

4. Analízis modell kidolgozása

48 – asd

Konzulens:

Ludmány Balázs

Csapattagok

Bödöcs Richárd	FG7VG1	bodocsr@gmail.com
Halász Bence Botond	Q7RW4E	worldofsteves@gmail.com
Hátori Péter Sándor	SGLJSB	hampet97@gmail.com
Lauber Anna	H9GBKF	lauber.anna@gmail.com
<u>Pintér Tamás</u>	JY4D5L	pintertamas99@gmail.com

2021.03.08.

4. Analízis modell kidolgozása

4.1 Objektum katalógus

4.1.1 Aszteroida

A játék alappillére. A játékosok az aszteroidákon mozoghatnak. Az aszteroidák tartalmaznak valamilyen nyersanyagot, valamint különböző vastagságú köpenyük van (0-10 réteg). Az aszteroidák ismerik a velük szomszédos aszteroidákat.

4.1.2 Aszteroidaöv

Az aszteroidaöv a pályát, azaz a térképet jelképezi. Ezen találhatóak az aszteroidák és a Nap. Felelőssége a játék inicializálása, a játék megnyerésének és elvesztésének ellenőrzése, az aszteroidák törlése a pályáról, valamint a napkitörések kezelése.

4.1.3 Figura

A játékban különböző figurák tudnak fúrni aszteroidákat és mozogni a szomszédos aszteroidákon vagy teleportkapu-párokon keresztül. Meg tudnak halni és különbözően reagálnak a radioaktív robbanásokra.

4.1.4 Köpeny

Az aszteroidákat különböző vastagságú köpenyek borítanak (0-10 réteg). Ezeket a rétegeket csökkentik a figurák fúrással.

4.1.5 Nyersanyag

A nyersanyag az aszteroidákban található. Különbözően reagálnak a kibányászásra a naptávolság függvényében.

4.1.6 Robot

A robotokat egy mesterséges intelligencia irányítja, képesek fúrni és mozogni szomszédos aszteroidákra, valamint teleportkapu-párokon át. Képesek meghalni a napviharban, ha nem üreges, teljesen kifúrt köpenyű aszteroidán állnak. A radioaktív robbanást túlélnek és egy szomszédos aszteroidán landolnak.

4.1.7 Szén

A szén egy nyersanyag, amelyet mindenféle következmény nélkül ki tudnak bányászni a telepesek.

4.1.8 Telepes

A telepeseket irányítják a játékosok. A telepések képesek fúrni, mozogni szomszédos aszteroidákra vagy teleportkapu-párok segítségével, bányászni, nyersanyagot az aszteroida magjába helyezni, robotot építeni és teleportkapu-párt építeni, lerakni. A telepések meg tudnak halni a napviharban, ha nem üreges, kifúrt aszteroidán állnak, valamint a radioaktív robbanásokat sem élik túl.

4.1.9 Teleportkapu

A teleportkapuknak van egy teleportkapu-párjuk, amelyek ha le vannak helyezve két nem szomszédos aszteroidára, akkor segítségükkel a figurák mozoghatnak az adott aszteroidák közt.

4.1.10 Urán

Az urán egy nyersanyag, melyhez ha lefűr egy figura és napközelben van, akkor felrobbantja az őt tartalmazó aszteroidát.

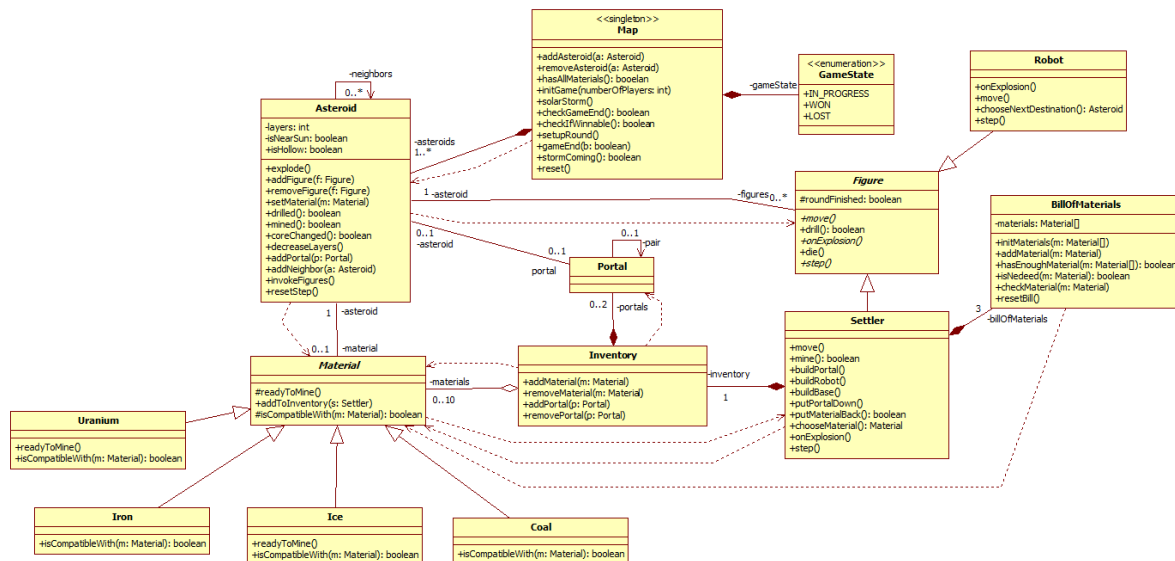
4.1.11 Vas

A vas egy nyersanyag, amelyet mindenféle következmény nélkül ki tudnak bányászni a telepések.

4.1.12 Vízjég

A vízjég egy nyersanyag, melyhez ha lefűr egy figura és napközelben van, akkor megolvad és eltűnik, tehát az őt tartalmazó aszteroida üregessé válik.

4.2 Statikus



struktúra diagramok

4.3 Osztályok leírása

4.3.1 Map

• Felelősség

A map osztály a játék alapvető logikáját biztosítja, számon tartja az összes aszteroidát, a játék aktuális állapotát, továbbá a körök lebonyolítását (lesz egy napvihar a következő körben, ki lép), továbbá, hogy nyerhető-e a játék vagy esetleg nyertek-e már a játékosok.

• Ősosztályok

-

• Interfészek

-

• Attribútumok

- - **gameState: GameState:** A játék aktuális állapotát tárolja el
- - **asteroids: ArrayList<Asteroid>:** Eltárolja az összes aszteroidát, amely a pályát alkotja

• Metódusok

- **+initGame(numberOfPlayers: int): void:** A játékot inicializálja, létrehozza az aszteroidákat és a játékosokat.
- **+ addAsteroid(a: Asteroid): void:** Hozzáad egy aszteroidát a térképhez (aszteroidaövhöz).
- **+ removeAsteroid(a: Asteroid): void:** Leszed egy aszteroidát a térképről.
- **+ hasAllMaterials(): boolean:** Leellenőrzi, hogy megvan-e minden nyersanyag az aszteroidaöv valamelyik aszteroidáján.
- **+ solarStorm(): void:** Generál egy napvihart ami végigsöpör a pályán..
- **+ checkGameEnd(): boolean:** Lellenőrzi, hogy vége lett-e már a játéknak vagy sem.
- **+ checkIfWinnable(): boolean:** Megnézi hogy nyerhető-e a játék vagy sem.
- **+ setupRound(): void:**
- **+ gameEnd(boolean b): boolean:** Kezeli a játék végét.
- **+ stormComing(): boolean:** Random kisorosolja, hogy az adott körben lesz-e napvihar
- **+ reset(): void:** Meghívja a játékban lévő összes aszteroidán a resetStep() függvényt.

4.3.2 Asteroid

· Felelősség

Az aszteroida(Asteroid) osztály valósítja meg a map-et építő kis elemeket. Számon tartja és frissíti az építőelem(aszteroida) tulajdonságait a játék fokozatos haladásával egyidejűleg.

· Ősosztályok

-

· Interfészek

-

· Attribútumok

- **- layers: int :** Ez az attribútum tárolja, hogy milyen vastag az aszteroida köpenye.
- **- isNearSun: boolean:** Ez azt jelöli, hogy az aszteroida napközeli van-e, vagy sem.
- **isHollow: boolean:** Ez jelöli, hogy az aszteroida üreges vagy sem.
- **- portal: Portal:** Tárolja az aszteroidán lévő portált.
- **- figures: ArrayList<Figure>:** Tárolja az aszteroidán lévő figurákat.
- **- material: Material:** Tárolja az aszteroida belsejében lévő nyersanyagot.
- **- neighbors: ArrayList<Asteroid>:** Az aszteroida szomszédjait tárolja.

· Metódusok

- **+ explode(): void:** Az aszteroida törlését végzi, amennyiben az felrobbant.
- **+ addFigure(f: Figure): void:** Hozzáad egy figurát az aszteroidához.
- **+ removeFigure(f: Figure): void:** Eltávolít egy figurát az aszteroidáról.

- **+ setMaterial(m: Material): void:** Beállítja az aszteroida magjában található nyersanyag típusát. (Ebbe beletartozik az üreges is.)
- **+ decreaseLayers(): void:** Csökkenti eggyel az aszteroida köpenyét.
- **+ addPortal(p: Portal): void:** Hozzáad egy portált az aszteroidához.
- **+ addNeighbor (a: Asteroid): void:** Hozzáad egy aszteroidát az aszteroida szomszédjaihoz.
- **+ invokeFigures(): void:** Minden rajta álló figurát léptet vagyis meghívja a step metódusukat.
- **+ resetStep(): void:** Meghívja minden az asteroidán álló figurán a setRoundFinished függvényt false értékkel, vagyis kinullázza azt, hogy a karakterek léptek-e már.
- **+ drilled(): boolean:** Amikor egy figura drillel ez a függvény hívodik meg az asteroidán és különböző értékek alapján vagy sikeres lesz vagy nem.
- **+ mined(): boolean:** Hasonlóan működik mint a drilled csak a settler mine() függvénye hívja meg.

4.3.3 Portal

· Felelősség

A portál osztály valósítja meg azt a tárgyat amelyet a játékosok építeni tudnak és lehelyezni továbbá azok között teleportálni így a telepések nem csak szomszédos mezőre lesznek képesek lépni. Mindig 1-1 teleportkapu áll párban, ezek között lehet utazni. A kapuk mindig tudják ki a párjuk ezt számon tartják egy változóban.

· Ősosztályok

-

· Interfészek

-

· Attribútumok

- - **pair: Portal:** Innen fogja tudni a portál, hogy melyik a párja.
- - **asteroid: Asteroid:** Ez a portált tartalmazó aszteroida.

· Metódusok

-

4.3.4 Figure

· Felelősség

A figura(Figure) absztrakt osztály reprezentálja az összes telepés és robot összességét. Tudják fúrni az aszteroida köpenyét, és meg tudnak halni.

· Ősosztályok

-
- **Interfészek**

-
- **Attribútumok**

- - **asteroid: Asteroid**: Ez az az aszteroida amina Figure áll.
- **# roundFinished: boolean**: Azt nézi, hogy a figura lépett-e már a körben.

- **Metódusok**

- **+ move(): void**: Absztrakt függvény a mozgás megvalósításához.
- **+ drill(): boolean**: A fúrást teszi lehetővé.
- **+ onExplosion(): void**: Robbanás alkalmával hívódik meg ez a függvény, ami a Figure típusától függően fog műveleteket végrehajtani.
- **+ die(): void**: Ez a függvény kezeli a figurák halálát.
- **+ step(): void**: Eldönti a figura következő lépését.

4.3.5 Robot

- **Felelősség**

Ez az osztály reprezentálja a robotokat. Mozgásuk egy mesterséges intelligencia döntése alapján történik. A aszteroida robbanása után egy szomszédos aszteroidán landolnak

- **Ősosztályok**

Figure

- **Interfészek**

-
- **Attribútumok**

-
- **Metódusok**

- **+ onExplosion(): void**: Híváskor átrakja egy véletlenszerűen választott szomszédos aszteroidára a robotot.
- **+ move(): void**: Ez a függvény hívódik meg, amikor a robot mozogni akar.
- **+ step(): void**: Eldönti, hogy a robot lép-e vagy fúr.
- **+ chooseNextDestination(): Asteroid**: Kiválasztja a következő aszteroidát, amelyre lépni fog.

4.3.6 Settler

- **Felelősség**

Ez az osztály reprezentálja a telepeseket. Mozgáskor a játékos választhat aszteroidát, amire tovább akarja léptetni a telepest. Bányászhat, építhet robotot, portált, ezeket pedig le is tudja rakni. A kibányászott nyersanyagokat vissza tudja tenni egy üreges aszteroida belsejébe. Ha megfelelnek a kritériumoknak az aszteroidán álló telepések, akkor építhetnek egy bázist is, ami a játék megnyeréséhez vezeti őket. Robbanáskor meghalnak.

- **Ősosztályok**

Figure

- **Interfészek**

-

- **Attribútumok**

- **- inventory: Inventory:** Ez a telepes táskája, amiben a nyersanyagokat és portálokat tárolja.
- **- billOfMaterials: List<BillOfMaterials>:** Receptek, ami tárgyak (robot, teleportkapu, base) építéséhez szükséges nyersanyagokat tárolják.

- **Metódusok**

- **+ move(): void:** Ez a metódus kezeli a játékos által választott mezőre átléptetését a telepesnek.
- **+ mine(): boolean:** A telepes kibányászik egy nyersanyagot.
- **+ buildPortal(): void:** Megépít egy portál párt a megfelelő nyersanyagokért cserébe.
- **+ buildRobot(): void:** Megépít egy robotot a megfelelő nyersanyagokért cserébe.
- **+ putPortalDown(): void:** Letesz egy portált az adott aszteroidára.
- **+ putMaterialBack(): boolean:** Visszatesz egy nyersanyagot az aszteroida belsejébe, ha az már üres.
- **+ chooseMaterial(): Material:** Kiválasztja a nyersanyagot.
- **+ onExplosion(): void:** Robbanáskor a telepes meghal.
- **+ buildBase(): void:** Felépíti a bázist a játékos, ezzel megnyerve a játékot.
- **+ step(): void:** A játékos eldönti, hogy lépni, fúrni vagy ásni szeretne.

4.3.7 Inventory

- **Felelősség**

Ennek az osztálynak a felelőssége a játékosoknál lévő nyersanyagok és portálok tárolása. Egyszerre maximum 2 portál és 10 nyersanyag lehet egy játékosnál.

- **Össosztályok**

-

- **Interfészek**

-

- **Attribútumok**

- - **materials: HashMap<Material, Integer>**: Itt tároljuk a nyersanyagokat (maximum 10)
- - **portals: ArrayList<Portal>**: Itt tároljuk a portálokat (maximum 2).

- **Metódusok**

- + **addMaterial(m: Material): void**: Hozzáad egy nyersanyagot az inventoryhoz.
- + **removeMaterial(m: Material[]): void**: Kiveszi a listában lévő számú és típusú nyersanyagot az inventoryból.
- + **addPortal(p: Portal): void**: Hozzáad egy portál-párt az inventoryhoz.
- + **removePortal(p: Portal): void**: Kivesz egy portált az inventoryból.

4.3.8 Material

- **Felelősség**

Ebből az osztályból származnak le a nyersanyagok. Két függvénye van, ami a mined() és az addToInventory(). A mined() akkor fog hívódni, amikor az aszteroida kéréget kifűrták, az addToInventory pedig, amikor kibányászták a magot.

- **Össosztályok**

-

- **Interfészek**

-

- **Attribútumok**

- - **asteroid: Asteroid**: Itt tároljuk a nyersanyaghoz tartozó aszteroidát.

- **Metódusok**

- **# readyToMine(): void**: Ez a függvény akkor hívódik meg, ha a nyersanyagot fedő köpenyt lefűrték. Ekkor a nyersanyagnak jelzi az aszteroida hogy készen áll a kibányászásra, a mag pedig reagálni tud erre, például azzal, hogy felrobban, vagy szublimál.

- **+ addToInventory(s: Settler): void:** Hozzáadja a Settler inventoryjához az adott nyersanyagot.
- **# isCompatibleWith(m: Material): boolean:** Az objektum ellenőrizheti, hogy olyan-e mint ő anélkül, hogy a többieket ismernie kellene.

4.3.9 Iron

- **Felelősség**

Ez az osztály reprezentálja a vasat.

- **Ősosztályok**

Material

- **Interfészek**

-

- **Attribútumok**

-

- **Metódusok**

- **+ isCompatibleWith(m: Material): boolean:** Az objektum ellenőrizheti, hogy olyan-e mint ő anélkül, hogy a többieket ismernie kellene.

4.3.10 Ice

- **Felelősség**

Ez az osztály reprezentálja a vízjeget. Ha napközben bányásszák ki, akkor szublimál.

- **Ősosztályok**

Material

- **Interfészek**

-

- **Attribútumok**

-

- **Metódusok**
- **+ readyToMine(): void:** Meghívásakor átállítja az aszteroida belsejét Hollow-ra, ezzel jelezve, hogy szublimált.
- **+ isCompatibleWith(m: Material): boolean:** Az objektum ellenőrizheti, hogy olyan-e mint ő anélkül, hogy a többieket ismernie kellene.

4.3.11 Coal

- **Felelősség**
Ez az osztály reprezentálja a szenet.
- **Ősosztályok**
Material
- **Interfészek**
-
- **Attribútumok**
-
- **Metódusok**
- **+ isCompatibleWith(m: Material): boolean:** Az objektum ellenőrizheti, hogy olyan-e mint ő anélkül, hogy a többieket ismernie kellene.

4.3.12 Uranium

- **Felelősség**
Ez az osztály reprezentálja az urániumot. Ha napközben bányásszák ki, akkor az őt tartalmazó aszteroida felrobban.
- **Ősosztályok**
Material
- **Interfészek**
-
- **Attribútumok**
-

- **Metódusok**
- **+ readyToMine(): void** Meghívásakor felrobbantja az aszteroidát.
- **+ isCompatibleWith(m: Material): boolean:** Az objektum ellenőrizheti, hogy olyan-e mint ő anélkül, hogy a többieket ismernie kellene.

4.3.13 GameState

- **Felelősség**
A GameState egy enumeration valósítja meg a játék lehetséges állapotait.
- **Ősosztályok**
-
- **Interfészek**
-
- **Attribútumok**
- **+ IN_PROGRESS:** Ez az attribútum reprezentálja azt az állapotot, amikor játék közben vagyunk.
- **+ WIN:** Ez az állapot reprezentálja azt az állapotot, amikor a játékosok nyertek.
- **+ LOST:** Ez az állapot reprezentálja azt az állapotot, amikor a játékosok vesztek.
- **Metódusok**
-

4.3.14 BillOfMaterials

- **Felelősség**
Az osztály teszi lehetővé, hogy tesztelni lehessen a különböző építési feltételeket.
- **Ősosztályok**
-
- **Interfészek**
-
- **Attribútumok**
- **- material: Material[]:** Tárolja egy listában egy adott dolog építéséhez szükséges materialokat. pl.: Portál építése 2 egység vas 1 egység víz 1 egység urán akkor a tároló tartalma 2 vas objektum 1 víz objektum 1 urán objektum

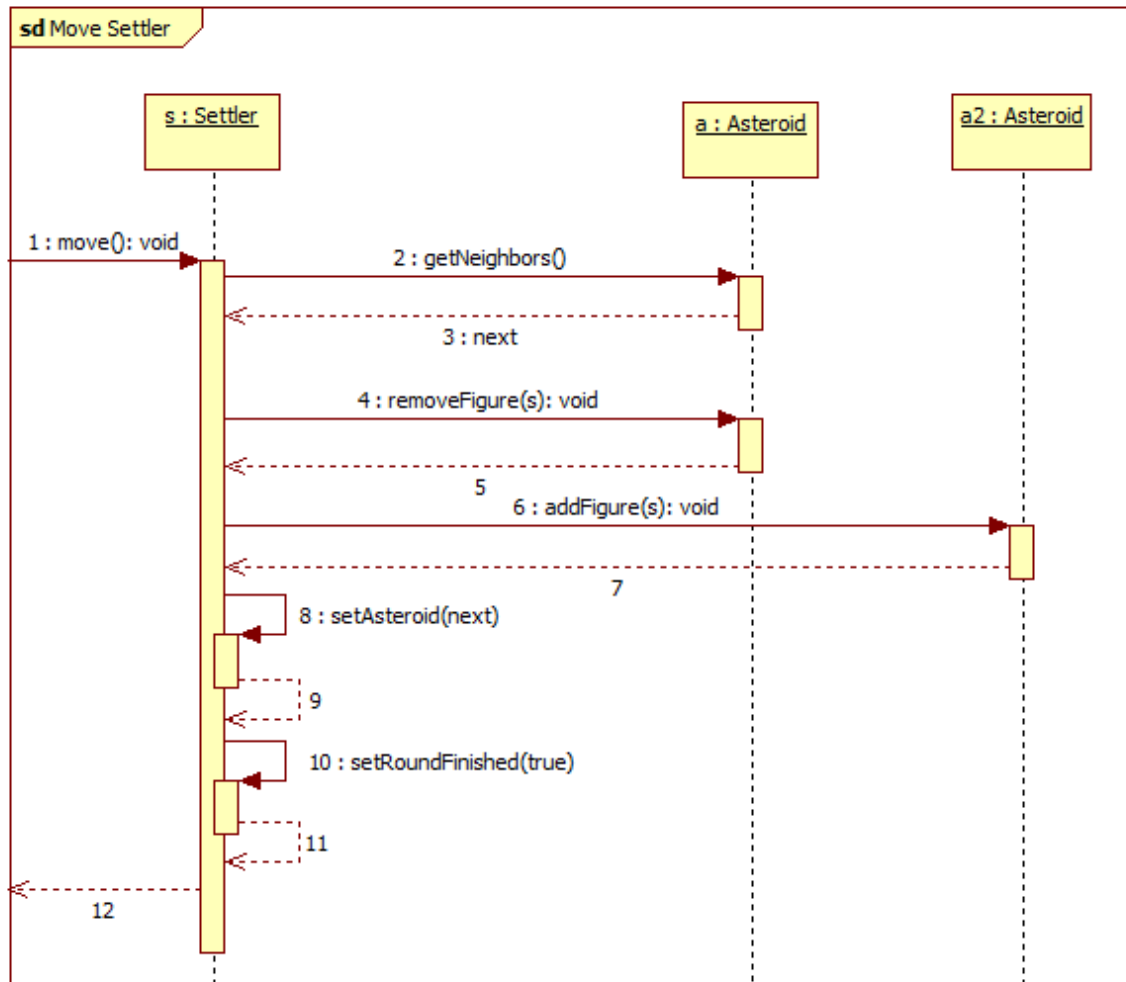
· **Metódusok**

- **+ initMaterials(m: Material[]):** Feltölti a tárolót olyan objektumokkal amik az építés feltételei.
- **+ addMaterial(m: Material):** Hozzáad egy materialt a listához
- **+ isNeeded(m: Material):** Megnézi hogy a paraméterként kapott nyersanyag kell-e az építéshez
- **+ resetBill(): void:** Visszatölti az árát a különböző építményeknek.
- **+ checkMaterial(m: Material):** Ami nyersanyag megvan azt kiveszi a listából.
- **+ hasEnoughMaterial(m: Material[]): boolean:** Megnézi, hogy elég nyersanyag áll-e rendelkezésre.

4.4 Szekvencia diagramok

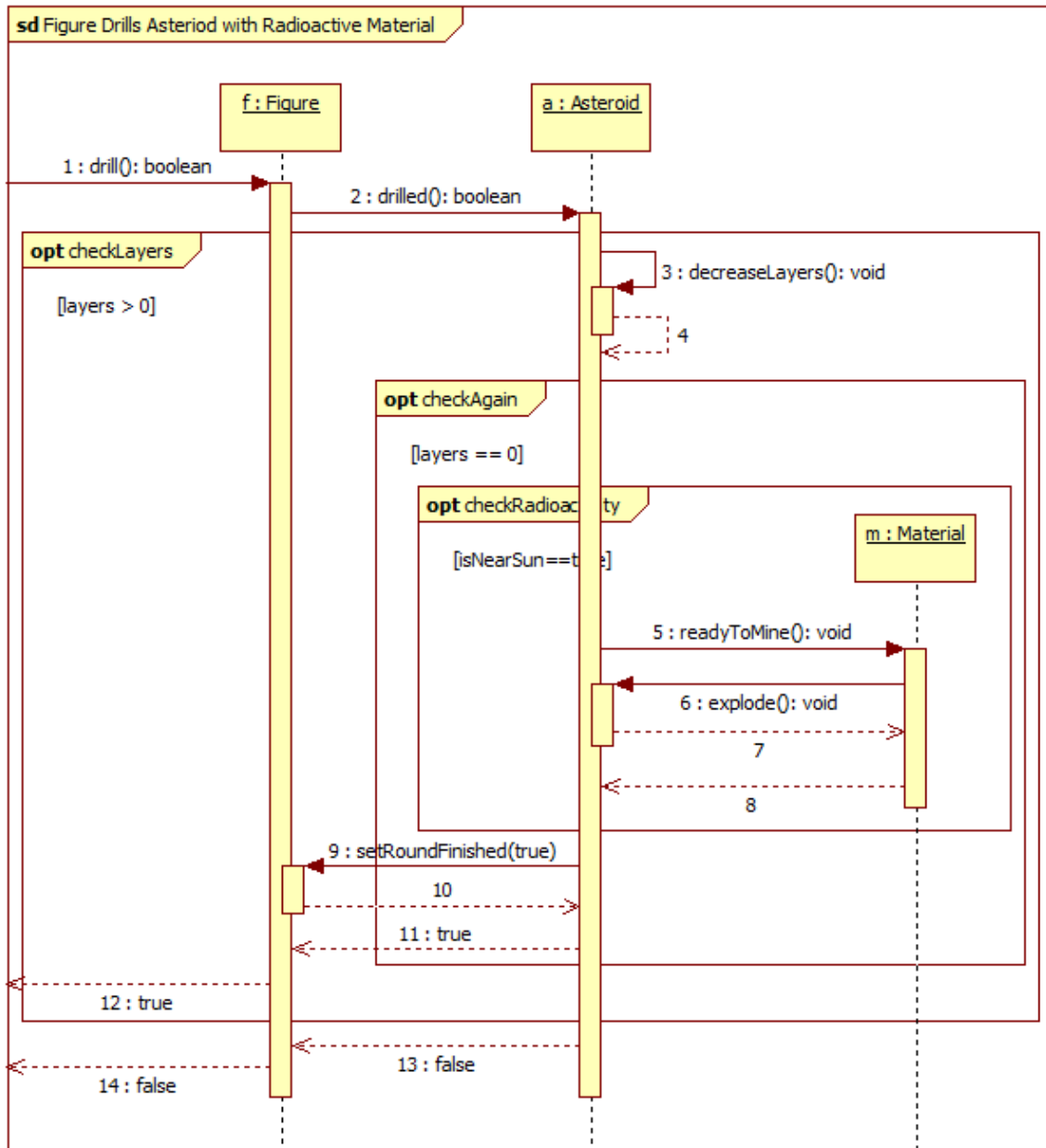
4.4.1 Move Settler

A telepes mozgását mutatja be.



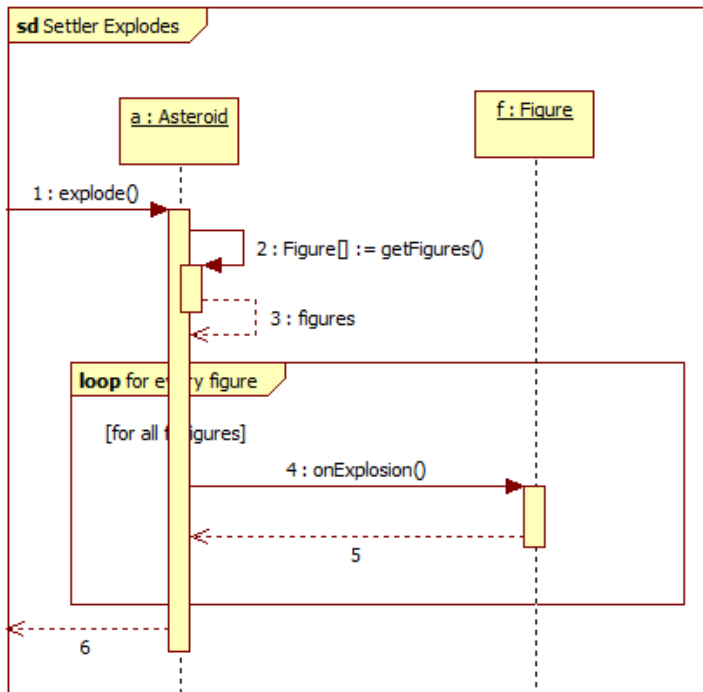
4.4.2 Figure Drills Asteroid with Radioactive Material

Azt az esetet mutatja be, amikor a Figure egy radioaktív anyaggal töltött aszteroida köpenyét fúrja.



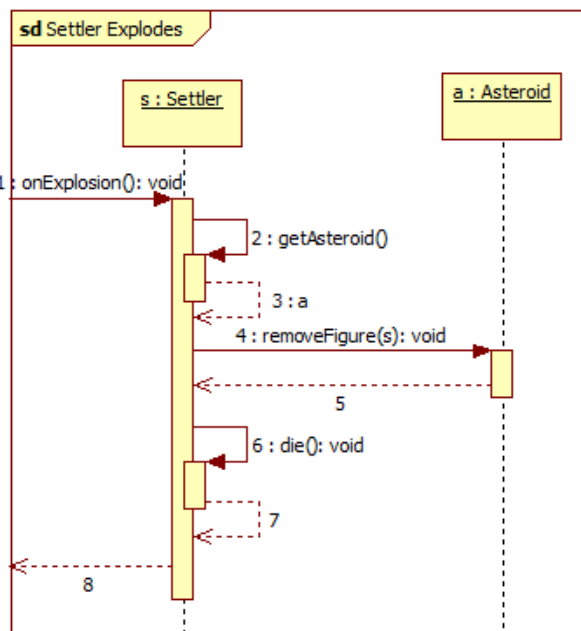
4.4.3 Figure Explodes

Azt az esetet mutatja be, amikor a figura kifúrja egy radioaktív magot tartalmazó aszteroida utolsó köpenyét napközben.



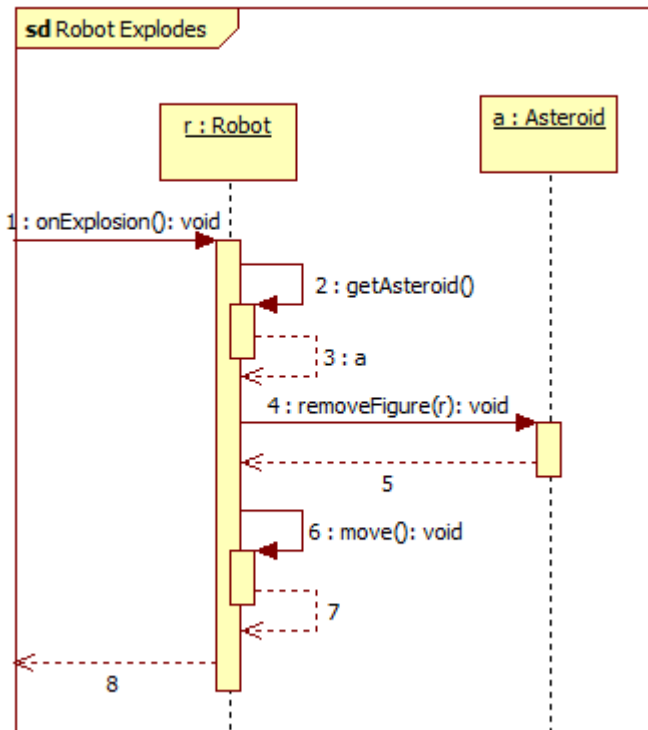
4.4.4 Settler Explodes

Azt az esetet mutatja be, amikor egy telepés kifúrja egy radioaktív magot tartalmazó aszteroida utolsó köpenyét napközben.



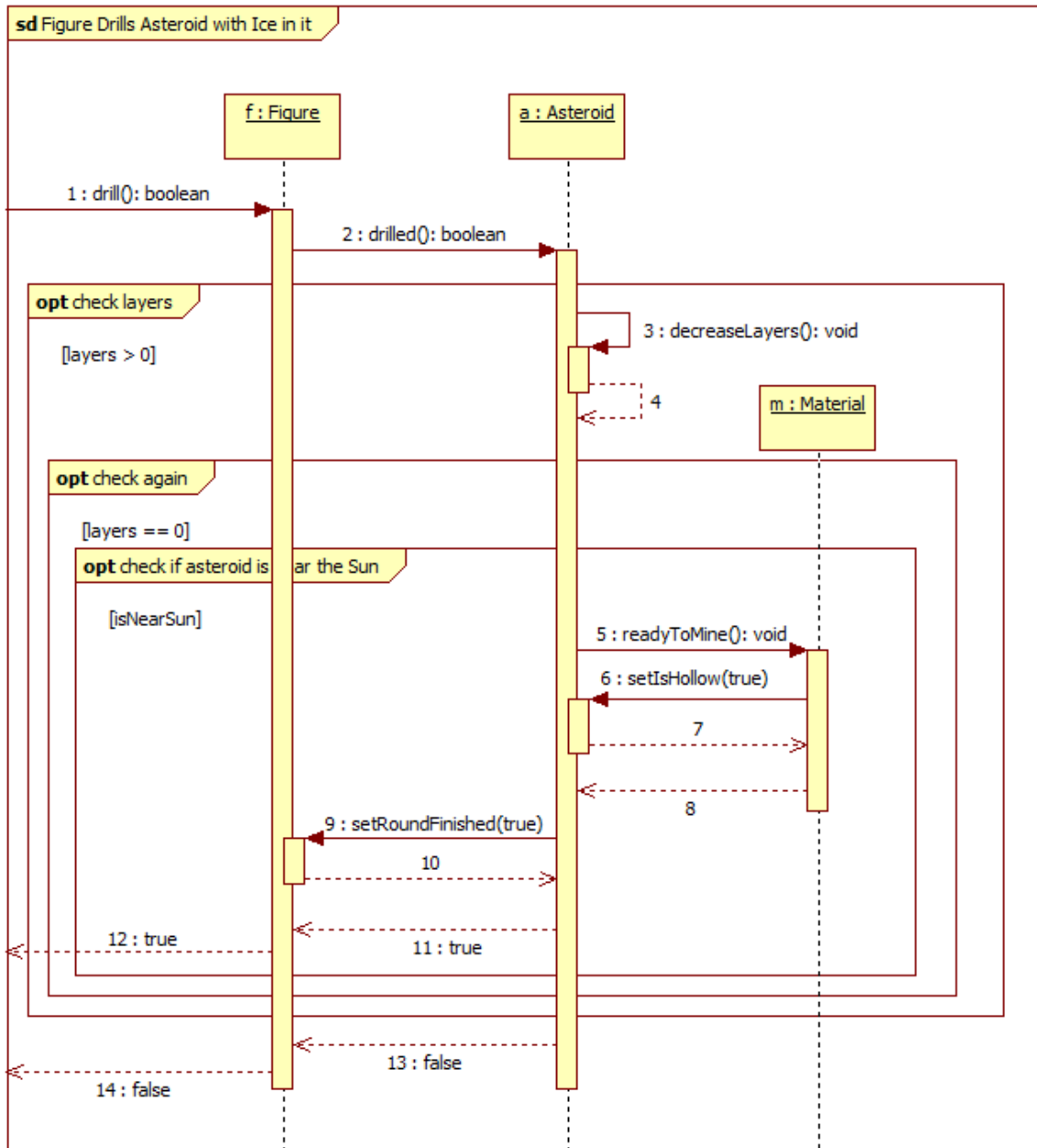
4.4.5 Robot Explodes

Azt az esetet mutatja be, amikor a robot kifúrja egy radioaktív magot tartalmazó aszteroida utolsó köpenyét napközeli.



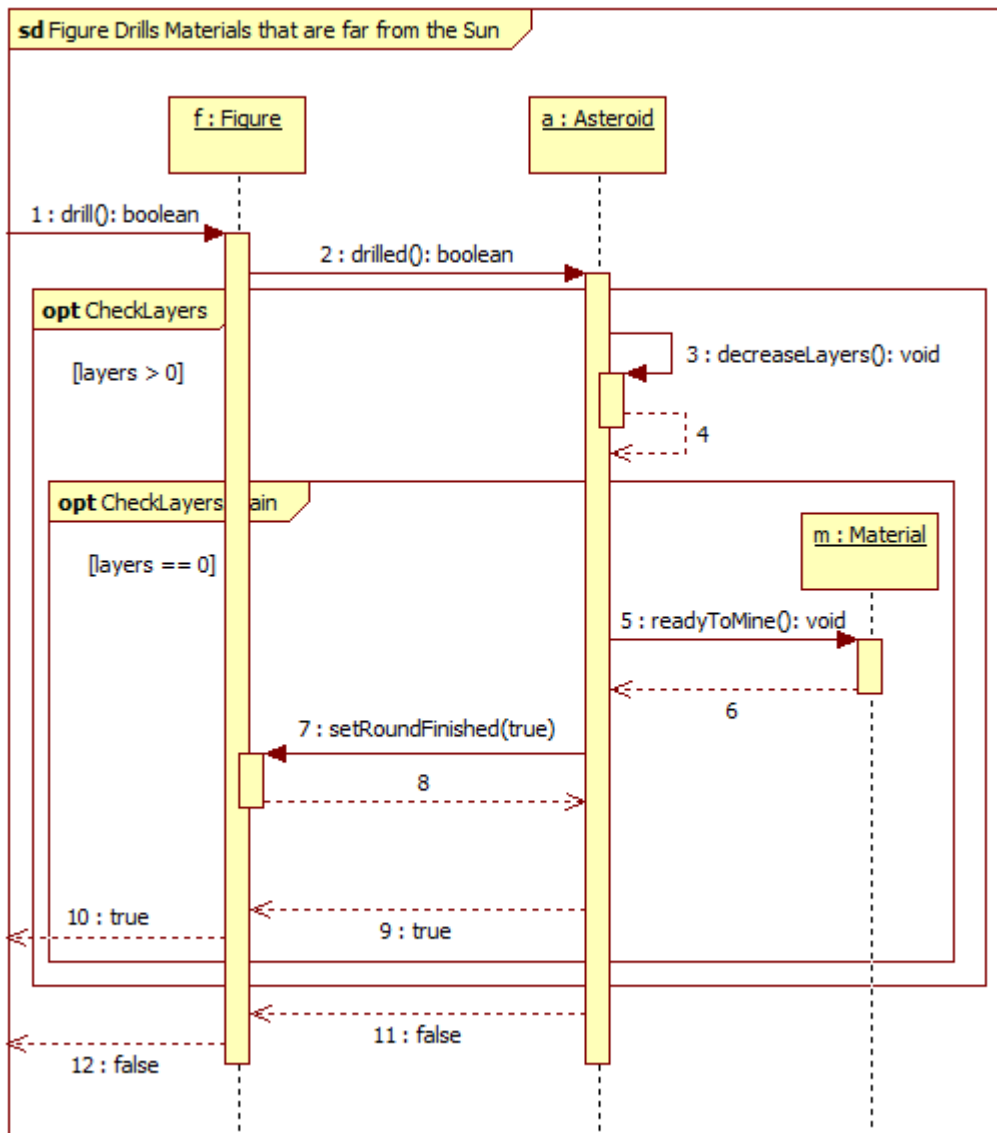
4.4.6 Figure Drills Asteroid with Ice in it

Azt az esetet mutatja be, amikor a Figure egy jéggel töltött aszteroida köpenyét fúrja.



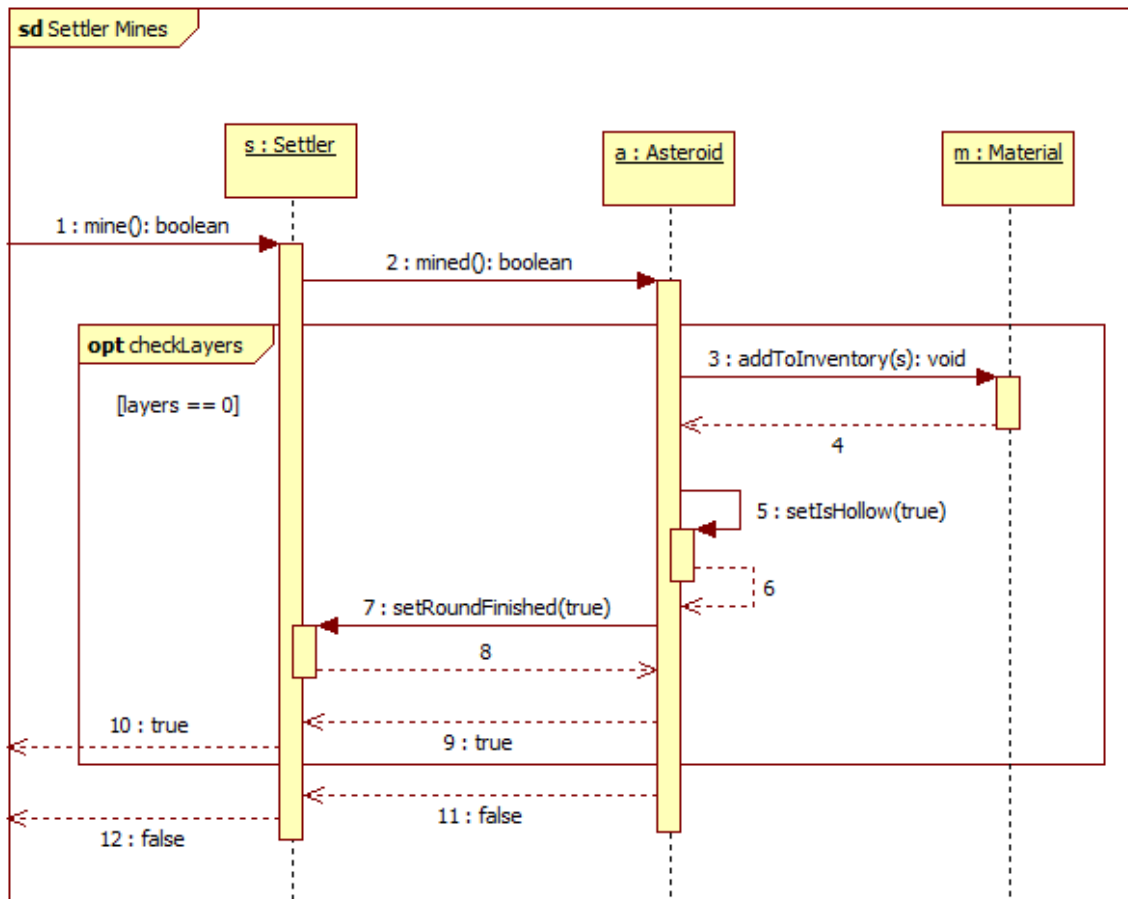
4.4.7 Figure Drills Asteroid with Common Materials in it (or special ones that are far from the Sun)

Azt az esetet mutatja be, amikor a figura egy közönséges nyersanyagot tartalmazó aszteroidát fúr meg (szén, vas) vagy speciálisat (jég, urán), amelyek nincsenek napközelben.



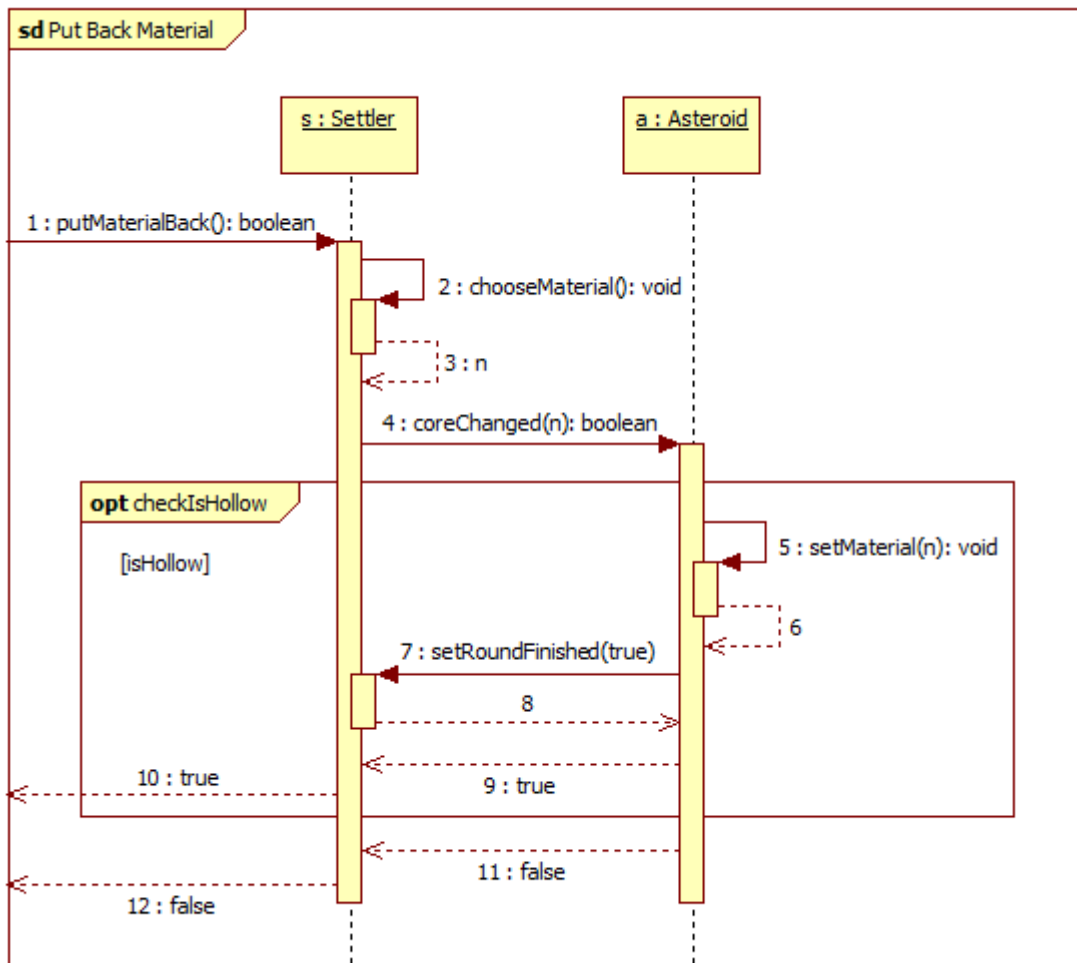
4.4.8 Settler Mines

Azt mutatja be, amikor a telepés kibányászik egy nyersanyagot az aszteroidából.



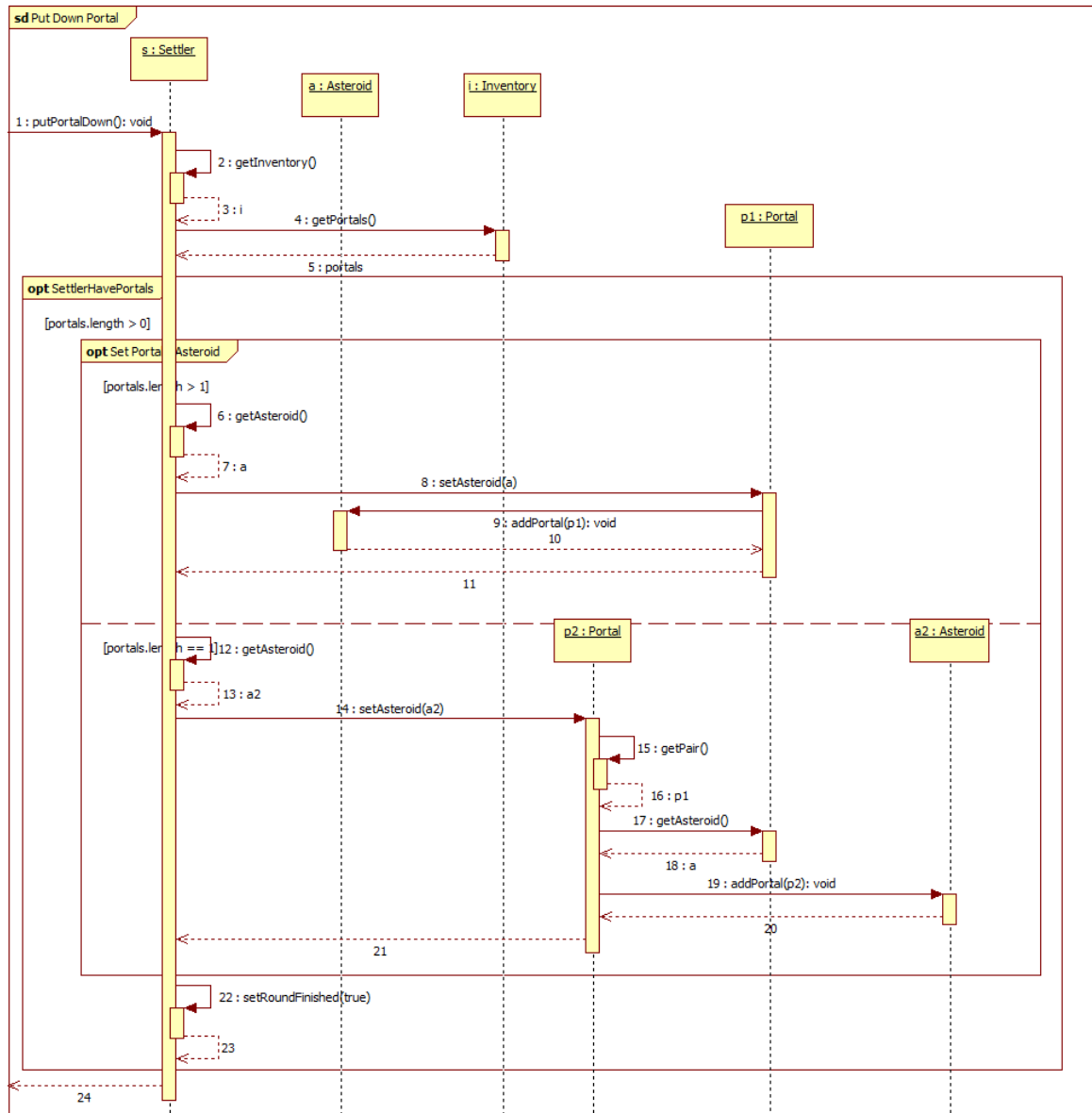
4.4.9 Put Material Back

Azt az esetet mutatja be, hogy a telepes visszatesz egy nyersanyagot az üres aszteroidába.



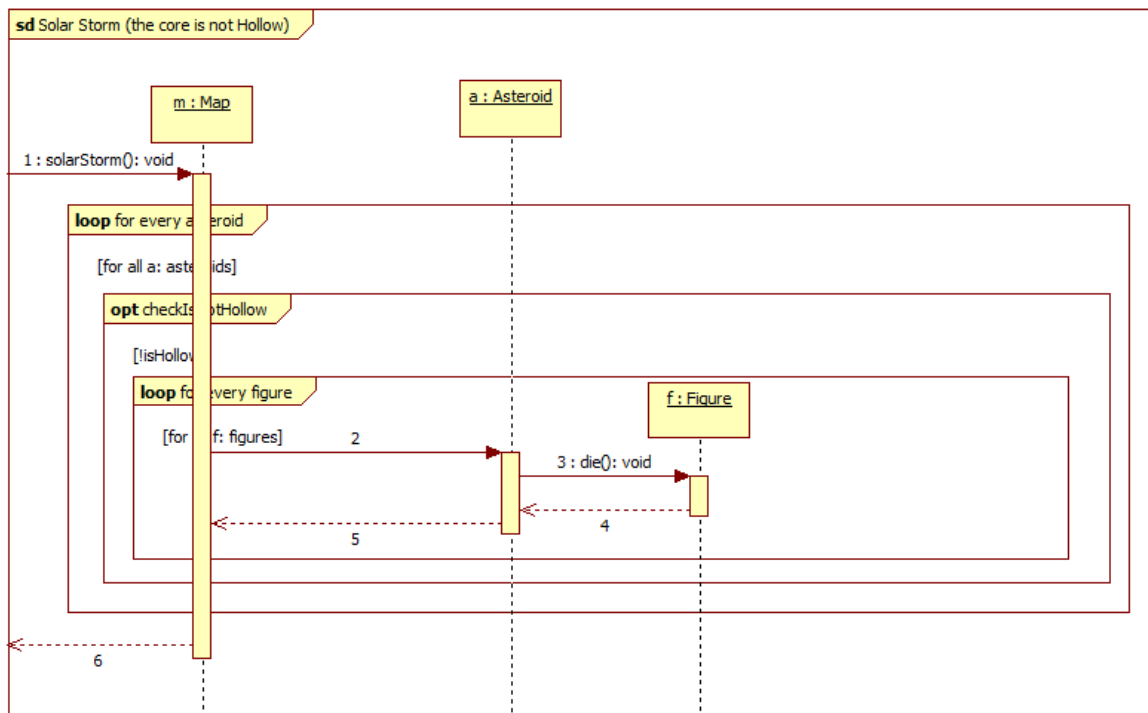
4.4.11 Put Portal Down

Azt az esetet mutatja be, hogy a telepés letesz egy teleportkapu-párt 2 különböző aszteroidára, amely hatására ezek az aszteroidák szomszédosak lesznek.



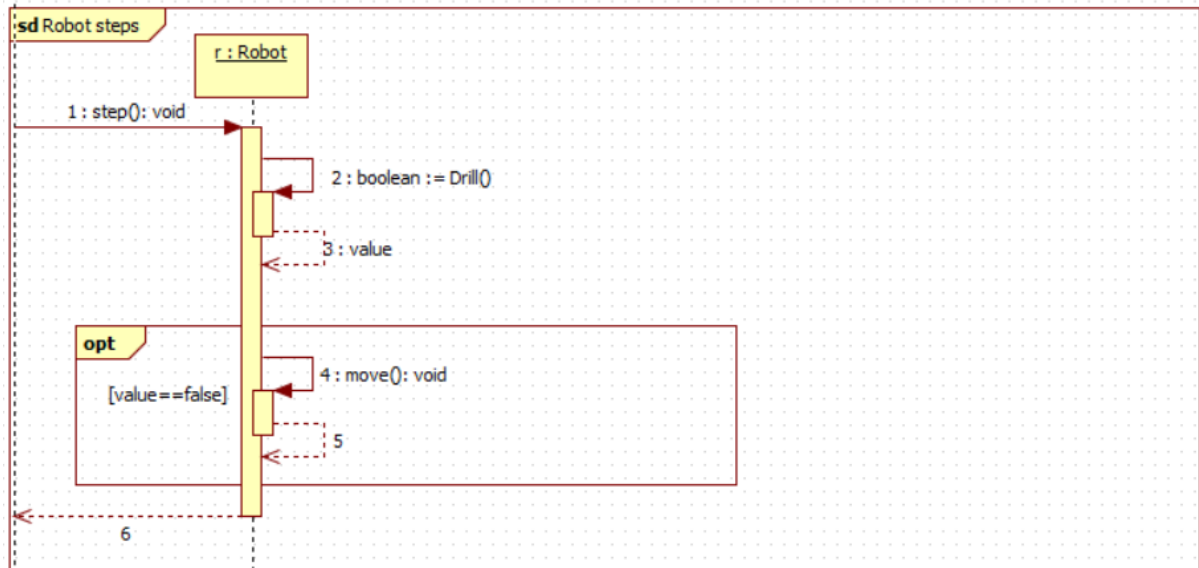
4.4.12 Solar Storm

A napvihar hatásait mutatja be azokra a figurákra, amelyek nem üreges aszteroidán állnak.



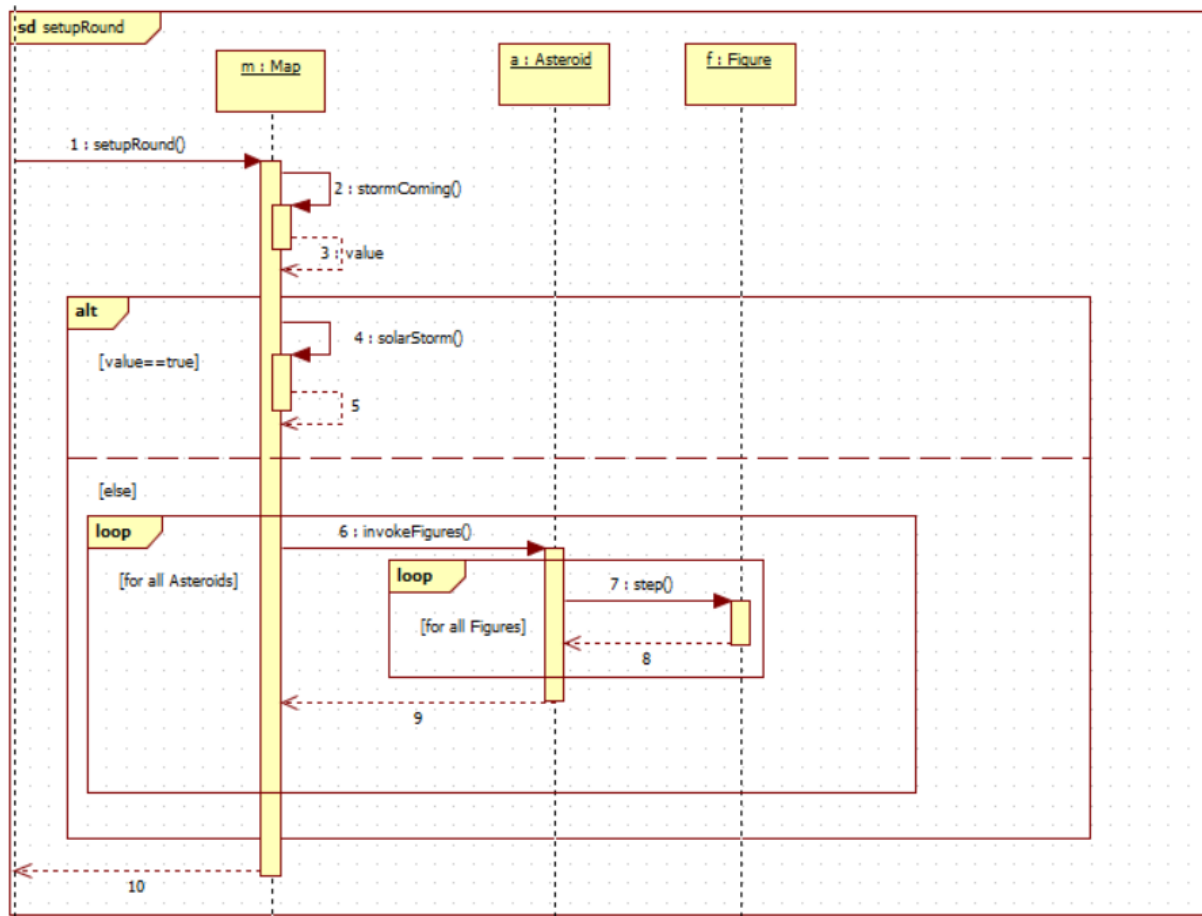
4.4.14 Robot steps

Adott körben a robot lépése: ha a jelenlegi aszteroidán még tud fújni, akkor fúr, ha nem, akkor átlép egy szomszédos aszteroidára.



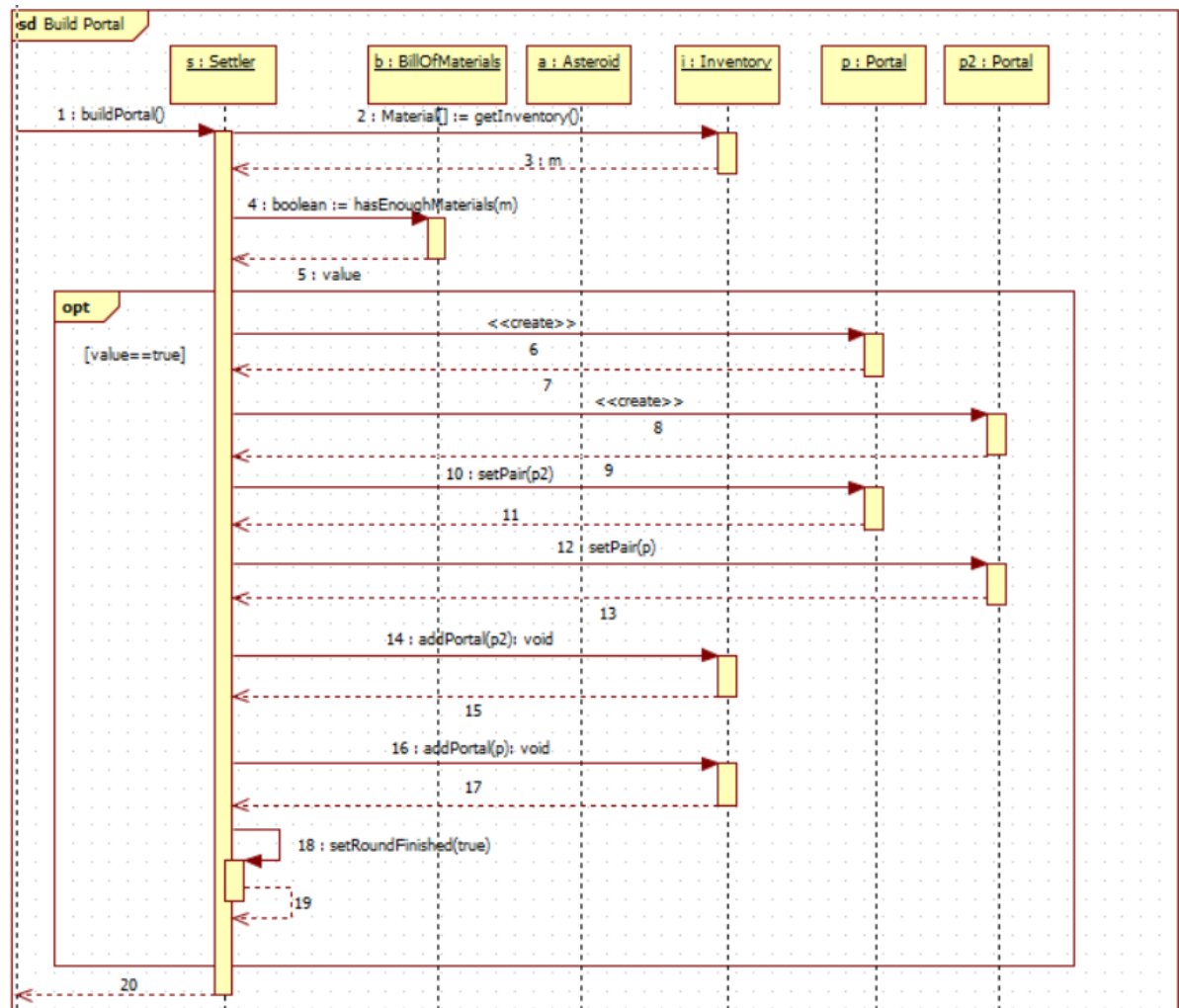
4.4.15 Setup round

Ez a függvény vezérli a köröket.



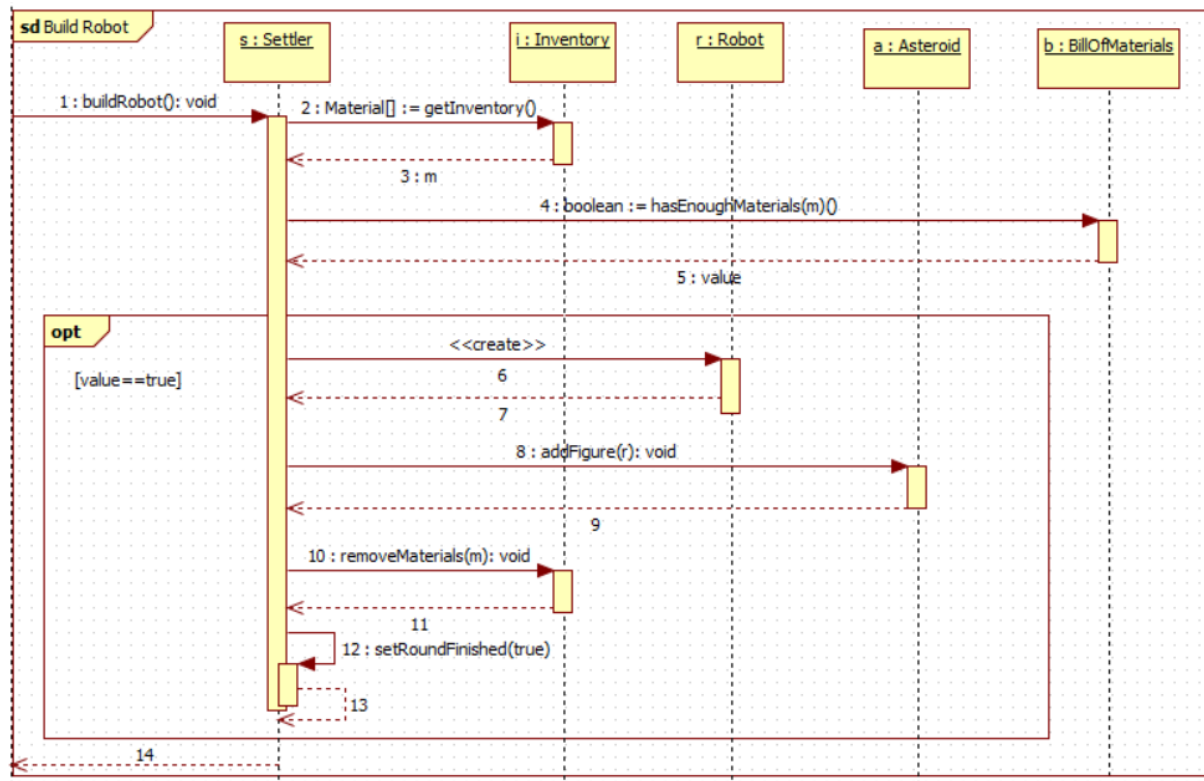
4.4.16 Build Portal

A teleportkapu-pár építését mutatja be.



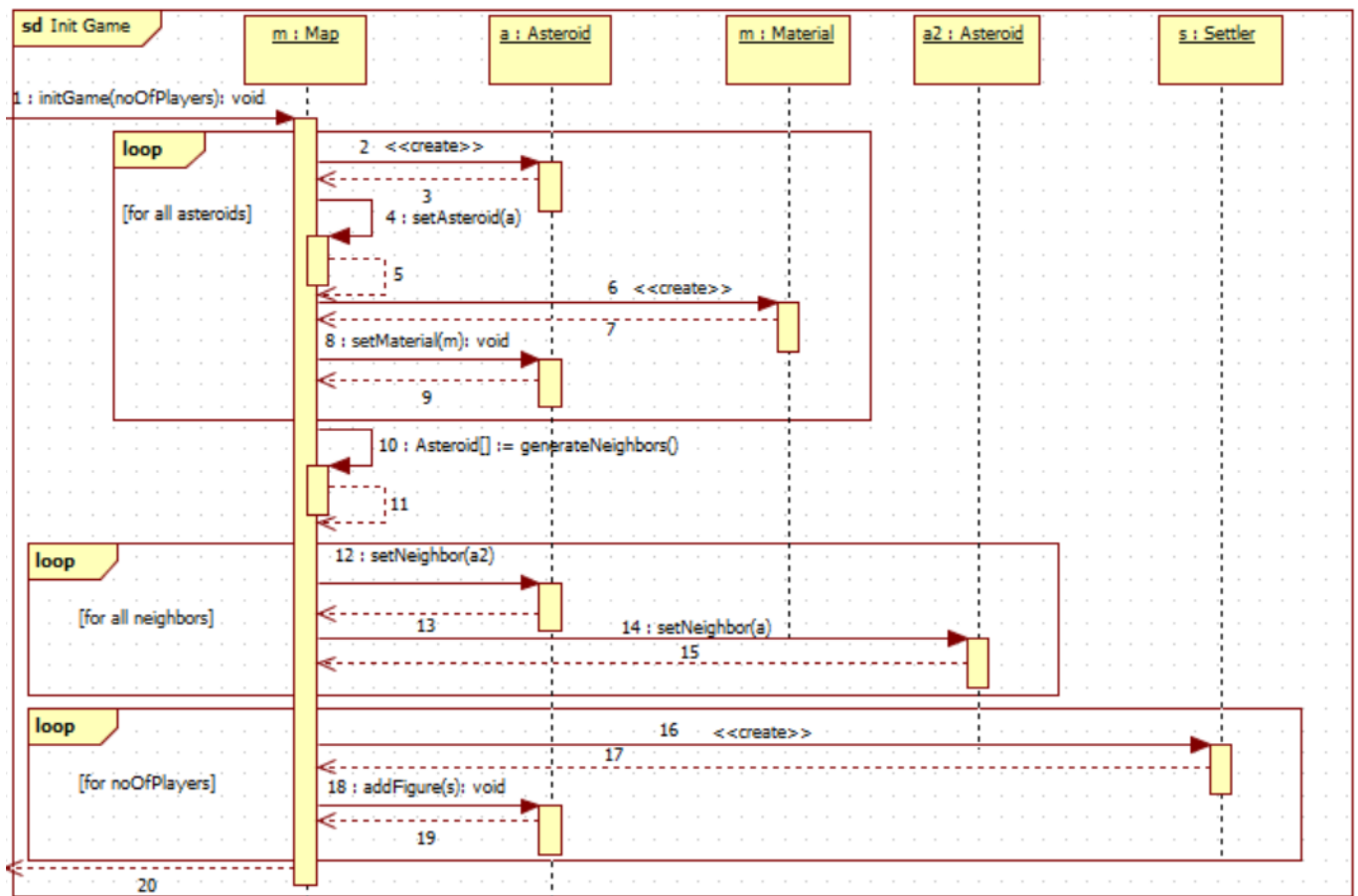
4.4.17 Build Robot

Egy robot építését mutatja be.



4.4.18 Init game

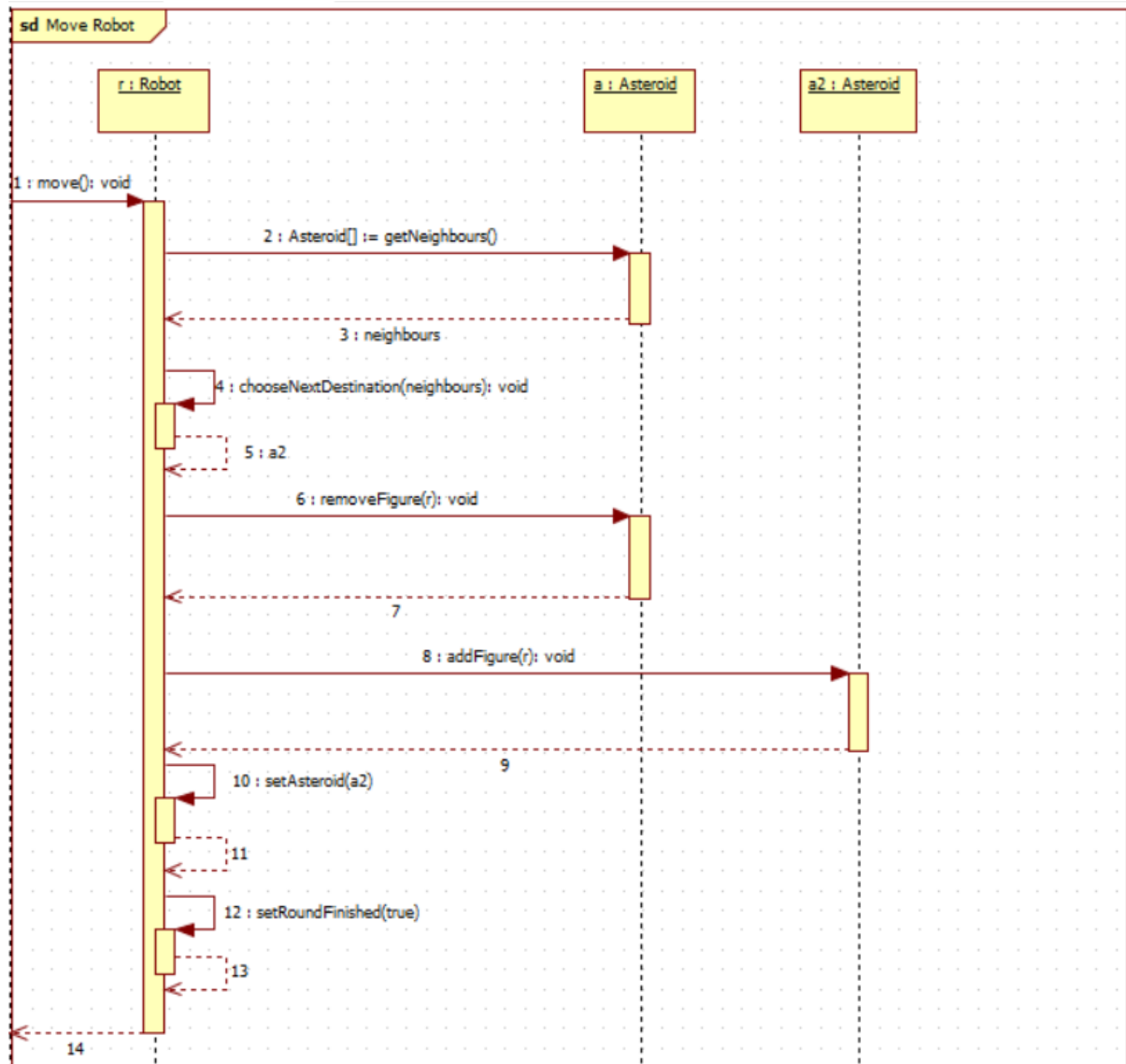
A játék inicializálása, játékosok és aszteroidák elkészítése.



- *Megjegyzés: A játékosok ugyanazon az aszteroidán kezdik meg a játékot.*

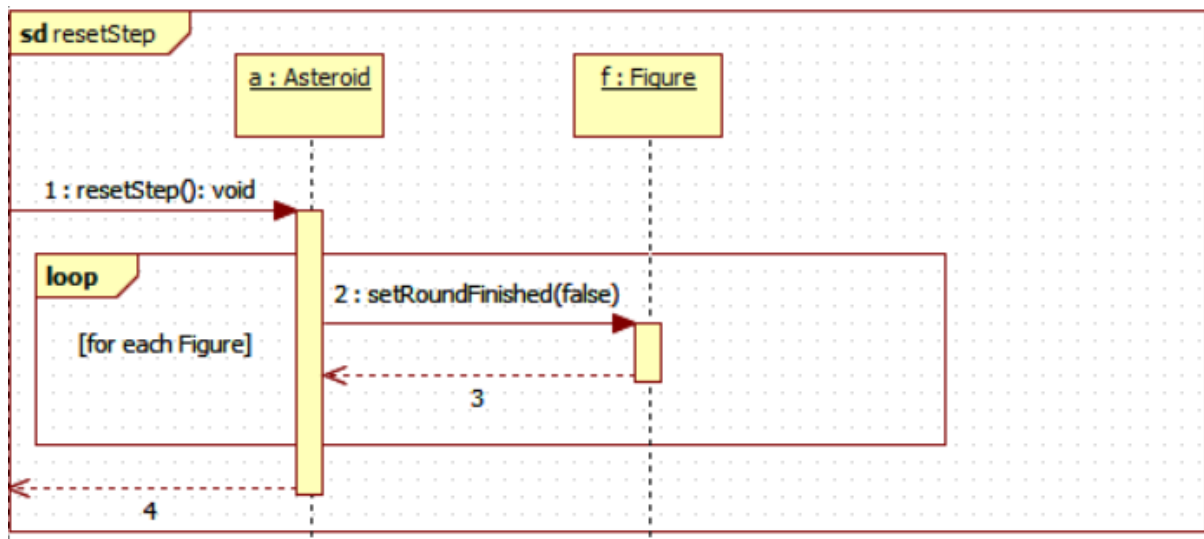
4.4.19 Robot moves

A robot mozgását mutatja be.



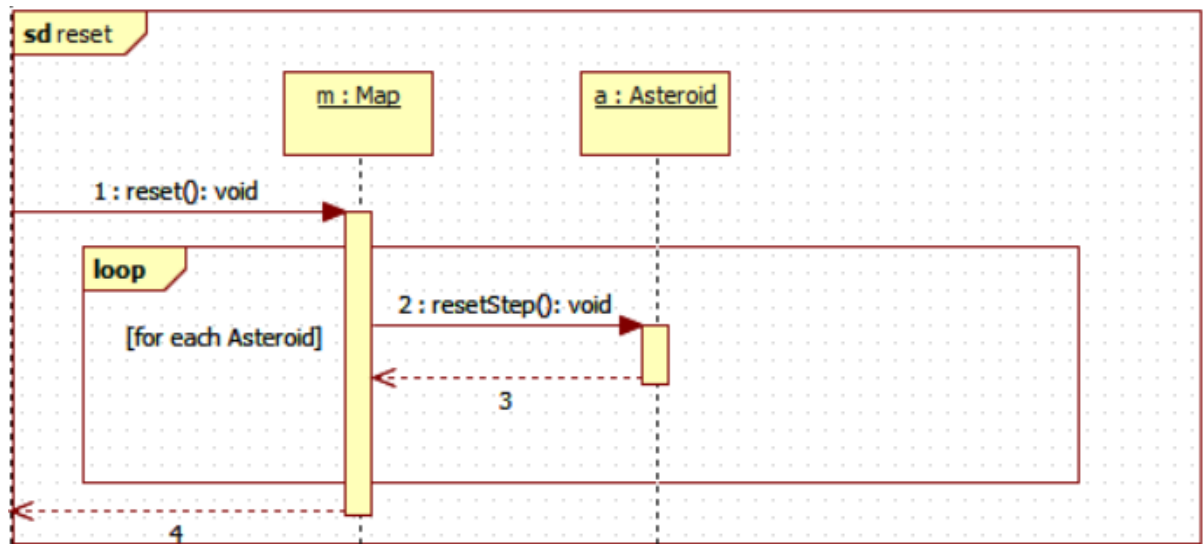
4.4.20 resetStep

Meghívja az Aszteroida az összes rajta lévő figurán a setRoundFinished függvényt ami, hamisra állítja, hogy egy karakter lépett-e már a körben.



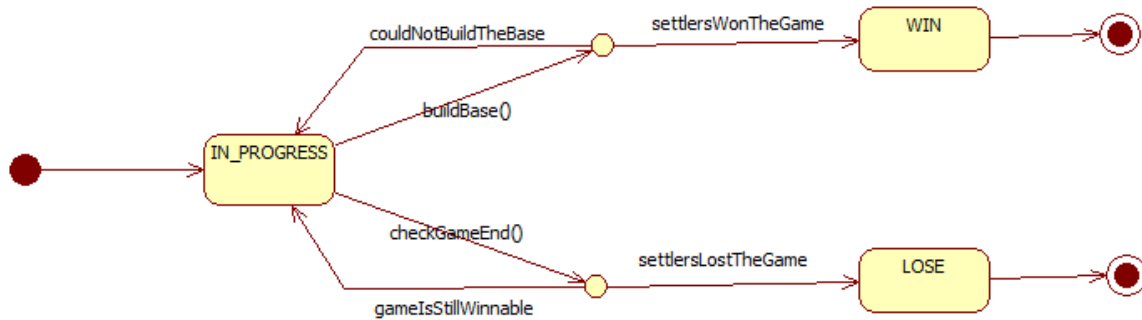
4.4.21 reset

Meghívja az összes játékban lévő Aszteroidán a reset függvényt ami meghívja a 4.4.20-as szekvencián szemléltetett függvényt.



4.5 State-chartok

4.5.1 Game end



4.6 Napló

Kezdet	Időtartam	Részrtvevők	Leírás
2021.03.03. 15:00	2 óra	Mindenki	Átbeszéltük, hogy mikben kellenek változtatások
2021.03.03. 17:00	1 óra	Halász Lauber Pintér	Szekvencia diagramok átírása
2021.03.04. 18:40	20 perc	Halász Hámori	Szekvencia diagramok javítása

2021.03.04. 19:00	1 óra	Bödőcs Halász Pintér	Szekvencia diagramok javítása, osztálydiagram javítása
2021.03.05. 14:30	2 óra	Bödőcs Halász Lauber Pintér	Szekvencia diagramok javítása State-chart diagram elkészítése
2021.03.05. 16:30	1 óra	Bödőcs Halász Pintér	Szekvencia diagramok frissítése és class diagram bővítése BillOfMaterials osztállyal.
2021.03.07. 22:00	1 óra	Hámori	Init Game szekvencia
2021.03.08. 10:00	0,5 óra	Bödőcs Halász Lauber Pintér	Dokumentum átnézése