

4. Analízis modell kidolgozása

4.1 Objektum katalógus

4.1.1 Sarkkutató

Az egyik játszható karaktertípus. A játék elején 4 egységnyi testhővel kezd és meg tudja nézni, hogy egy szomszédos jégtábla hány embert bír el.

4.1.2 Eszkimó

Az egyik játszható karaktertípus. A játék elején 5 egységnyi testhővel kezd és képes iglut építeni.

4.1.3 Jégtábla

Ezen állhatnak a játékosok, de ha túl sokan állnak rajta egyszerre, akkor felborul és mindenki vízbeesik, aki rajta állt. Lehetnek tárgyak belefagyva, amiket a jégtáblát fedő hóréteg eltávolítása után lehet kiásni.

4.1.4 Lapát

Egyike a jégből kiásható tárgyaknak. Ha a játékosnak van ásója, akkor egyszerre 2 egységnyi havat takaríthat el egy munkaráfordítással.

4.1.5 Kötél

Egyike a jégből kiásható tárgyaknak. A játékosnak kötéltre van szüksége ahhoz, hogy kimenekítsen egy másik, vízbe esett játékost.

4.1.6 Búvárruha

Egyike a jégből kiásható tárgyaknak. Ha egy vízbeesett játékosnak van búvárruhája, akkor nem kell másnak kimenekítenie és folytathatja a körét, mintha semmi se történt volna.

4.1.7 Élelem

Egyike a jégből kiásható tárgyaknak. A felhasználásával a játékosok növelhetik a testhőjüket eggyel.

4.1.8 Iglu

Az eszkimók tudnak iglukat építeni. Az igluban el lehet bújni a vihar elől, így elkerülve a vihar miatti testhő csökkenést.

4.1.9 Hóvihar

Időnként feltámad egy hóvihar, ami néhány jégtáblát újabb réteg hóval borít be. Akit elkap a hóvihar, annak egységnyivel csökken a testhője.

4.1.10 Jelzőrakéta

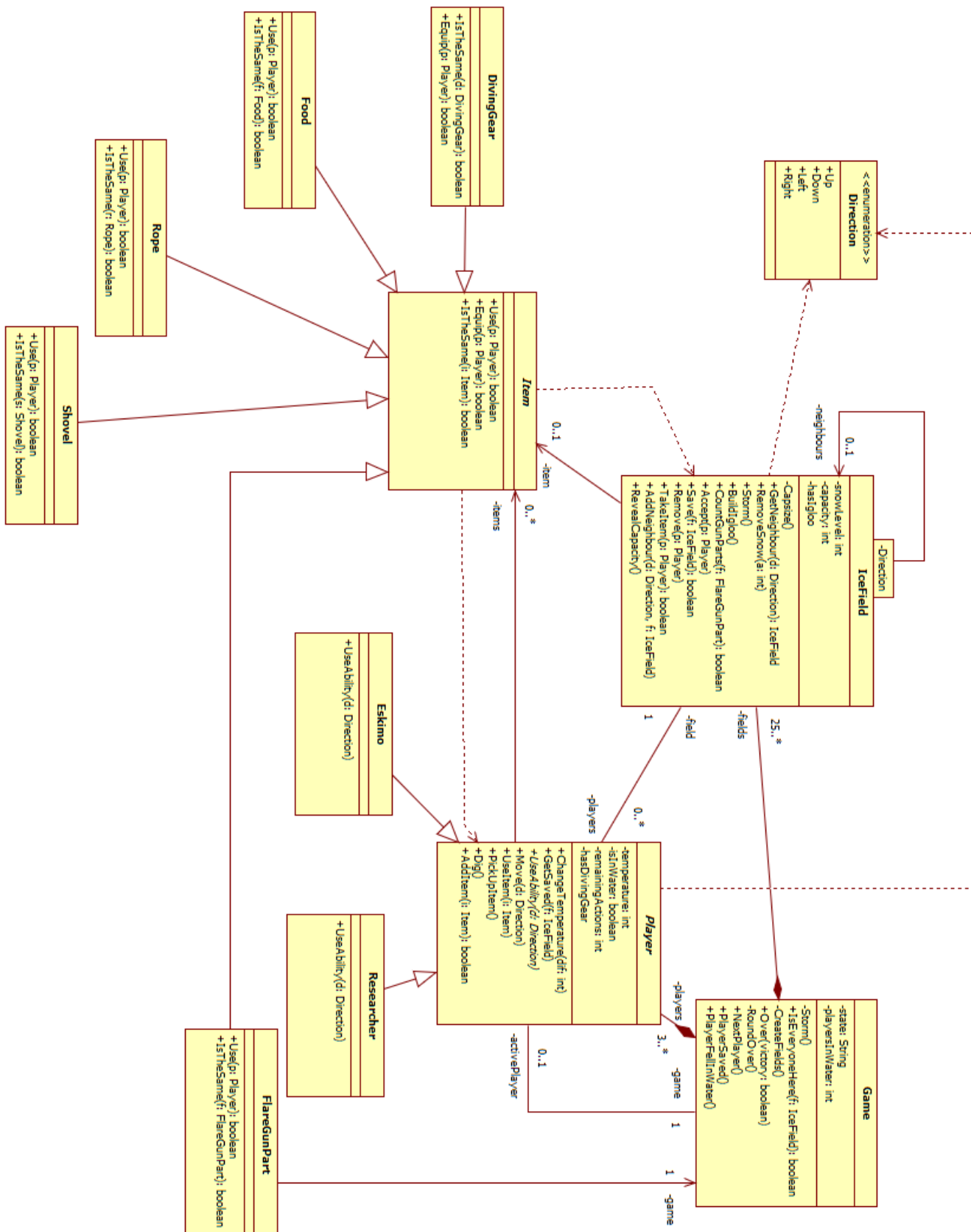
A játékot ennek az elsütésével lehet megnyerni, amikor minden játékos egy jégtáblán áll.

4.1.11 Jelzőfény, pisztoly, patron

Ezekre van szükség a jelzőrakéta összerakásához. Ezek a játék kezdetekor jégbe vannak fagyva és ugyanúgy lehet kiásni őket, mint a többi tárgyat.

4.2 Statikus struktúra diagramok

(Megjegyzés: a nevesített asszociációvégekhez és attribútumokhoz implicit getter és setter függvények tartoznak, amelyeket a diagram az olvashatóság kedvéért nem jelöl.)



4.3 Osztályok leírása

4.3.1 DivingGear

- **Felelősség**

A játékos túlél a vízben is kör végén, ha van neki, illetve ki tud jutni a vízből.

- **Ősosztályok**

Item → DivingGear

- **Metódusok**

- **boolean IsTheSame(DivingGear: d):** True-val tér vissza.
- **boolean Equip(Player p):** Ugyan azt csinálja, mint az Item, csak pluszba beállítja a játékoson a hasDivingGear-t.

4.3.2 Eskimo

- **Felelősség**

Képes Iglut építeni.

- **Ősosztályok**

Player → Eskimo

- **Metódusok**

- **boolean UseAbility(Direction: d):** Iglut épít oda, ahol éppen áll. True-val tér vissza, ha sikerült.

4.3.3 FlareGunPart

- **Felelősség**

Ezekből kell összegyűltetni mindhármat és használni egy helyen állva a játék megnyeréséhez.

- **Ősosztályok**

Item → FlareGunPart

- **Metódusok**

- **boolean Use(Player p):** Megnézi, hogy teljesülnek-e a győzelem feltételei és jelez a Game-nek, ha igen. True-val tér vissza, ha sikerült használni.
- **boolean IsTheSame(FlareGunPart: f):** True-val tér vissza.

4.3.4 Food

- **Felelősség**

Fel lehet használni a játékost testhőjének növelésére.

- **Ősosztályok**

Item → Food

- **Metódusok**
 - **boolean Use(Player p):** A játékos testhőjét növeli. True-val tér vissza.

4.3.5 Game

- **Felelősség**
Inicializálja a játékot. Nyilvántartja a játék állapotát. Minden kör végén vihart indít és ellenőrzi a játékosok állapotát.

- **Attribútumok**
 - **-state:** String: A játék állapota (folyamatban van, vége + nyert/vesztett)
 - **-fields:** IceField[]: A jégtablák.
 - **-players:** Player[]: A játékosok.
 - **-activePlayer:** player: A soron levő játékos.
 - **-playersInWater:** int: A vízbeesett játékosok száma.
- **Metódusok**
 - **boolean IsEveryoneHere(IceField f):** Megnézi, hogy mindenki az átvett fielden áll e.
 - **void Over(boolean victory):** Befejezi a játékot. Ha a paraméter true, akkor a játékosok nyertek.
 - **void NextPlayer():** Átállítja, hogy ki az aktuális játékos és átállítja a hátralevő akcióit 4-re. Ha vége egy körnek, akkor meghívja a RoundOver()-t.
 - **void PlayerSaved():** Jelzés, hogy egy játékost kimentettek a vízből.
 - **void PlayerFellInWater():** Jelzés, hogy egy játékos vízbe esett.

4.3.6 IceField

- **Felelősség**
Tárolja a jég tábla adatait (szomszédok, rajta álló játékosok, hószint, befagyott tárgy, iglu). Lehet mozogni közöttük, felborul, ha sokan állnak rajta, ki lehet ásni a tárgyat, ha nincs rajta hó. (A hóval fedett lyuk egy 0 kapacitású jég tábla.)
- **Attribútumok**
 - **-neighbours:** IceField: A szomszédos jég tablák a megadott irányba.
 - **-snowLevel:** int: A jég táblán levő hó réteg vastagsága.
 - **-players:** Player[]: A rajta álló játékosok.
 - **-item:** Item: A jég táblába fagyott tárgy.
 - **-capacity:** int: Ennyi játékos fér el rajta anélkül, hogy felborulna.
 - **-hasIgloo:** True, ha van rajta iglu.
- **Metódusok**
 - **IceField GetNeighbour(Direction d):** Visszaadja a szomszédos jég tablát a megadott irányba.
 - **void removeSnow(int a):** Csökkenti a hó réteg vastagságát a megadott értékkel.
 - **void Storm():** Növeli a hó réteget és csökkenti a rajta levő játékosok testhőjét, ha nincs a jég táblán iglu.
 - **void BuildIgloo():** Iglut épít a jég táblára

- **boolean CountGunParts(FlareGunPart f):** Megszámolja, hogy hány játékosnak van itt alkatrésze a jelzőpisztolyhoz. True-val tér vissza, ha az összes alkatrész megvan.
- **void Accept(Player p):** Felveszi a játékost a rajta álló játékosok közé. Felborul ha, túl sokan állnak így már rajta.
- **boolean Save(IceField f):** Végighívja a GetSaved(Icefield f)-et a rajta álló játékosokon. True-val tér vissza, ha legalább 1 játékos ki lett mentve.
- **void Remove(Player p):** Kiveszi a játékost a rajta állók listájából.
- **boolean TakeItem(Player p):** A kapott játékosnak kiássa a befagyott tárgyat. True-val tér vissza, ha sikerült.
- **void AddNeighbour(Direction d, IceField f):** Beállítja kapott irányba levő szomszédnak a kapott IceFieldet.
- **void RevealCapacity():** Felfedi, hogy hányan férnek el a jég táblán anélkül, hogy felborulna. Innentől kezdve végig látszik.
- **boolean IsTheSame(Food: f):** True-val tér vissza.

4.3.7 Item

- **Felelősség**

A használható tárgyak heterogén kollekciója.

- **Attribútumok**

- **-name:** string: A tárgy neve.(hogyan azonosítani lehessen)

- **Metódusok**

- **boolean Use(Player p):** A kapott játékos használja a tárgyat. True-val tér vissza, ha sikerült.
- **boolean Equip(Player p):** A kapott player felveszi a tárgyat. Ha sikerül True-val tér vissza.
- **boolean IsTheSame(Item: i):** False-al tér vissza. A leszármazottaknak van egy ugyan ilyen nevű függvényük, amiben magukból vesznek át egy példányt paraméterként és azok true-val térnek vissza. Így meg lehet nézni, hogy a kapott item ugyanolyan fajta-e, mint ez.

4.3.8 Player

- **Felelősség**

Nyilvántartja a játékos adatait. A játékos akcióit végrehajtja. Ha elfogynak az akciói, szól a Game-nek, hogy jöhet a következő.

- **Attribútumok**

- **-field:** IceField: A jég tábla, amin éppen áll
- **-temperature:** int: A játékos jelenlegi testhője
- **-isInWater:** boolean: True, ha vízbe van esve.
- **-remainingActions:** int: A hátralévő akciók száma a játékos körében.
- **-items:** Item[]: A játékos birtokolt tárgyai.
- **-game:** Game: A játék.
- **-hasDivingGear:** True, ha van búvárruhája.

- **Metódusok**

- **void ChangeTemperature(int dif):** Változtatja a játékos testhőjét a megadott értékkel(nem mehet 0 alá)
- **boolean GetSaved(IceField: f):** Kimenekítik a vízből a kapott IceField-re, ha vízbe van esve. True-val tér vissza, ha ki lett mentve.
- **void UseAbility(Direction: d):** Használja a játékos képességét a megadott irányba. Ha null-t vesz át, akkor ott, ahol áll a játékos. Levon egy akciót, ha sikerült.
- **void Move(Direction: d):** Másik jégtablálára mozog a megadott irányba. Levon egy akciót, ha sikerült.
- **void UseItem(Item i):** Használja a megadott tárgyat a játékos. **void PickupItem():** Kiássa a jégből a belefagyott tárgyat. Levon egy akciót, ha sikerült.
- **void Dig():** Eltakarít egy réteg havat a jégmezőről, ahol áll. **void AddItem(Item i):** Hozzáadja az Item-et a játékos tárgyaihoz. Levon egy akciót, ha sikerült.
- **void PickupItem():** Kiássa a jégtablából a befagyott tárgyat.
- **boolean AddItem(Item i):** A játékos táskájába teszi a tárgyat, ha még nincs neki egy ilyen. True-val tér vissza, ha sikerült.

4.3.9 Researcher

- **Felelősség**

Meg tudja nézni egy IceField kapacitását.

- **Ősosztályok**

Player → Researcher

- **Metódusok**

- **boolean UseAbility(Direction: d):** Meg tudja nézni egy szomszédos(vagy ahol áll, ha d null) IceField kapacitását. True-val tér vissza.

4.3.10 Rope

- **Felelősség**

Ki lehet menteni a játékosokat a vízből a környező mezőkről.

- **Ősosztályok**

Item → Rope

- **Metódusok**

- **boolean Use(Player p):** Kimenti a vízbeesett játékosokat a szomszédos mezőkről. True-val tér vissza, ha legalább egy játékost kimentett.
- **boolean IsTheSame(Rope: r):** True-val tér vissza.

4.3.11 Shovel

- **Felelősség**

Havat lehet vele eltakarítani a jégmezőről.

- **Össztályok**

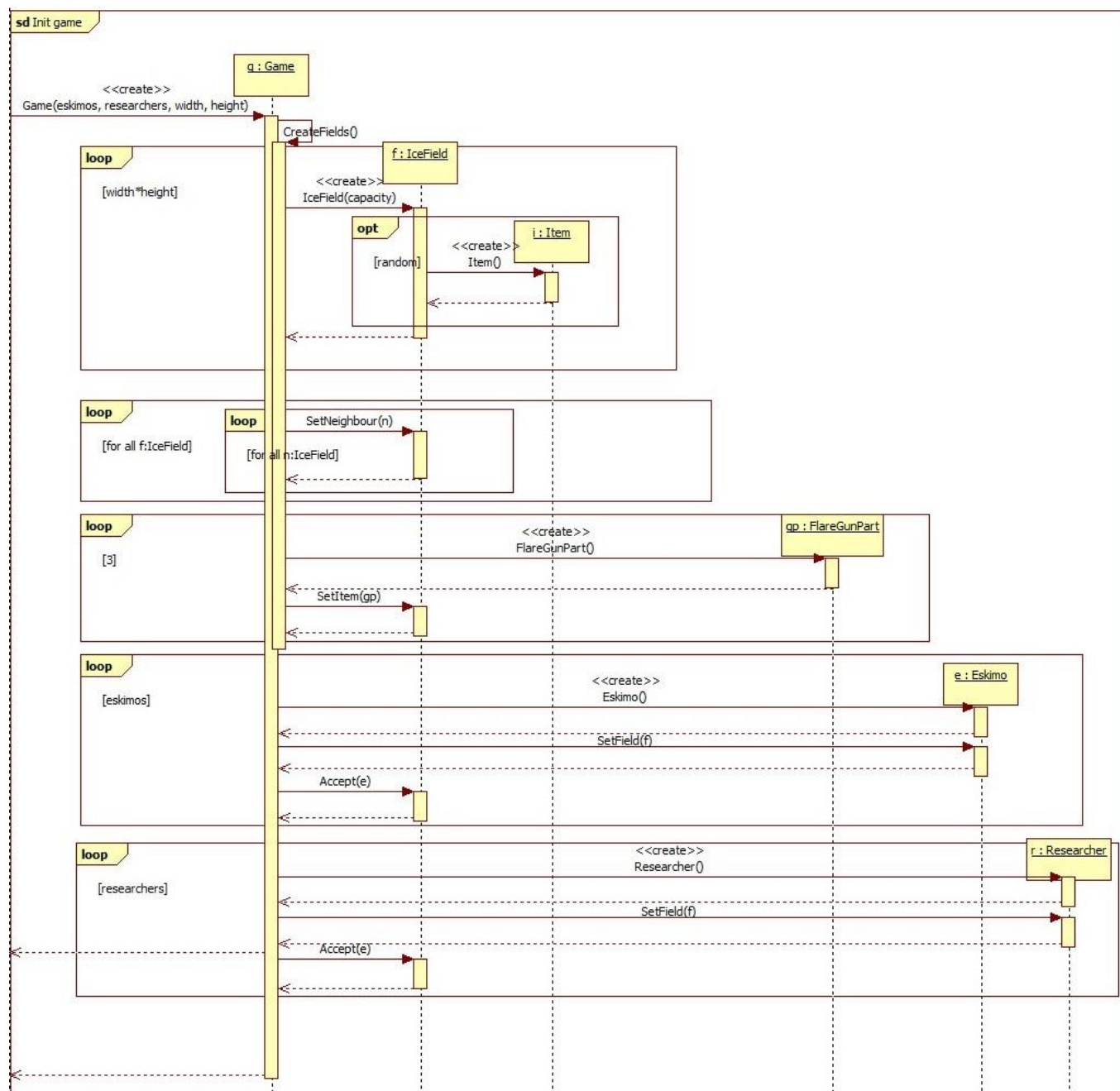
Item → Food

- **Metódusok**

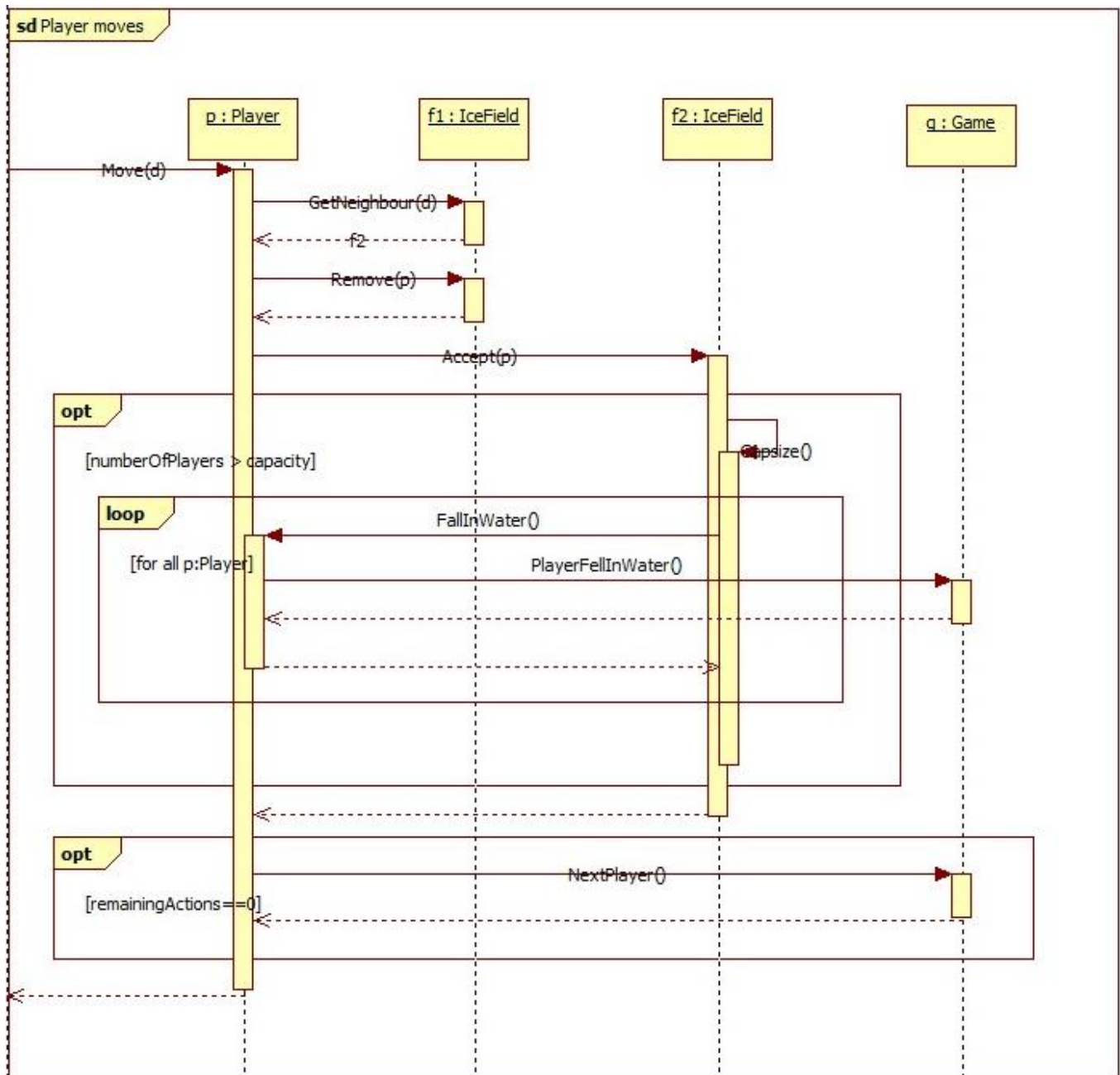
- **boolean Use(Player p):** Két egységnyi hóréteget takarít el a jégmezőről, ahol a használója áll.
- **boolean IsTheSame(Shovel: s):** True-val tér vissza.

4.4 Szekvencia diagramok

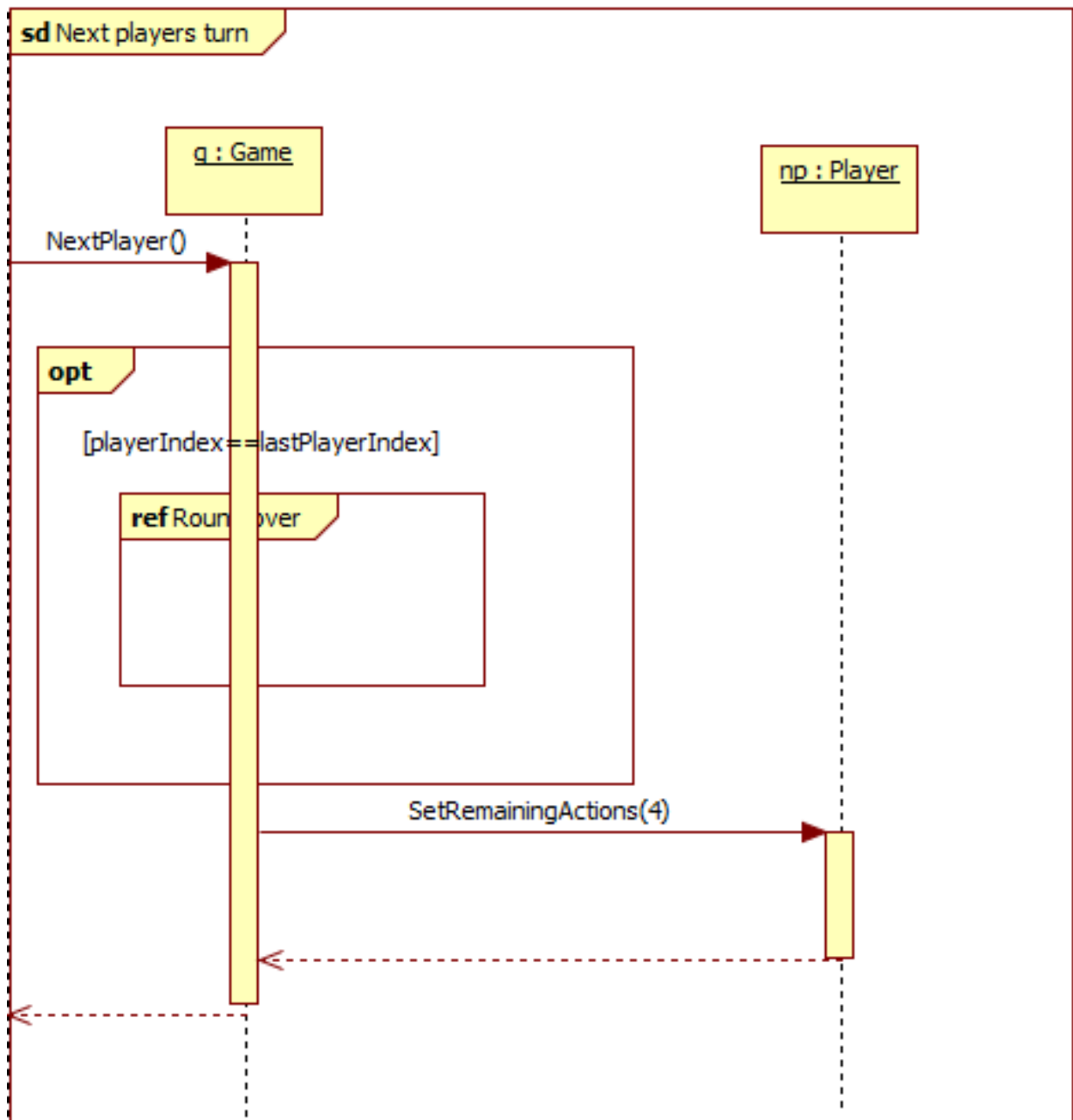
4.4.1 Init Game



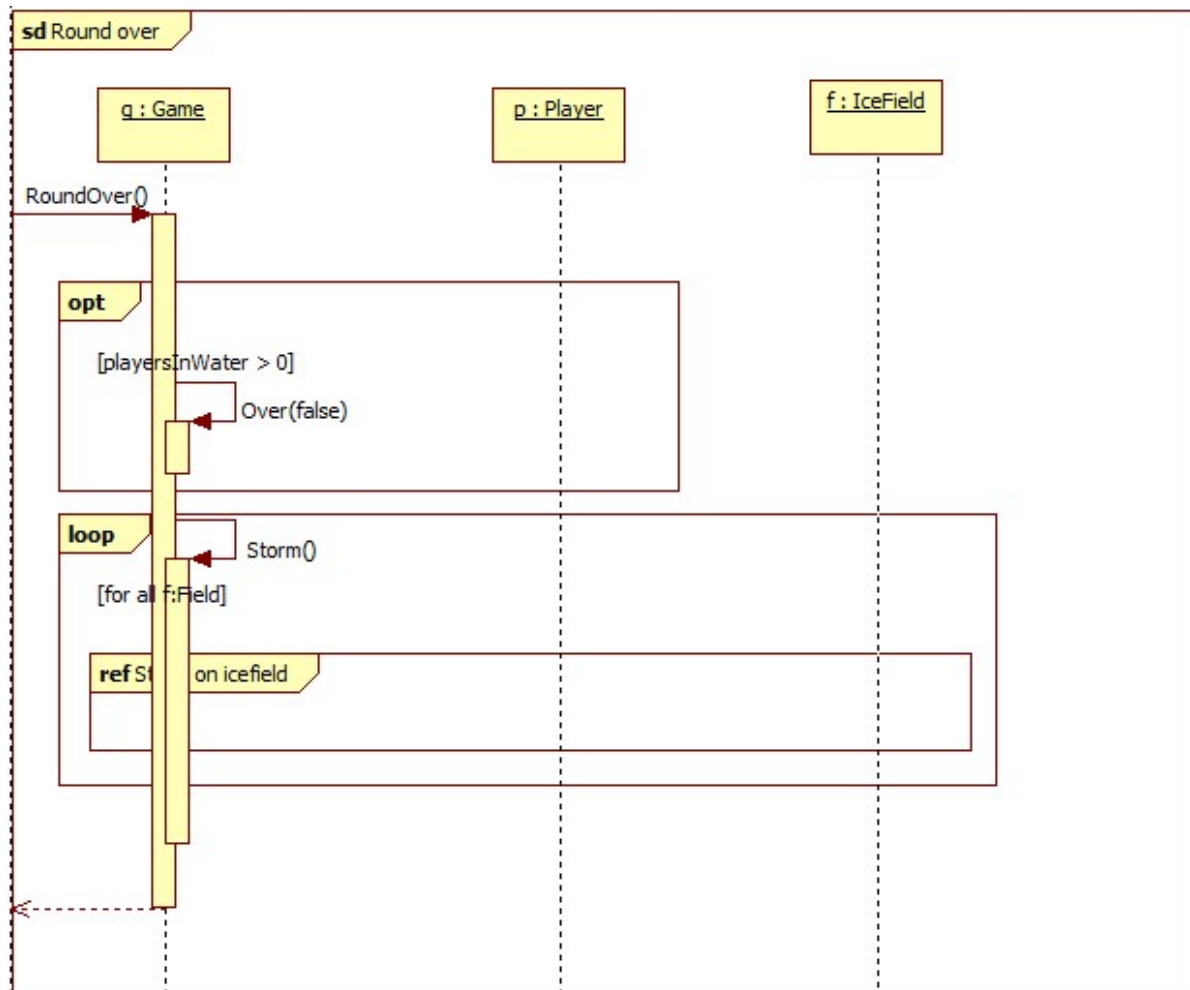
4.4.2 Player Moves



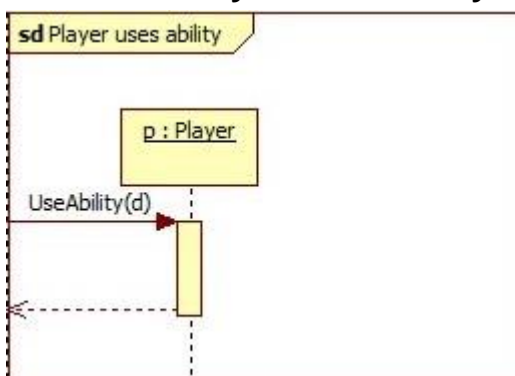
4.4.3 Next Player's turn



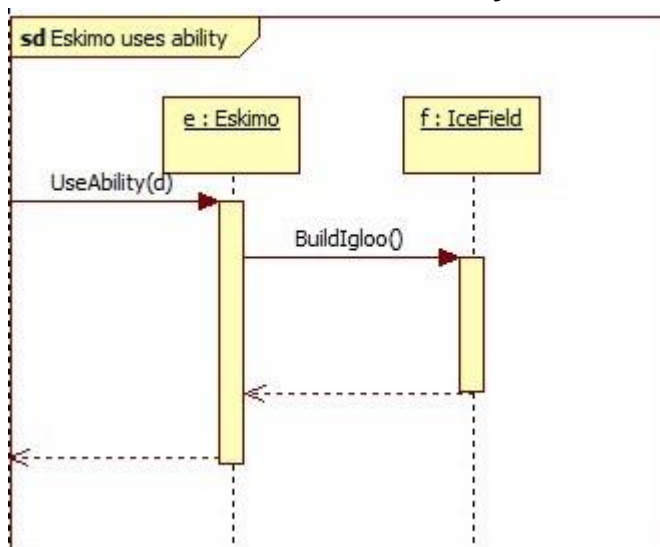
4.4.4 Round Over



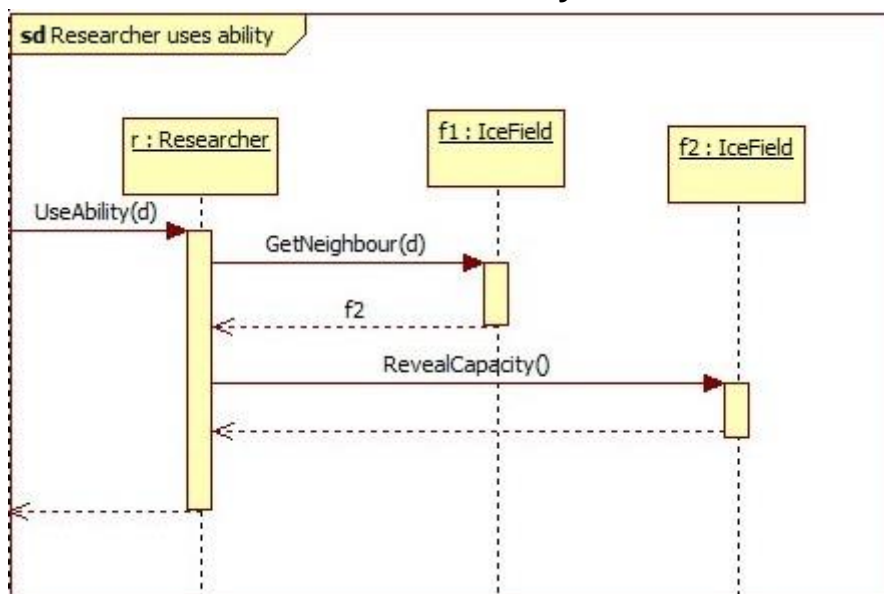
4.4.5 Player Uses Ability



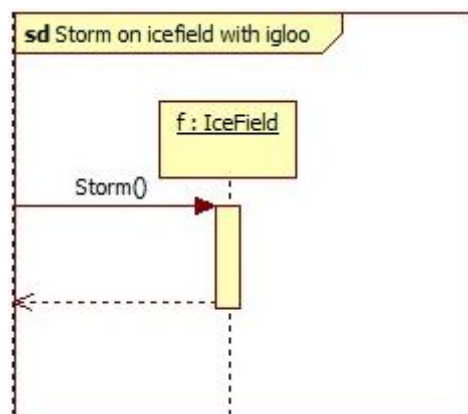
4.4.6 Eskimo Uses Ability



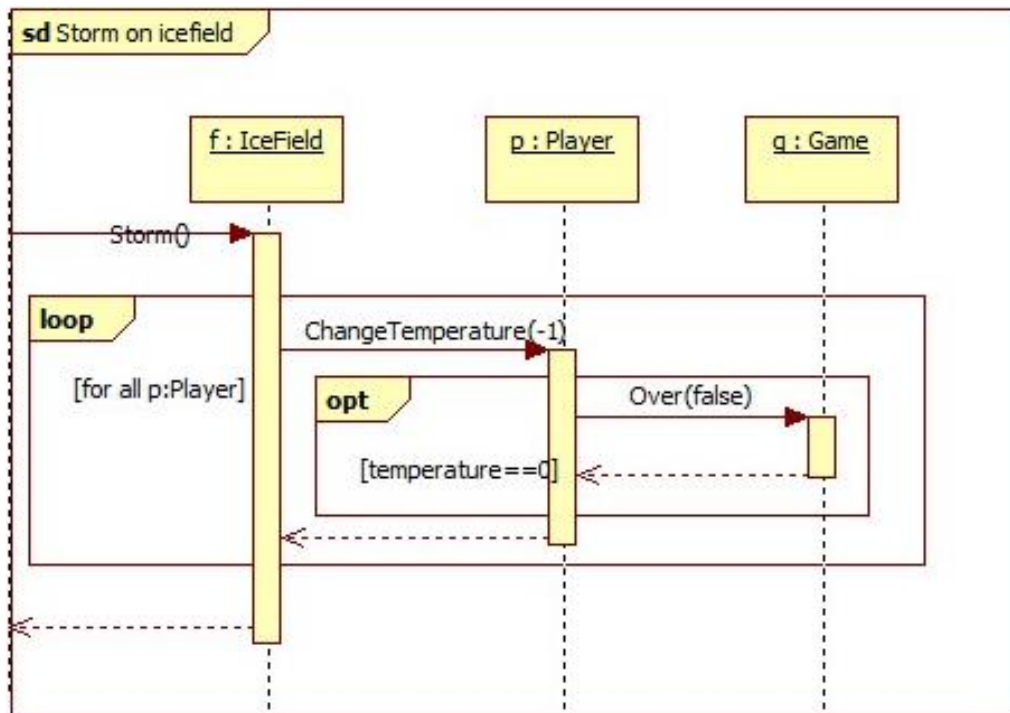
4.4.7 Researcher Uses Ability



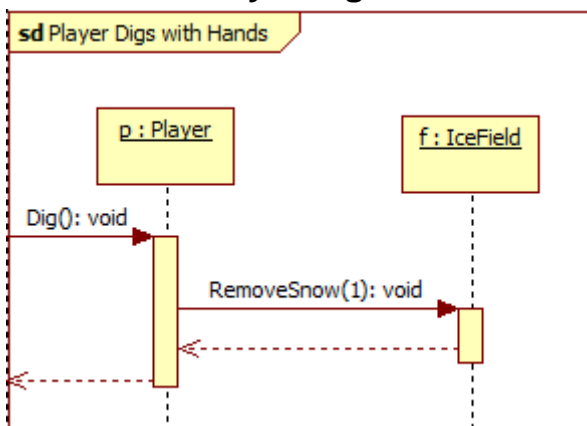
4.4.8 Storm on Icefield with Igloo



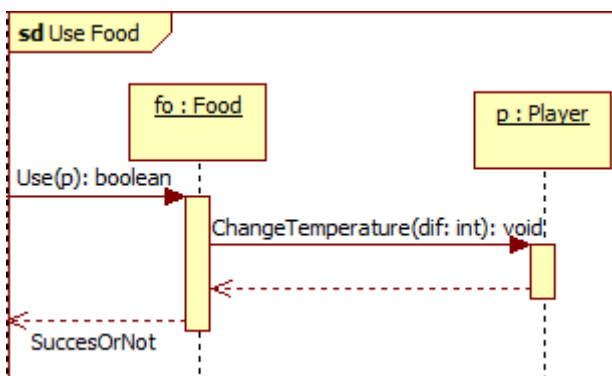
4.4.9 Storm On Icefield



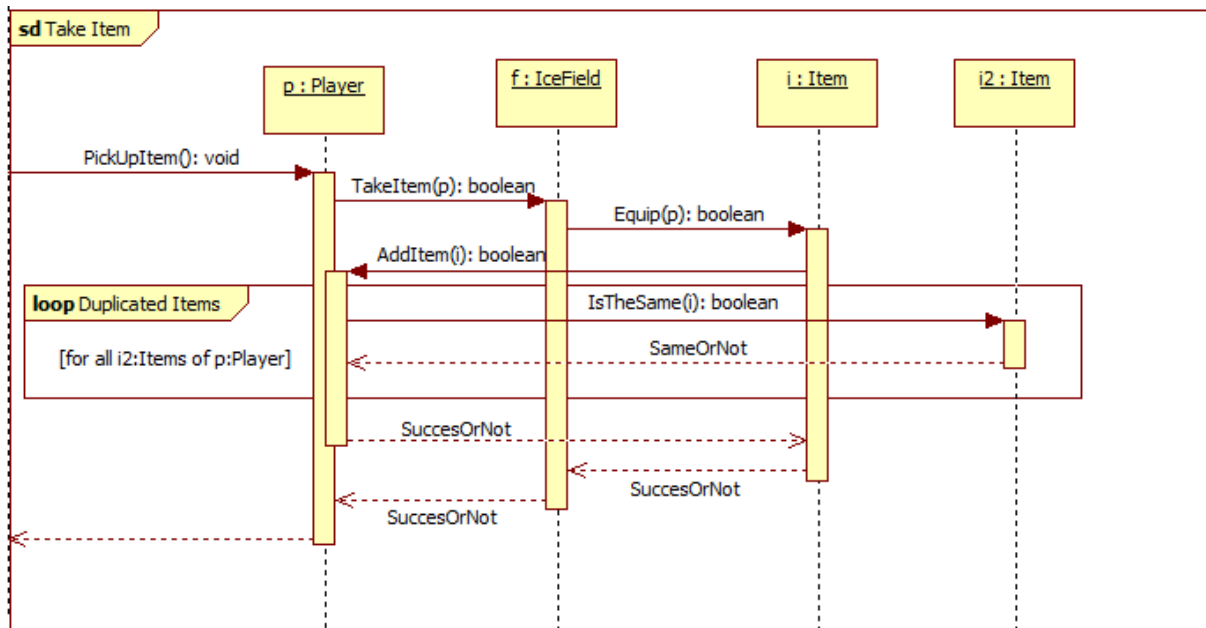
4.4.10 Player Digs with Hands



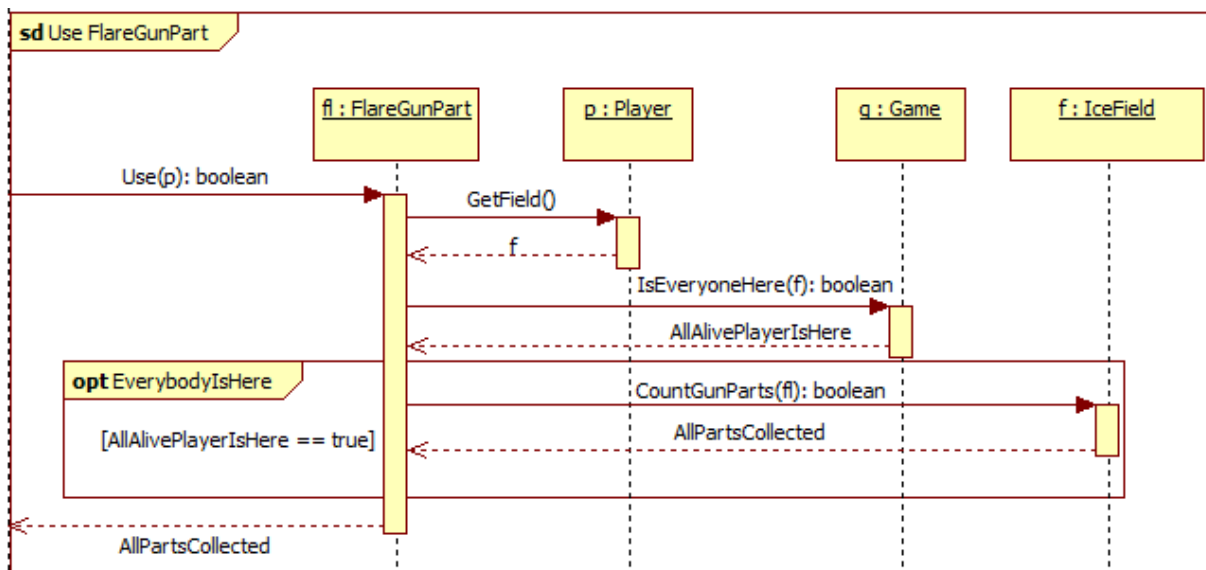
4.4.11 Use Food



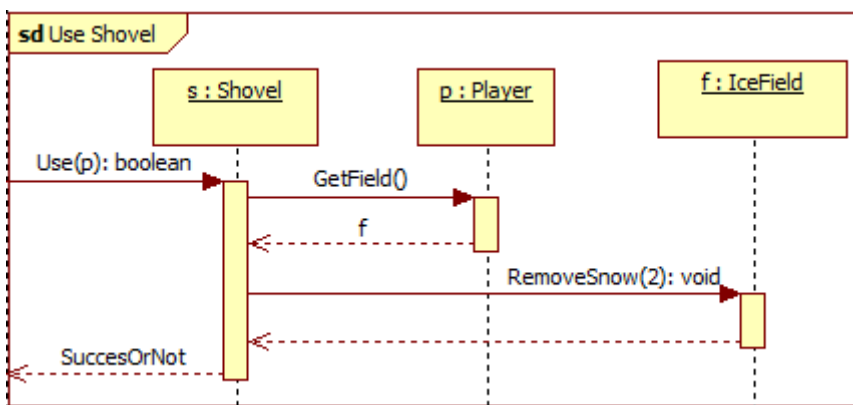
4.4.12 Take Item



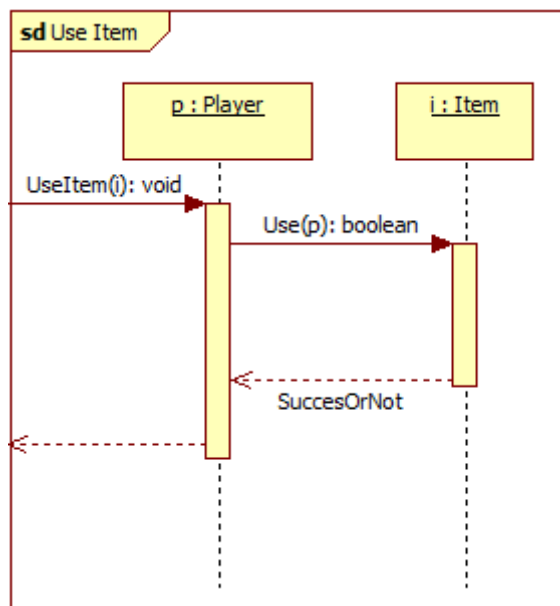
4.4.13 Use Flare Gun Part



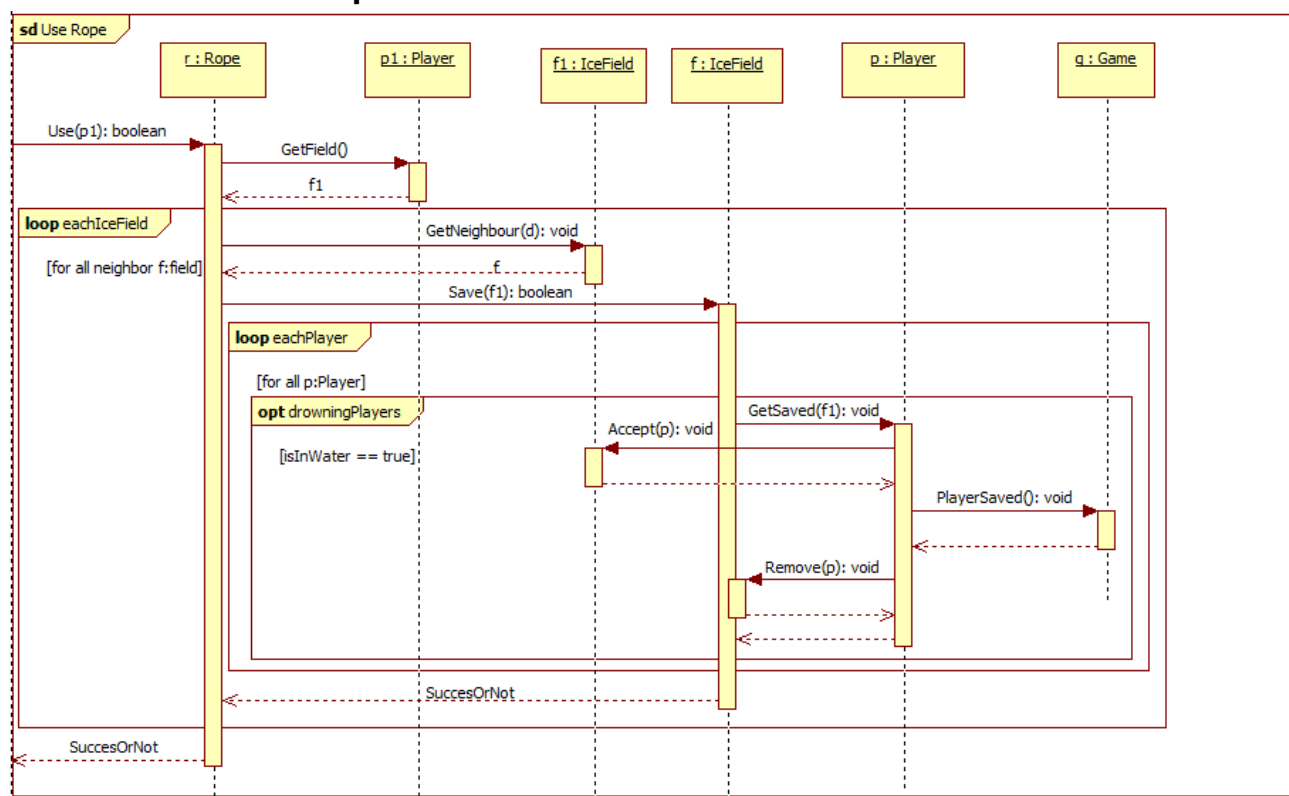
4.4.14 Use Shovel



4.4.15 Use Item



4.4.16 Use Rope



4.5 State-chartok

Állapotgépre nincs szükség a programterv bemutatásához.

4.6 Napló

Kezdet	Időtartam	Résztevők	Leírás
2020.03.07. 16:00	1 óra	Karmacsi Suciu Seres Takács	Megbeszélés: Változtatások egyeztetése.
2020.03.07. 16:30	1 óra	Seres	Szekvenciadiagramokon változtatások eszközlése: 3.4.11., 3.4.12., 3.4.15
2020.03.07. 16:30	1 óra	Karmacsi	Osztálydiagramon és leíráson változtatások: 3.2, 3.3
2020.03.07. 10:30	1 óra	Takács	Szekvenciadiagramokon változtatások eszközlése: 3.4.2., 3.4.4., 3.4.8
2020.03.09. 11::00	1 óra	Suciu	Dokumentum véglegesítése, nyomtatás