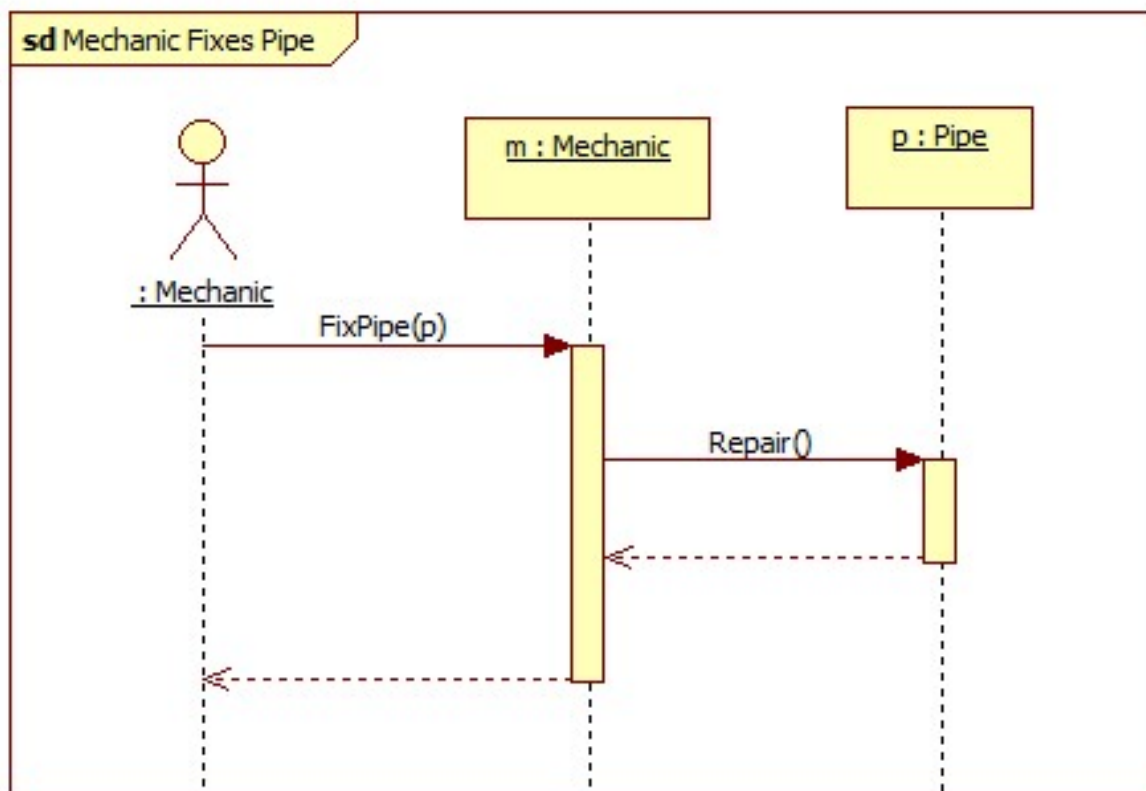
### 4.4.4 Mechanic Connects Pipe



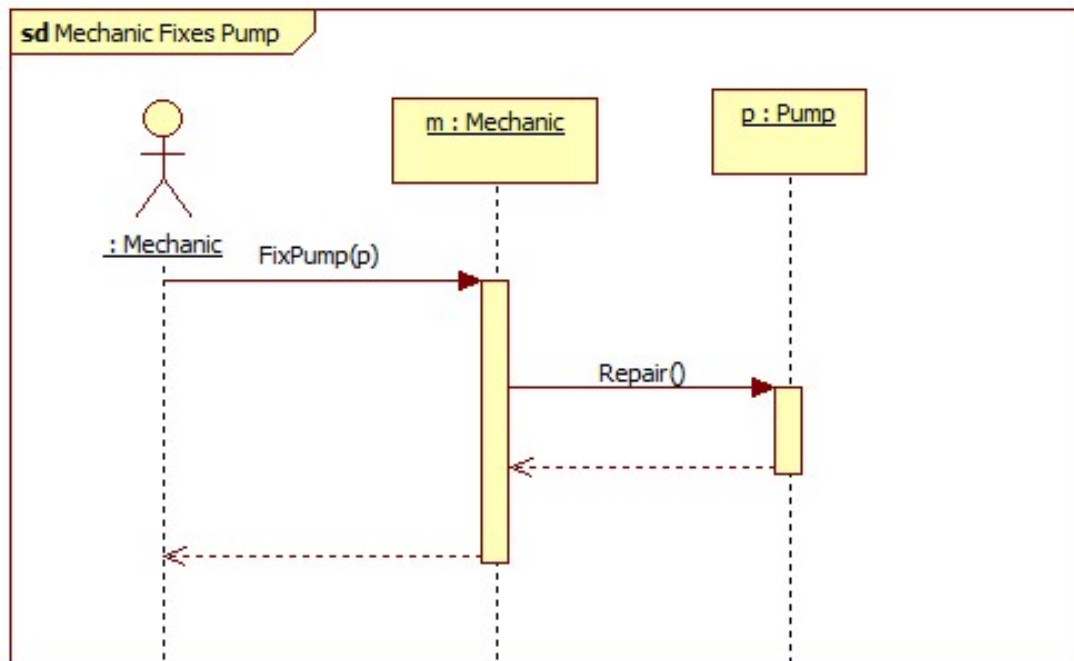
#### 4.4.5 Mechanic Disconnects Pipe



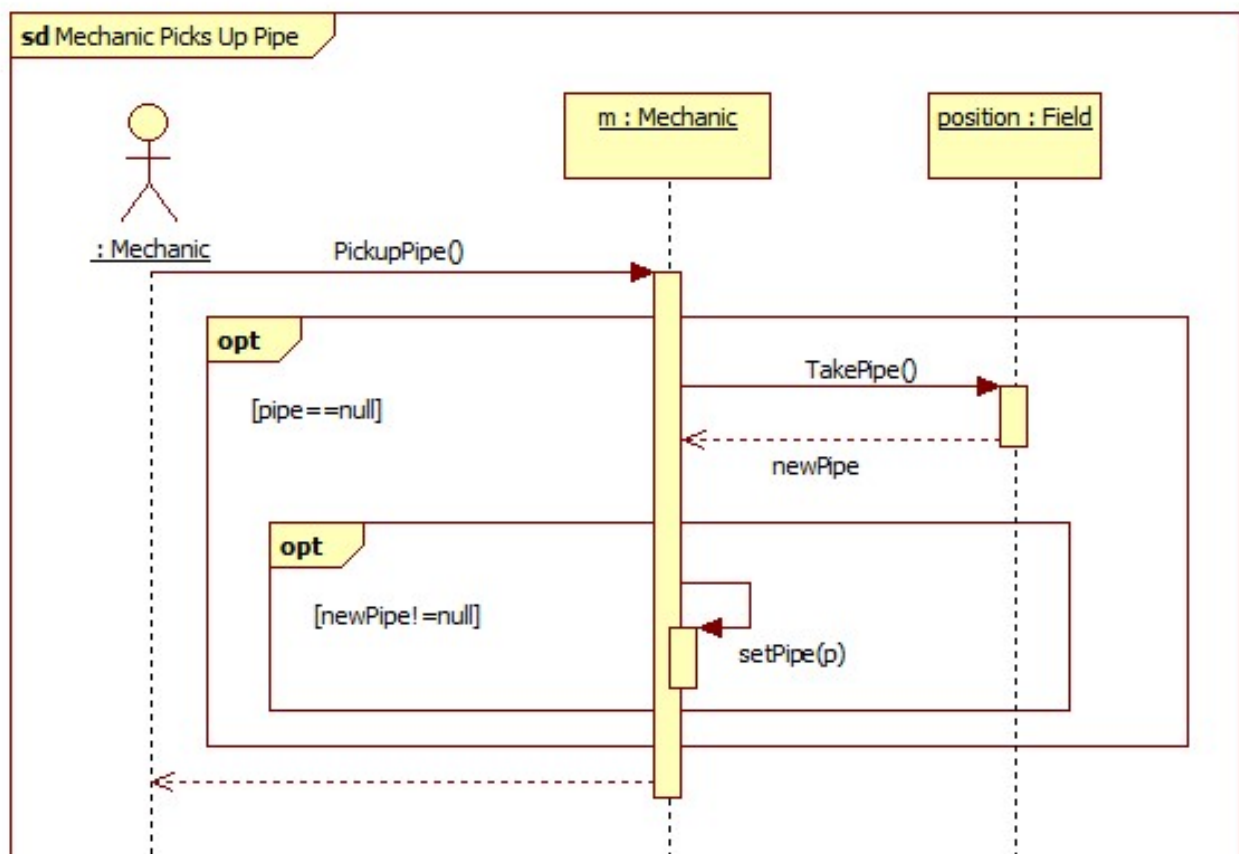
#### 4.4.6 Mechanic Fixes Pipe



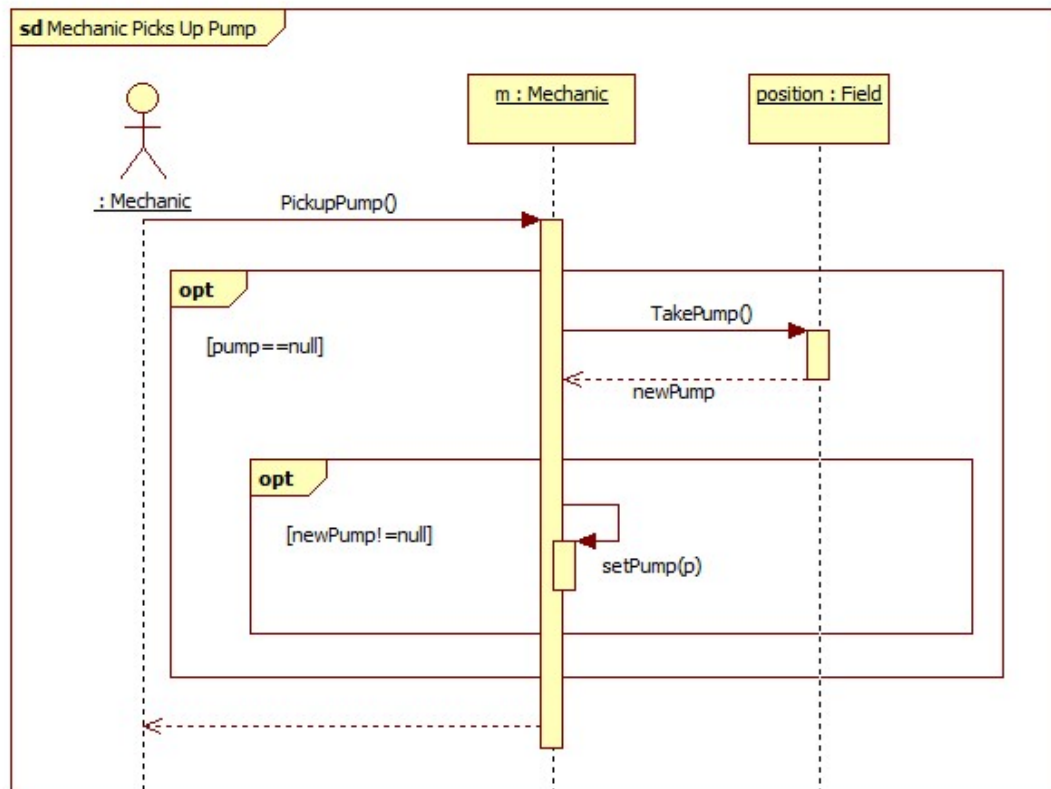
#### 4.4.7 Mechanic Fixes Pump



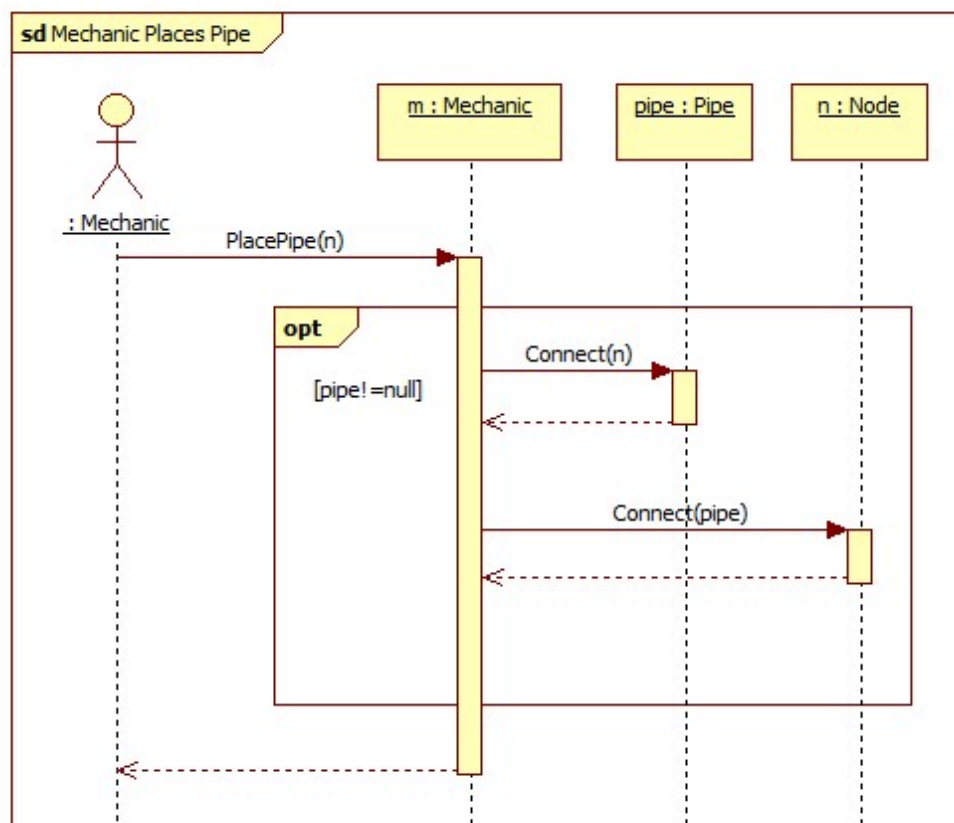
#### 4.4.8 Mechanic Picks Up Pipe



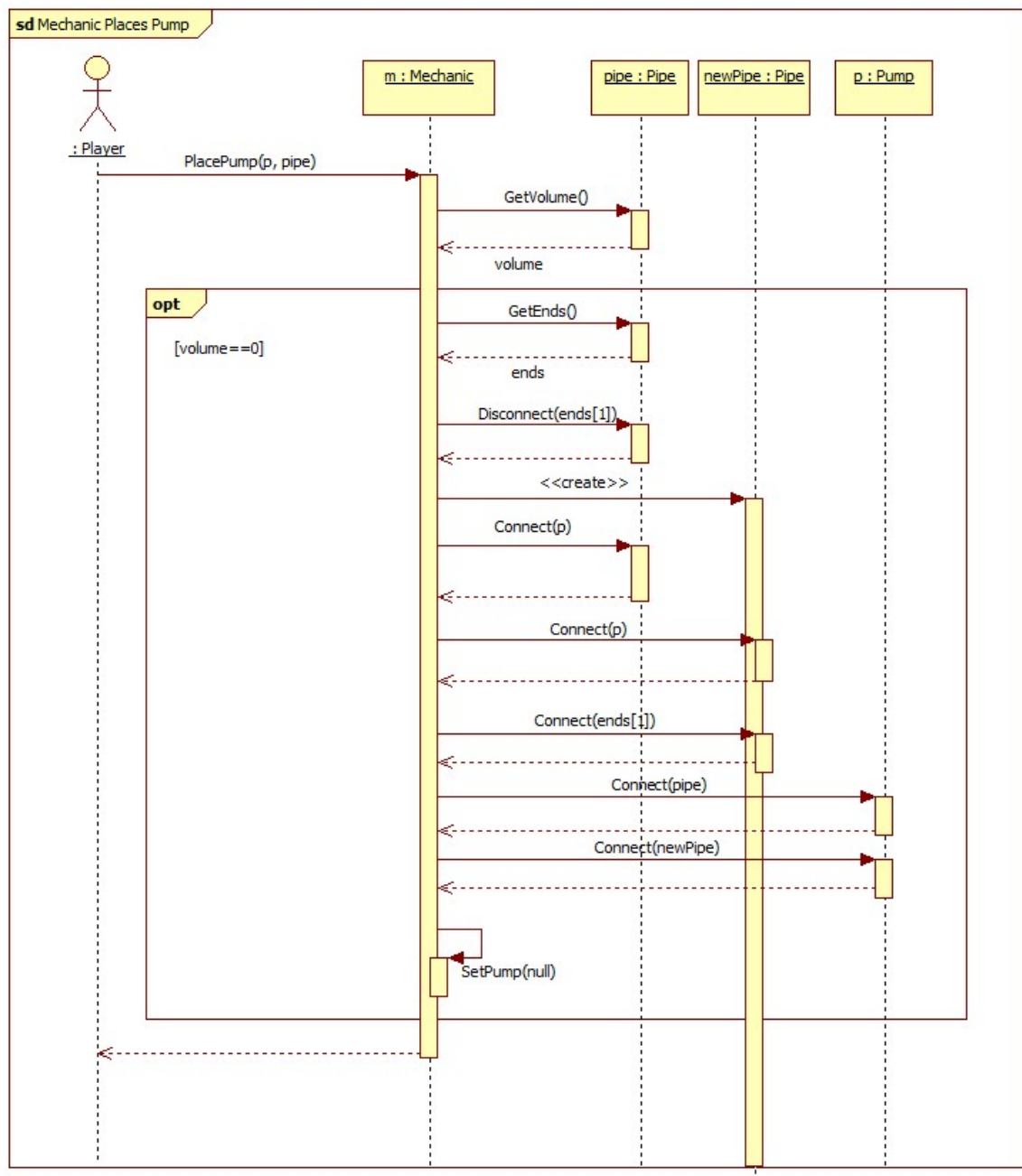
## 4.4.9 Mechanic Picks Up Pump



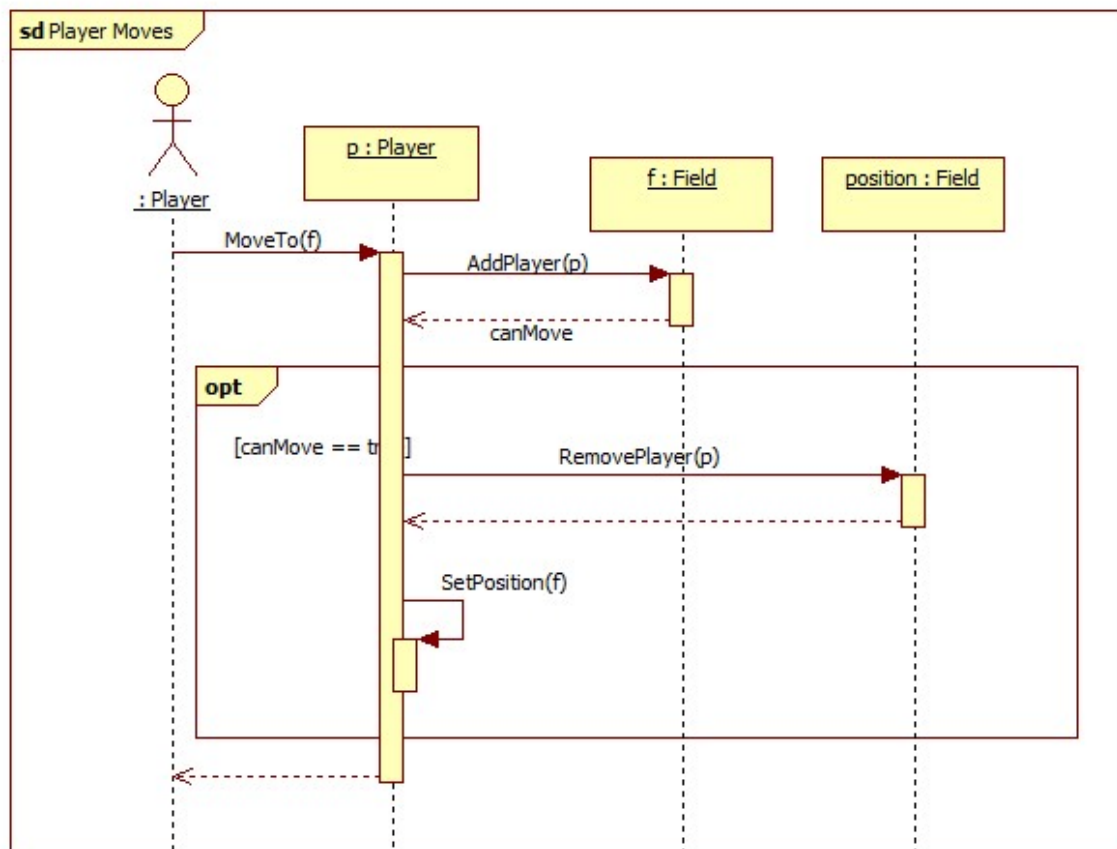
## 4.4.10 Mechanic Places Pipe



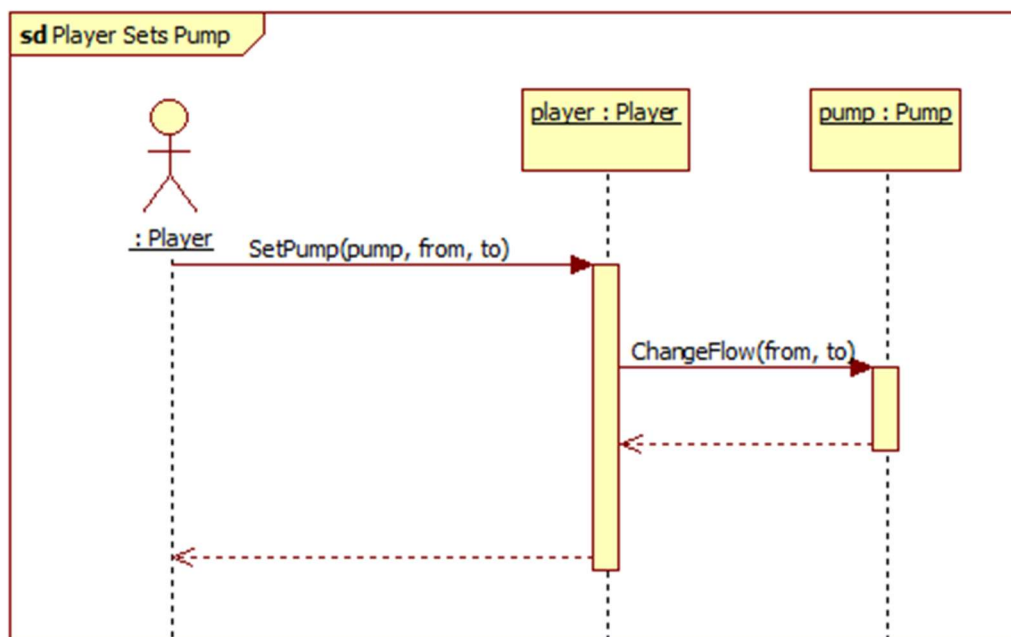
## 4.4.11 Mechanic Places Pump



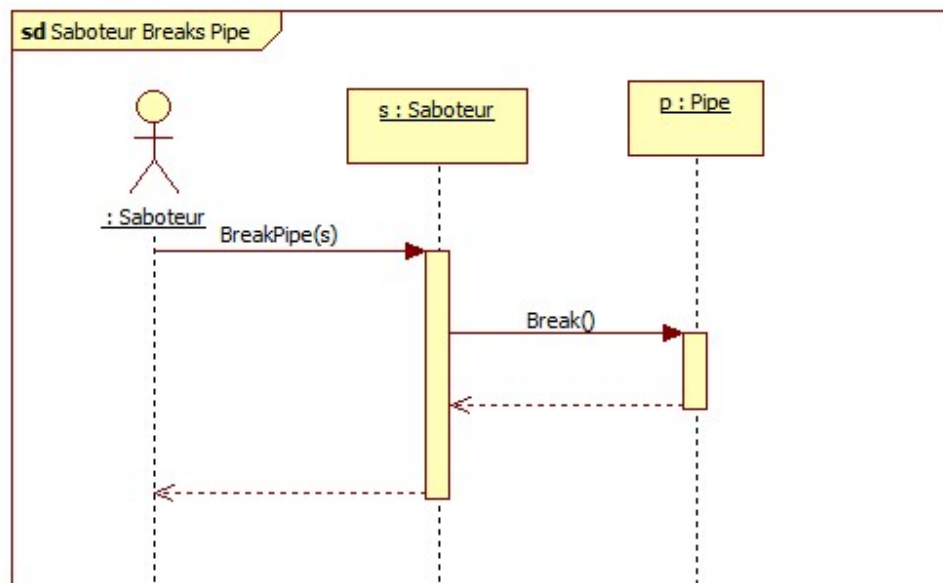
## 4.4.12 Player Moves



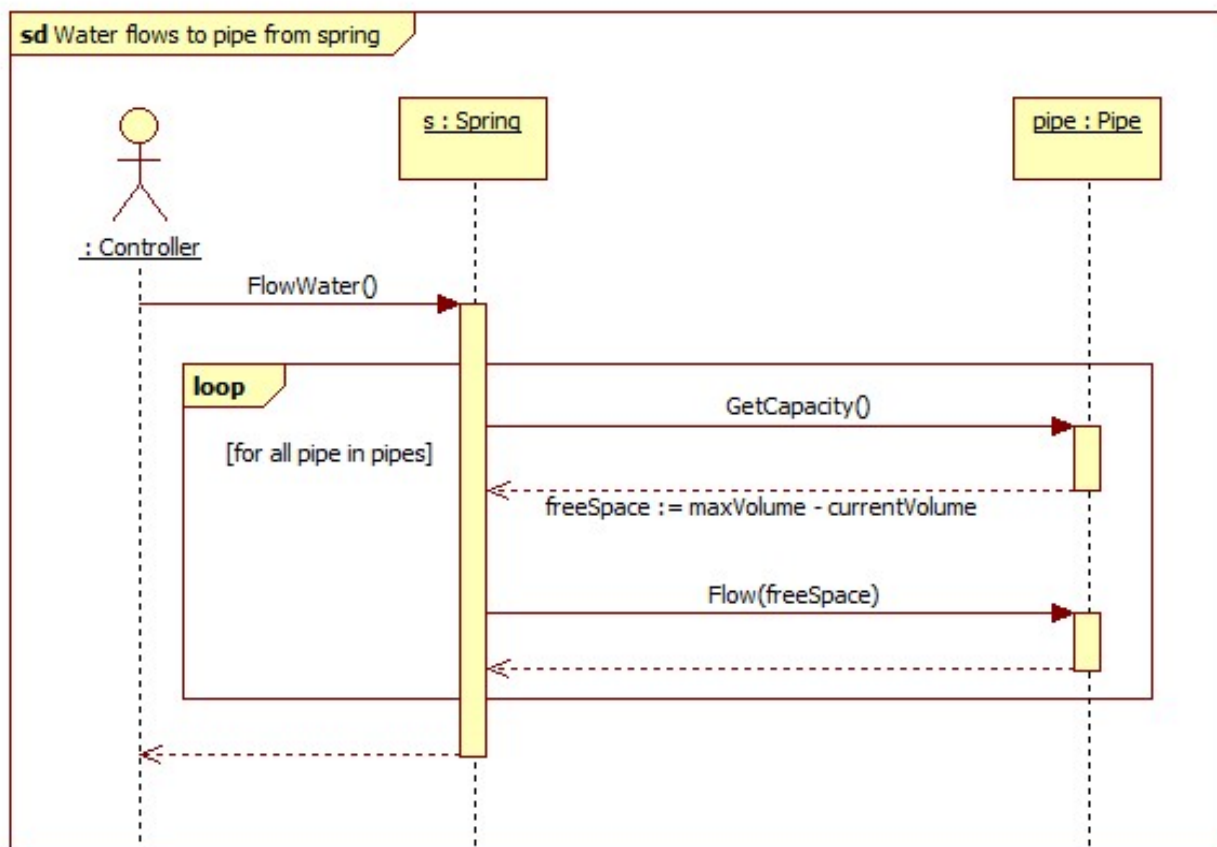
## 4.4.13 Player Sets Pump



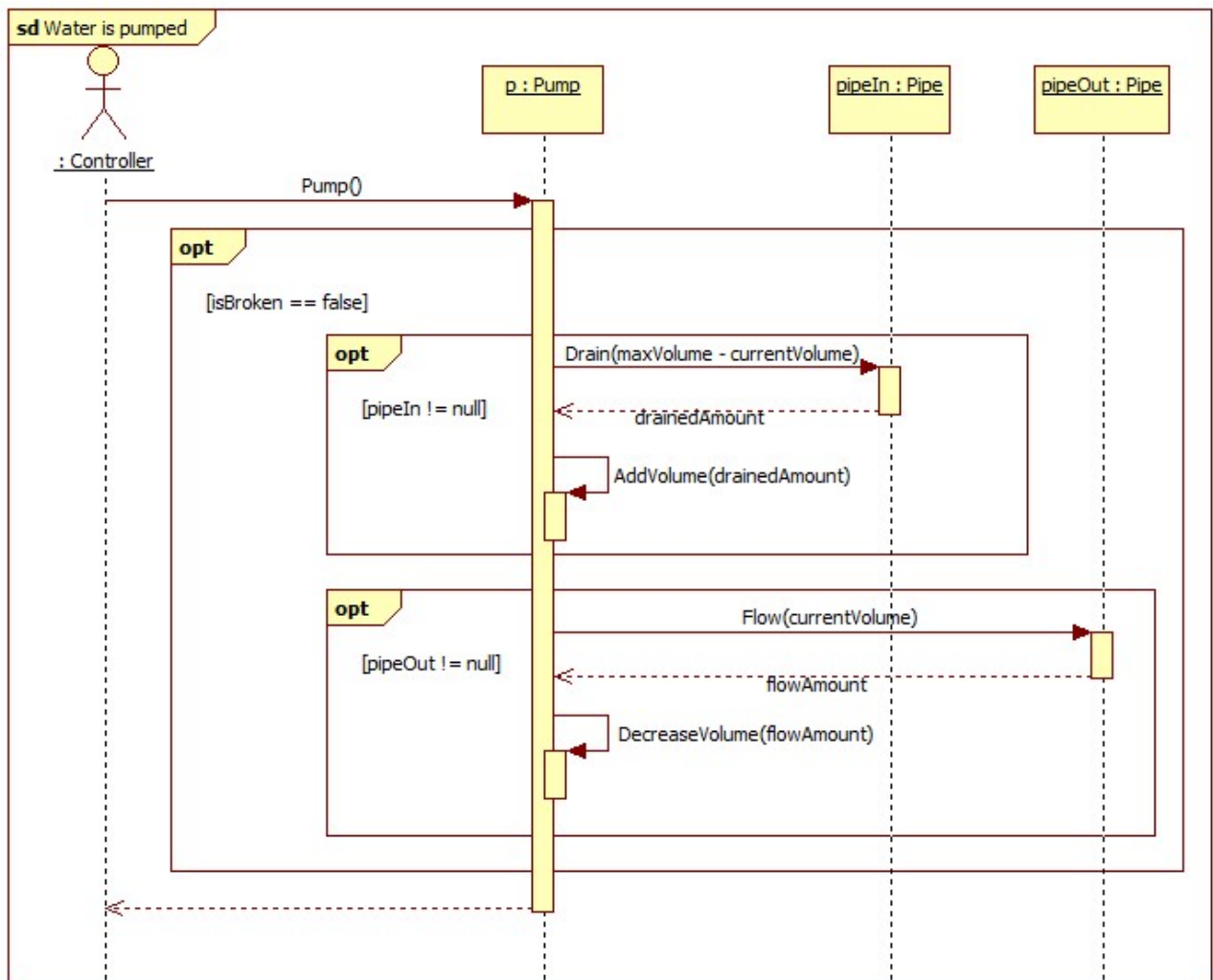
#### 4.4.14 Saboteur Breaks Pipe



#### 4.4.15 Water Flows To Pipe From Spring

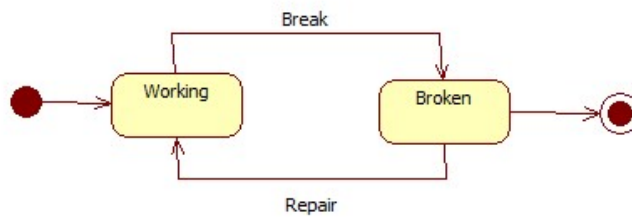




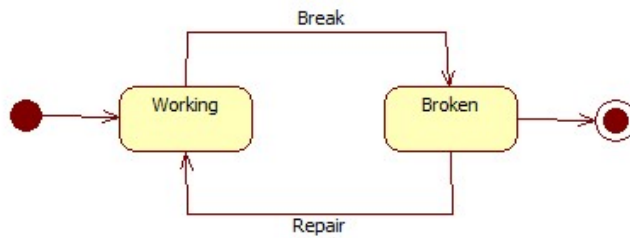
**4.4.16 Water Is Pumped**

## 4.5 State-chartok

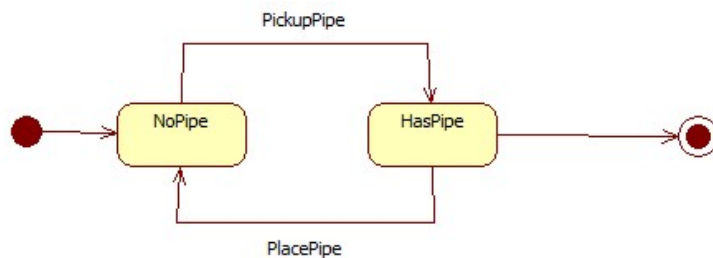
### 4.5.1 Is Pipe Broken



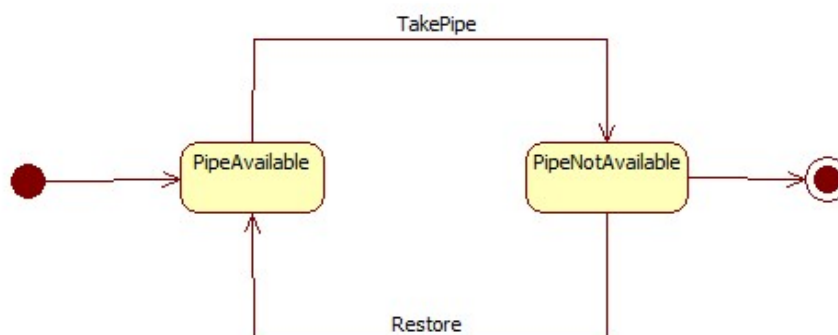
### 4.5.2 Is Pump Broken



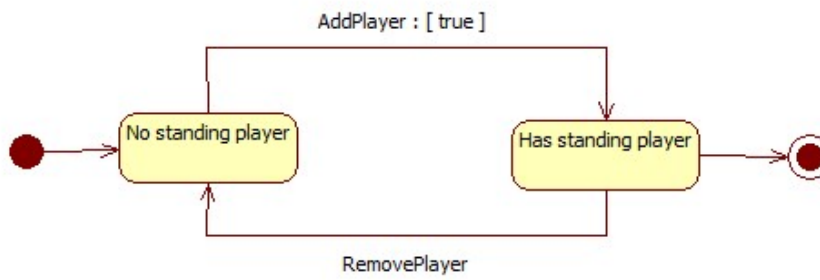
### 4.5.3 Mechanic Has Pipe



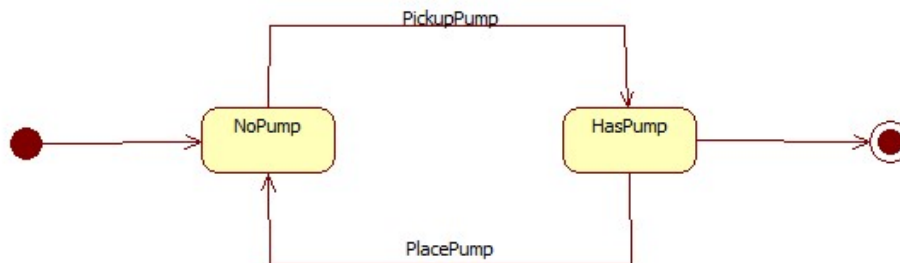
### 4.5.4 Pipe Availability At Cistern



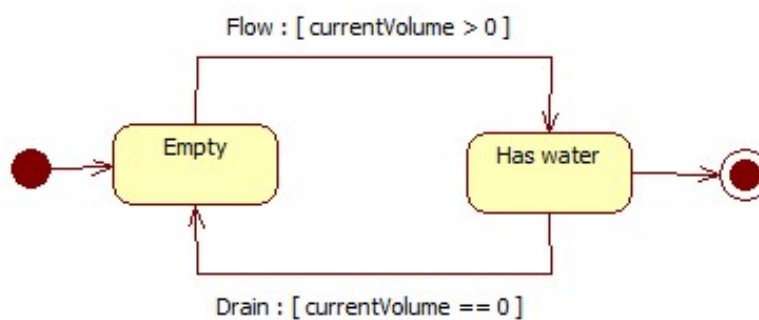
#### 4.5.5 Pipe Has Standing Players On It



#### 4.5.6 Mechanic Has Pump



#### 4.5.7 Water Volume In Pipe



**5. Napló**

Kezdet	Időtartam	Résztevők	Leírás
2023.03.22. 18:00	2.5 óra	Barabási Ganzer Kurcsi Nagy Palásti	Értekezlet. Döntés: Miket kell kijavítani, pontosítani, pótolni, feladatok kiosztása.
2023.03.23. 17:00	2 óra	Barabási	Javítások: osztálydiagram javítása. 4.5.7 állapot diagram elkészítése
2023.03.23. 15:00	2.5 óra	Nagy	Javítások: szekvenciadiagramok javítása. 4.5.6 állapot diagram elkészítése
2023.03.24. 13:00	3 óra	Palásti	Javítások: szekvenciadiagramok javítása. 4.5.4, 4.5.5 állapot diagramok elkészítése
2023.03.24. 12:00	2.5 óra	Ganzer	Javítások: szekvenciadiagramok javítása. 4.5.3 állapot diagram elkészítése
2023.03.25. 18:00	2 óra	Kurcsi	Javítások: dokumentum szerkesztése. 4.5.2 állapot diagram elkészítése