

DOKUMENTÁCIÓ

67 – Brainstormers

Konzulens:
Simon Balázs

Csapattagok:

Mohácsi Márton

T8ZT88

moha.mohacsi@gmail.com

Simon Zoltán

HRSNUG

simonzolika1999@gmail.com

Kárpáti Márk András

O1BG0Z

mark.karpati1999@gmail.com

Bárkányi Csaba

BQI7QQ

csaba.barkanyi.csaba@gmail.com

Tóth Ádám

QEYOW2

toth99adam@gmail.com

2021-05-10

2. KÖVETELMÉNY, PROJEKT FUNKCIONALITÁS

67 – Brainstormers

Konzulens:
Simon Balázs

Csapattagok:

Mohácsi Márton

Simon Zoltán

Kárpáti Márk András

Bárkányi Csaba

Tóth Ádám

T8ZT88

HRSNUG

O1BG0Z

BQI7QQ

QEYOW2

moha.mohacsi@gmail.com

simonzolika1999@gmail.com

mark.karpati1999@gmail.com

csaba.barkanyi.csaba@gmail.com

toth99adam@gmail.com

2021-05-10

2. Követelmény, projekt, funkcionalitás

2.1 Bevezetés

2.1.1 Cél

Áttekintést nyújt a készülő szoftver felhasználási területeiről, korlátairól és jövőbeli felhasználóról. Részletezi és rendszerezi a követelményeket.

2.1.2 Szakterület

Szórakoztatóipari termék, egy stratégiai játék.

2.1.3 Definíciók, rövidítések

JDK: (Java Development Kit) A projekt fejlesztéséhez használt fejlesztőeszköz.

JRE: (Java Runtime Environment) A projekt futtatásához használt környezet.

Use-Case: A rendszer használatának egy lehetősége.

2.1.4 Hivatkozások

Ütemterv: <https://www.iit.bme.hu/targyak/BMEVIIIAB02/%C3%BCtemtervh%C3%A1t%C3%A1rid%C5%91k>

2.1.5 Összefoglalás

A dokumentum első fejezeteiben egy egyszerűbb áttekintést kapunk a játék felépítéséről és működéséről, valamint egyéb paraméterek is felvázolásra kerülnek. Ezek közé vehető a felhasználókra és a termékre vonatkozó megkötések és ajánlások.

A 2.3. fejezettől kezdve pedig az eddig vázlatosan felvett követelmények és funkciók kerülnek pontosításra, hogy azok jobban körül rajzolják magát a készülő terméket. A fejezet végén a use-case-ek vázolásával a fejlesztési munka látványosabb előrehaladása is megvalósul.

Az utolsó fejezetekben pedig a dokumentummal kapcsolatos egységek jelennek meg valamint a projekt jövőbeli előrehaladását körül rajzoló projektterv és az eddigi munkát bemutató napló kap helyet.

2.2 Áttekintés

2.2.1 Általános áttekintés

A szoftver architektúrája három fő részből fog állni. A modell fogja ábrázolni a játék komponenseit, ezek működését és az ezek közötti kapcsolatokat. A nézet felelős azért, hogy a játék komponensei megjelenjenek a felhasználó előtt. A kontroller rész pedig azért felel, hogy a felhasználó a modellben ábrázolt elemekkel kapcsolatba tudjon lépni.

A játékban egyszerre egy játékos vesz részt, így hálózati elvárásai nincsenek a szoftvernek.

2.2.2 Funkciók

A játékos a telepeseket irányítja egy aszteroidaövben. A telepesek az aszteroidákat járják megfelelő nyersanyagok után kutatva, hogy létrehozhassák a saját bázisukat. A munkájukat autonóm robotok is segítik.

Az aszteroidaöv részét képezi nap, amelyből lehet egy vagy akár több is, de a helyzetük minden esetben rögzített. Az az aszteroidák naphoz viszonyított helyzete, vagyis, hogy

napközelben vannak-e időnként változik a játék során. A napok véletlen időnként napkitörést idéznek elő, amely a saját aszteroidaövük végig futja.

Az öv másik részegységét, az aszteroidák különböző vastagságú közetrétegek borítják. A fontos nyersanyagok (akár például: vízjég, vas, szén, urán stb.) az aszteroidák magjában találhatóak. Vannak veszélyes aszteroidák, amelyek magjának anyaga erősen radioaktív, illetve üreges aszteroidák is, amelyek magjában nem található nyersanyag. Az aszteroidák magja nyersanyagok vannak, ha nem üreges.

A telepesek és a robotok egy körben csak egyfajta műveletet képesek végrehajtani. A telepes lehetséges műveletei közé tartozik például a mozgás, fúrás, bányászat, robotépítés. Mozgás során a telepes az űrhajójával átmegy egy szomszédos aszteroidára (minden aszteroidának néhány, vagy akár több száz szomszédja is lehet). Fúrás során a játékos a telepesét irányítva körönként egy egységnyivel tudja mélyíteni az aszteroida köpenyébe fúrt lyukat. Bányászat során a telepes kinyeri a már kifúrt aszteroida magjában lévő nyersanyagot, de ez a lépés csak akkor lehetséges, ha az aszteroida köpenyét már sikerült teljesen átfúrnia. A bányászat során a kitermelt aszteroida magja üregessé válik. Kiemelendő, hogy a telepes a már nála lévő nyersanyagokat visszahelyezheti egy üreges aszteroida magjába, de természetesen egy körben csak egy egységnyit. A telepes nehéz felszerelése miatt csak limitált számú nyersanyag szállítására képes így, ha már előzőleg magához vett maximális számú nyersanyagot, akkor a bányászat nem lehetséges.

A telepesek képesek teleportkapukat építeni megkönnyítve ezzel az aszteroidaöv felfedezését és kitermelését. A telepes az utazása során kinyert nyersanyagokat felhasználva képes a kapukat előállítani, amelyekből egyszerre három darab fér el nála. A kapu építéshez szükséges anyagok lehetnek például az urán és a vízjég. Egy kapu csak egy másikkal lehet összeköttetésben, viszont egy aszteroidán több is elhelyezhető, így nagyobb hálózat is könnyen felépíthető belőlük. A teleportkapu csak akkor lép működésbe, ha a párja is lehelyezésre került, ami után a telepes egyik kapun belépve a másik kapuban találja magát.

A telepesek robotok építéséhez bizonyos nyersanyagokkal kell rendelkezni. Ezen nyersanyagok lehetnek például a vas, szén és urán is. A telepes egy lépésben csak egy robotot tud elkészíteni, amely elkészítése után azonnal üzembe is áll. A robotok szomszédos aszteroidák közötti mozgásra és fúrásra képesek. Ezen kívül képesek teleportkapukon keresztül távolabbi aszteroidákat is elérni. A robotok bányászni nem tudnak, mert nyersanyagot nem tudnak magukkal vinni. A robotokra továbbra is fennál, hogy önállóságuk ellenére is egy lépésben csak egyfajta műveletet tudnak ellátni. Ezen művelet lehet az aszteroidák közötti mozgás vagy egy aszteroidának a magjáig való lefúrás, hogy ezzel könnyítse a telepesek munkáját.

Ha egy telepes egy napközelben lévő aszteroida magját átfürja, és a mag tartalma vízjég, akkor a vízjég szublimál. Ha egy telepes vagy robot egy radioaktív aszteroida magjába fúr, és az aszteroida éppen napközelben van, akkor az aszteroida három kör műlva felrobban. A robbanás következtében a telepes meghal a robot pedig egy szomszédos aszteroidán landol. A radioaktív anyag tehát csak naptávolban lévő aszteroidából nyerhető ki. Az aszteroida megsemmisülése maga után vonja az aszteroidán lévő kapu és pájrának is a megsemmisülését. További veszélyt jelentenek a telepesekre és a robotokra a napviharok, amelyek időnként elérik az aszteroidaövet. A napvihar csak úgy élhető túl, ha a telepes vagy robot egy kifúrt és üreges aszteroidán áll, mivel ekkor felmérve az eseményeket magukat mentve gyorsan elbújnak a napvihar idejére. A teleportkapuk működésükönél adódóan ellenállnak a napviharoknak.

A játékos számára játék kétféleképpen érhet véget. Ha minden telepes meghalt, akkor a játékos veszített. Ha azonban sikerült minden fajta nyersanyagból legalább három egységet

kibányászna és egy közös aszteroidára összegyűjteni azokat, akkor a telepesek felépíthetik a bázist és megnyerik a játékot.

2.2.3 Felhasználók

A készülő szoftver használatának nincsen életkorbeli megkötése. Annyit feltételez egyedül a felhasználóról, hogy tudnak olvasni, illetve minimális szinten tudják kezelní a számítógépet.

2.2.4 Korlátozások

A szoftver futtatásához Java 8+ JRE környezet telepítésére van szükség, egyéb korlátozás, megkötés nincs a környezetre vonatkozóan.

2.2.5 Feltételezések, kapcsolatok

A hivatkozásokban szereplő weboldal a projekt ütemtervét határozza meg.

2.3 Követelmények

2.3.1 Funkcionális követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Use-case	Komment
R01	A játékos lép a telepessel a pályán, valamely szomszédos aszteroidára.	Bemutatás, vagy kiértékelés.	alapvető	Megrendelő leírása.	mozgás	
R02	A játékos fúr egy telepessel egy aszteroida felszínén.	Bemutatás, vagy kiértékelés.	alapvető	Megrendelő leírása.	fúrás	
R03	A játékos kibányászik egy nyersanyagot egy telepessel egy aszteroidából.	Bemutatás, vagy kiértékelés.	alapvető	Megrendelő leírása.	bányászás	
R04	A fúrás során a sziklaréteg vastagsága csökken.	Bemutatás, vagy kiértékelés.	alapvető	Megrendelő leírása.	fúrás	
R05	A játékos összerak egy robotot.	Bemutatás, vagy kiértékelés.	fontos	Megrendelő leírása.	építés	
R06	A játékos teleportkapup árt épít.	Bemutatás, vagy kiértékelés.	fontos	Megrendelő leírása.	építés	

R07	A játékos elhelyez egy teleportkaput egy aszteroida felszínén.	Bemutatás, vagy kiértékelés.	fontos	Megrendelő leírása.	kapu lerakás	
R08	A játékos visszahelyez egy nyersanyagot egy üreges aszteroidába.	Bemutatás, vagy kiértékelés.	fontos	Megrendelő leírása.	visszahelyezés	
R09	A játékos napközelben lévő, radioaktív aszteroidát fűr, ami felrobban.	Bemutatás, vagy kiértékelés.	alapvető	Megrendelő leírása.	fúrás	
R10	A telepes meghal.	Bemutatás, vagy kiértékelés.	alapvető	Megrendelő leírása.	napvihar, fúrás	
R11	A telepes bázist épít, így megnyeri a játékot.	Bemutatás, vagy kiértékelés.	alapvető	Megrendelő leírása.	építés	
R12	A robot önállóan lép, valamely szomszédos aszteroidára.	Bemutatás, vagy kiértékelés.	fontos	Megrendelő leírása.	autonóm cselekvés	
R13	A robot fűr.	Bemutatás, vagy kiértékelés.	fontos	Megrendelő leírása.	autonóm cselekvés	
R14	A robot meghal.	Bemutatás, vagy kiértékelés.	fontos	Megrendelő leírása.	autonóm cselekvés	
R15	Napvihar történik.	Bemutatás, vagy kiértékelés.	fontos	Megrendelő leírása.	napvihar	
R16	Az aszteroida felrobban.	Bemutatás, vagy kiértékelés.	fontos	Megrendelő leírása.	fúrás	
R17	Az aszteroida magjában lehet egy nyersanyag.	Bemutatás, vagy kiértékelés.	alapvető	Megrendelő leírása.	bányászás	
R18	Egy nyersanyag lehet radioaktív	Bemutatás, vagy kiértékelés.	alapvető	Megrendelő leírása.	bányászás	

R19	Ha az összes telepes meghal, a játékos veszít.	Bemutatás, vagy kiértékelés.	alapvető	Megrendelő leírása.	fúrás	
-----	--	------------------------------	----------	---------------------	-------	--

2.3.2 Erőforrásokkal kapcsolatos követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
E01	A szoftver fejlesztéséhez Java JDK 8+-ra van szükség.	Kiértékelés	alapvető	csapat	
E02	A szoftver fejlesztéséhez, verziókezeléshez Github fiókra van szükség.	Kiértékelés	alapvető	csapat	
E03	A szoftver használatához Java JRE 8+-ra van szükség.	Kiértékelés	alapvető	csapat	
E04	A szoftver használatához olyan operációs rendszerre van szükség, amin Java futtatható.	Kiértékelés	alapvető	csapat	

2.3.3 Átadással kapcsolatos követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
P01	A szoftvert akkor tekintjük késznek, ha az alapvető funkcionális követelmények teljesülnek.	Bemutatás és kiértékelés	alapvető	csapat	
P02	A szoftvert akkor tekintjük késznek, ha a hozzá tartozó dokumentáció is elkészült.	Bemutatás és kiértékelés	alapvető	csapat	
P03	A szoftvert akkor tekintjük késznek, ha a hozzá tartozó telepítési és felhasználói	Bemutatás és kiértékelés	alapvető	csapat	

	útmutató is elkészült.				
--	---------------------------	--	--	--	--

2.3.4 Egyéb nem funkcionális követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
O01	A szoftvertől elvárjuk, hogy könnyen kezelhető legyen.	Bemutatás	fontos	csapat	
O02	A szoftvertől elvárjuk, hogy ne szaggasson, folyékonyan, zavartalanul lehessen használni.	Bemutatás	fontos	csapat	

2.4.1 Use-case leírások

Use-case neve	mozgás
Rövid leírás	Egy entitás az aktuális tartózkodási helyéről egy másik, szomszédos aszteroidára kerül.
Aktorok	Felhasználó Kontroller
Forgatókönyv	1a. Az entitás megváltoztatja a pozíóját egy szomszédos aszteroidára. 1b. Olyan teleportkaput használva, amelynek a párja is el van helyezve egy másik aszteroidán, az entitás a másik kapu aszteroidájára kerül.

Use-case neve	fűrás
Rövid leírás	Egy entitás fűrást végez az aszteroidán.
Aktorok	Felhasználó Kontroller
Forgatókönyv	1. Az entitás az aszteroida köpenyében egy egységgel mélyíti a lyukat. 2a. Ha átfürta a köpenyt, akkor a mag tartalma elérhetővé válik bányászáshoz. 2b. Ha az aszteroida tartalma radioaktív és az aszteroida napközelben tartózkodik, a nyersanyag belső számlálója nől. 2. 2b.2 Ha a számláló elérte a 3-at, akkor felrobban. 3b. A felrobbant aszteroidán tartózkodott telepesek megsemmisülnek, a robotok a felrobbant aszteroidával szomszédos aszteroidára repülnek. Az aszteroidán lévő teleportkapuk és párjaik megsemmisülnek.

	4b. Ha minden telepes meghalt, akkor a telepesek veszítenek.
--	--

Use-case neve	bányászás
Rövid leírás	Egy bányászára képes entitás bányászik.
Aktorok	Felhasználó, Kontroller
Forgatókönyv	1. Ha az entitás tárolójában kevesebb, mint 10 egység nyersanyag van, az aszteroida köpenye át van fűrva és a magjában van nyersanyag, akkor az entitás a magban lévő nyersanyagból egy egységet a saját tárolójába rak.

Use-case neve	visszahelyezés
Rövid leírás	Aszteroida magjába visszahelyezhető nyersanyag. Így felszabadítható a tároló kapacitás valamelyen más nyersanyag típus számára.
Aktorok	Felhasználó
Forgatókönyv	1a. Ha a telepes átfúrt köpenyű, üreges aszteroidán tartózkodik, akkor a saját tárolójában lévő nyersanyagok közül kiválaszthat egy egységnyit visszahelyezésre. 1b. Ha a telepes olyan átfúrt köpenyű aszteroidán tartózkodik, amely magjában található nyersanyag, de fér még, akkor a saját tárolójában lévő nyersanyagok közül kiválaszthat egy egységnyit visszahelyezésre. 2. A kiválasztott nyersanyag visszahelyeződi a tárolóból az aszteroida magjába.

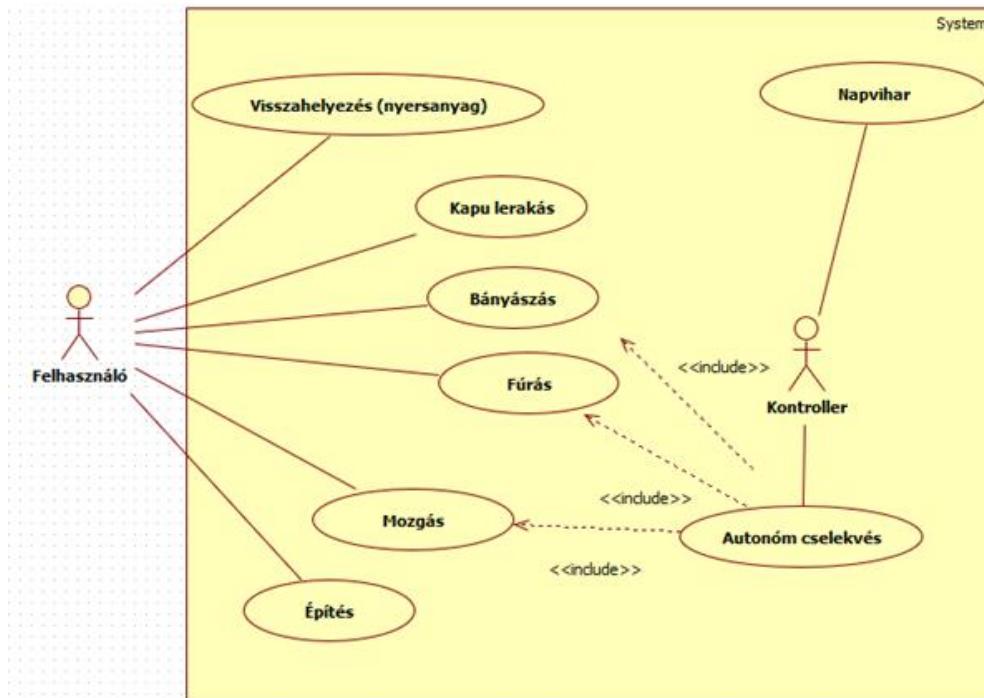
Use-case neve	építés
Rövid leírás	A telepes a megfelelő nyersanyagok felhasználásával különböző segédeszközöket, gépeket tud építeni.
Aktorok	Felhasználó
Forgatókönyv	1a. Ha egy telepes tárolójában van legalább egy egységnyi vas, szén és urán, akkor ezen nyersanyagokból egy-egy egységet elhasználva létrehozhat egy robotot. 2a. A robot azon az aszteroidán helyeződik el, amelyen a telepes tartózkodott, amikor létrehozta. 1b. Ha egy telepes tárolójában van legalább egy egységnyi vízjég és urán, illetve nincs nála teleportkapu, akkor a felsorolt nyersanyagokból egy-egy egységet elhasználva létrehozhat egy teleportkapu-párt. Az egyszerre létrehozott két kapu összeköttetésben lesz, amikor a pár minden eleme el lesz helyezve egy-egy aszteroidán. 1c. Ha egy aszteroidán egyszerre tartózkodó telepesek tárolójában együttesen megtalálható az összes nyersanyagtípusból típusonként legalább 3 egységnyi, akkor a telepesek megépíthetik a bázist. Ezzel megnyerik a játékot.

Use-case neve	kapu lerakás
Rövid leírás	A telepesek az aszteroidákon teleportkapukat tudnak lerakni.
Aktorok	Felhasználó
Forgatókönyv	1. Ha egy telepés tárolójában van legalább egy teleportkapu, akkor azt le tudja rakni az aktuális tartózkodási helyén (aszteroidán). Ha egy kapupár mindenként eleme el van helyezve egy-egy aszteroidán, akkor az egyik kapuba lépve a másik kapu aszteroidájára kerül a telepés vagy robot.

Use-case neve	autonóm cselekvés
Rövid leírás	Egy nem játékos által vezérelt entitás autonóm módon dönt néhány cselekvés végrehajtásáról. Ezekből körönként egyet tud elvégezni a felhasználó beavatkozása nélkül.
Aktorok	Kontroller
Forgatókönyv	1a. Entitás fúrás. 1b. Entitás mozgás. 1c. Ha egyiket sem tudja megtenni, akkor nem csinál semmit.

Use-case neve	napvihar
Rövid leírás	Időnként napvihar figyelhető meg. minden pillanatban meg lehet állapítani, hogy éppen van-e napvihar.
Aktorok	Kontroller
Forgatókönyv	1a. Ha napvihar alakul ki, a nem átfűrt köpenyű aszteroidán vagy az aszteroidában nem üreges, akkor az aszteroidán tartózkodó entitások megsemmisülnek. 2a. Ha minden telepés meghalt, akkor a telepesek veszítenek. 1b. Ha napvihar közben a telepesek vagy robotok átfűrt köpenyű aszteroidán tartózkodnak, ami üreges, akkor megmenekülnek a napvihartól.

2.4.2 Use-case diagram



2.5 Szótár

aktív teleportkapu	Egy teleportkapu akkor aktív, ha el van helyezve egy aszteroidán (tehát nincsen elpusztítva, illetve nem egy telepénél van).
aszteroida	A játéktér egy mezője, melyen az entitások tartózkodhatnak. Van burkolata, illetve a belseje melyben nyersanyag lehet.
aszteroidaöv	A teljes játéktér mely napokból s aszteroidákból áll.
bányászat	Az a folyamat, amivel az adott aszteroida belsejéből kitermeljük az adott nyersanyagot
bázis	Egy épület melynek felépítésével megnyerjük a játékot.
elhasznál	A nyersanyag beépül valamibe (egy robotba, teleportkapuba stb.) s ezáltal elhasználódik, megsemmisül.
entitás	A robotokat s telepeseket egybefogó gyűjtőfogalom az entitás.
építés	Egy folyamat mely során a megfelelő nyersanyagok elhasználásával egy telepes létrehoz egy robotot/teleportkaput/bázist.
épület	A teleportkapukat s bázist egybefoglaló gyűjtőfogalom az épület.
folyamat	A játék egy adott körében egy adott entitás lépése.
fúrás	Az a folyamat, amivel egy entitás az adott aszteroida kérgét vékonyítja.
mozog	Egy entitás mozgás során az egyik aszteroidáról egy másokra megy át.
nap	Egy olyan játéktérben jelenlévő elem mely meghatározza, hogy mely aszteroidák vannak napközelben.
napkitörés	Egy random esemény mely egy adott naphoz tartozó aszteroidákra hat azokkal tulajdonságának megfelelően interaktál.
robot	Egy a játékos által nem vezérelt entitás mely mozgásra s fúrásra képes.
telepes	Egy a játékos által vezérelt entitás mely mozgásra, fúrásra, bányászásra, nyersanyagok/teleportkapuk hordozására s épületek/robotok építésére képes.
teleportkapu	Egy épület melynek van egy párja, az épületen keresztülhaladva a párájában találjuk magunkat, ha az aktív.
üreges aszteroida	Olyan aszteroida, amely magában nincsen nyersanyag.
radioaktív	Olyan speciális nyersanyag, amely napközelben 3 kör után felrobban.
autonóm	A robotok a játékos beavatkozása nélkül mozognak az aszteroidák között és bányásszak azokat.
nyersanyag	Az aszteroidákból kinyerhető anyagok, amely kulcsfontosságúak a játékban való előrehaladásban.
szublimál	Az anyag napközelbeli átfűrt aszteroidán elpárolog.

2.6 Projekt terv

Határidő	Feladat
2021.02.22.	Projekt követelményeinek funkcionalitásainak, Use-case-einek elkészítése.
2021.03.01.	Analízis modell felének elkészítése.
2021.03.08.	Analízis modell második felének elkészítése.
2021.03.16.	A programváz tervének elkészítése.
2021.03.22.	A program vázának véglegesítése.
2021.03.29.	A prototípus koncepciójának elkészítése.
2021.04.12.	A szoftver részletes terveinek elkészítése.
2021.04.19.	A szoftver prototípusának véglegesítése.
2021.04.26.	A grafikus felület specifikációja.
2021.05.03.	A grafikus változat elkészítése.
2021.05.10.	A grafikus változat véglegesítése és a projekt összefoglalása.

Mindegyik feladatban a csapat összes tagja részt vesz. A projekthez választott fejlesztői környezet az Eclipse és a csoportmunkát, verziókezelést Github segítségével valósítjuk meg.

2.7 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2021.02.13. Git megbeszélés 16:00-17:30	1,5 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Git értekezlet. Létrehoztuk a git projektet. minden csapattag klónozta a saját gépére a git könyvtárat. Megismerkedtünk a git alapvető funkcióival. Létrehoztunk két fejlesztési ágat. Ezek a development és main. A developmentben végezzük a fejlesztést. A mainbe csak a stabil verziókat másoljuk. Ennek célja, hogy ha a developmentben visszafordíthatatlan hibák keletkeznek, akkor a main utolsó verzióját visszaállíthassuk. A main tartalmát elsősorban Kárpáti fogja frissíteni.
2021.02.16. Áttekintés, követelmények rögzítése.	2 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	A megbeszélés során a csapat rögzítette a szoftver célját, korlátozásait, valamint részletesen átbeszéltük a követelményeket. A funkciók leírását Bárkányi

			vállalta el, pénteki határidővel. Az eddigi követelményeket Tóth rögzítette. Megbeszéltük, hogy a szerdai értekezleten részt vesz Simon, Kárpáti és Tóth.
2021.02.17 Egyéb követelmények rögzítése	0,5 óra	Tóth	Tóth pontosította az általános áttekintést, valamint rögzítette az erőforrásokkal, átadással, valamint egyéb nem funkcionális elvárásokkal kapcsolatos követelményeket.
2021.02.19 Use-Case megbeszélés	2 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	A megbeszélésen egyeztettük a funkciókat, use-case leírásokat és a szótár bejegyzéseket. Bárkányi ismertette a funkciók leírását és ezeket pontosítottuk. Simon pedig a Use-case-eket vázolta fel, amiket szintén a csapat minden tagja megvitattott. A megbeszélés eredménye, hogy Bárkányi beilleszti a dokumentumba a végleges funkciókat és Simon véglegesíti és lejegyzi a Use-Case-eket. Mohácsi vállalta, hogy elkészíti a szótárat. Tóth a Projekt tervet, Kárpáti pedig a Use-Case diagrammot dolgozza ki.
2021.02.20 Dokumentum összefoglalása, megbeszélés	2 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Mindenki bemutatta az általa kidolgozott feladatot, ezeknek megoldását megvitattuk és beillesztettük a dokumentumba.
2021.02.21 Dokumentum lezárása	0,5 óra	Bárkányi Tóth	A dokumentum formázása és lezárása.

3. Analízis modell kidolgozása 1.

67 – Brainstormers

Konzulens:
Simon Balázs

Csapattagok

Mohácsi Márton

Simon Zoltán

Kárpáti Márk András

Bárkányi Csaba

Tóth Ádám

T8ZT88

HRSNUG

O1BG0Z

BQI7QQ

QEYOW2

moha.mohacsi@gmail.com

simonzolika1999@gmail.com

mark.karpati1999@gmail.com

csaba.barkanyi.csaba@gmail.com

toth99adam@gmail.com

2021-05-10

3. Analízis modell kidolgozása

3.1 Objektum katalógus

3.1.1 Aszteroida

Azon objektum, amin telepesek, űrlények és robotok állhatnak, valamint azon égitestek, amelyek között közelekedhetnek. A különféle aszteroidákat különböző vastagságú sziklarétegek boríthatják. A sziklarétegek alatt nyersanyagok találhatóak. A nyersanyagot a játékos kibányászhatja, ezzel üregessé téve az aszteroidát. Ugyanakkor vannak aszteroidák, amelyekben nincs nyersanyag, ő belőlük nem nyerhető ki semmi, üresnek tekintendők, így, ha átfúrják a sziklarétegüköt, akkor ők automatikusan üregesek lesznek. Ha egy telepes vagy egy robot egy üreges, átfúrt aszteroidán tartózkodik akkor, ha napszél közeledik annak bekövetkezése előtt az üregben elbújik magától.

Az aszteroidák felszínén építhet a telepes teleportkaput, amelyel akár újabb aszteroida is elérhető távolságba kerülhet számára. A kapuk kívül lehetősége van bázis építésére is, amellyel sikerre is vitte a küldetését és megnyerte a játékot.

Ha egy aszteroida felrobban, akkor a rajta álló telepesek és űrlények meghalnak, valamint a rajta lévő robotok egy szomszédos aszteroidára sodródnak. A robbanás következtében a teleportkapu megsemmisül a párával együtt.

3.1.2 Bázis

A telepes építheti egy aszteroida felszínére bizonyos nyersanyagokból, ezzel megnyerve a játékot.

3.1.3 Nap

A pályán elhelyezkedve napkitöréseket okoznak, amelyek végigsöpörnek a hozzájuk tartozó aszteroidákon. Az aszteroidák elhelyezkedhetnek hozzájuk közel, vagy távol a játék során változva.

3.1.4 Robot

Önállóan tudnak lépni az aszteroidák között, illetve azok felszínén fújni is képesek. A játékos is megépítheti őket, ha a megfelelő nyersanyagok a rendelkezésére állnak. Az építés után egyből munkába is álnak.

3.1.5 Szén

Egy nyersanyag, amit az aszteroidákból lehet kibányászni.

3.1.6 Telepes

A játékos által irányított objektum. Tudnak mozogni az aszteroidák között, amelyhez teleportkapukat is használhatnak. A telepesek ezen kívül képesek aszteroidák közetrétegét átfúrni, és abból nyersanyagot kibányászni. A játékos velük tud létrehozni dolgokat, amelyek lehetnek bázisok, robotokat, de akár teleportkapuk is.

3.1.7 Teleportkapu

A játékos által irányított telepes képes építeni bizonyos nyersanyagok bírtokában, de egyszerre mindig csak egy párt. A kapuk csak aszteroidák felszínére helyezhetők le. Miután a pár minden része lekerült egy-egy különböző aszteroida felszínére a telepesek, illetve robotok képesek mozogni közöttük. Ha az adott aszteroida, amin elhelyezkedik a kapu megsemmisül, akkor a pár minden része megsemmisül, többet nem lehet használni.

3.1.8 Urán

Egy radioaktív nyersanyag, amit az aszteroidákból lehet kibányászni. Ha az aszteroida, ahol van napközelben tartózkodik, akkor az három környi napközelben lét után felrobban.

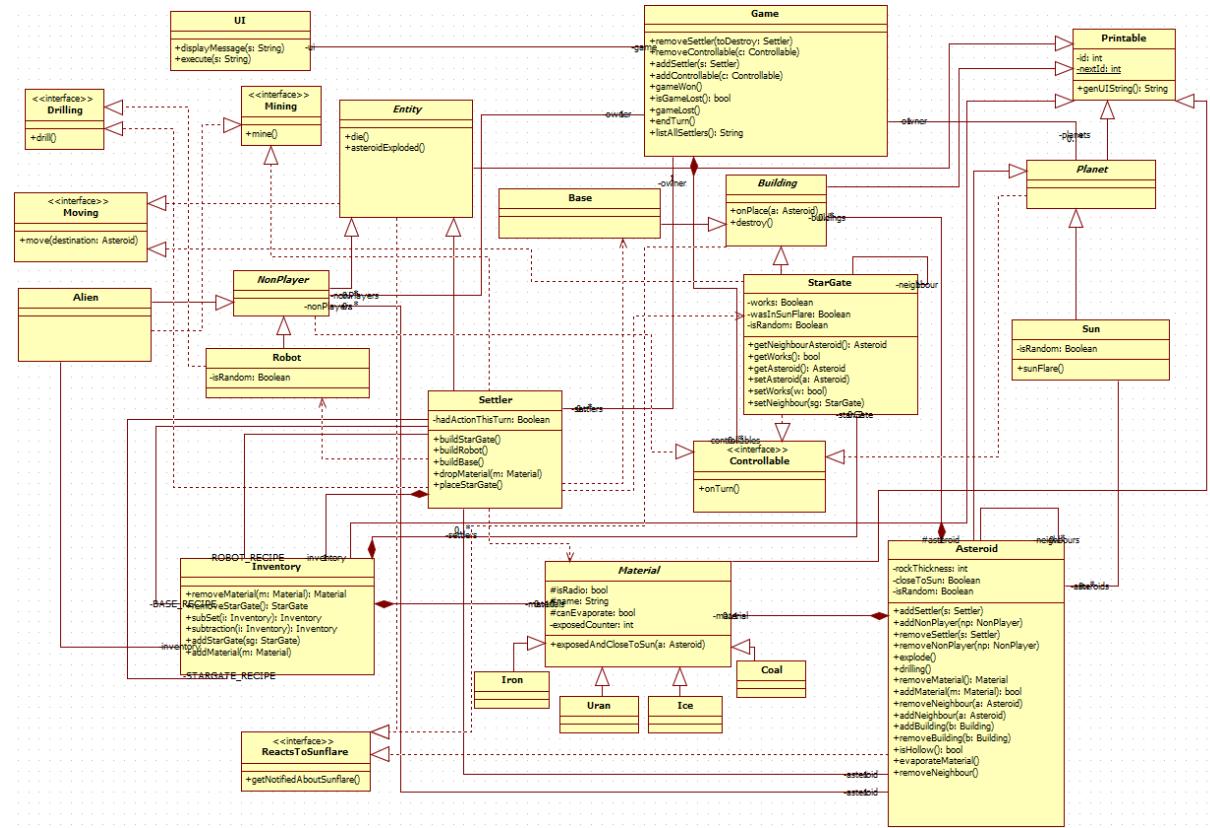
3.1.9 Vas

Egy nyersanyag, amit aszteroidákból lehet kibányászni.

3.1.10 Vízjég

Egy nyersanyag, amit aszteroidákból lehet kibányászni. Ha az aszteroida, ahonnan bányásszák napközelben van, akkor elpárolog.

3.2 Statikus struktúra diagramok



3.3 Osztályok leírása

3.3.1 Asteroid

- **Felelősség**

Számon tartja a rajta tartózkodó entitásokat, rá épített épületeket. Fúrhatóak és kinyerhető belőlük nyersanyag.

- **Ősosztályok**

Printable -> Planet -> Asteroid

- **Interfészek**

Controllable, ReactsToSunFlare

- **Attribútumok**

- **rockThickness : int:** Egész szám, amely nyilvántartja, hogy éppen milyen vastag a sziklaréteg az aszteroida felszínén.
- **closeToSun : bool:** Az aszteroida napközelségét jelző tagváltozó.
- **materials : Material[]:** Material lista, ami az aszteroidában lévő nyersanyagokat tárolja.
- **nonPlayers : NonPlayer[]:** NonPlayer típusú objektumokat tároló tagváltozó, amely az aszteroida felszínén tartózkodó nem játékos által irányított objektumokat tárolja.
- **settlers : Settler[]:** Settler típusú objektumokat tároló tagváltozó, amely az aszteroida felszínén tartózkodó játékos által irányított objektumokat tárolja.
- **buildings : Building:** Az aszteroida felszínén található Building típusú objektumokat tárolja.
- **neighbours : Asteroid:** Az aszteroida szomszédait tároló tagváltozó.
- **capacity : Int:** Hány nyersanyag fér az aszteroidába.
- **isRandom : Bool:** Véletlenszerűen változzon-e a napközelség.
- **mySun: Sun:** Az aszteroida napja.
- **owner : Game:** A játékot vezénylő objektum.
- **id: String** Egy adott objektumot azonosító egyedi jelölés.

- **Metódusok**

- **addEntity (Entity e):** Hozzáad egy Entity objektumot az aszteroida felszínéhez, eltárolja azt az entities tagváltozóban.
- **addNonPlayer(NonPlayer):** Hozzáad egy NonPlayer objektumot az aszteroida felszínéhez, eltárolja azt a nonPlayers tagváltozóban.
- **addSettler(NonPlayer):** Hozzáad egy Settler objektumot az aszteroida felszínéhez, eltárolja azt a settlers tagváltozóban.
- **removeNonPlayer(NonPlayer):** Eltávolítja a paraméterként átvett NonPlayer az aszteroida felszínéről, kiveszi őt az nonPlayers tagváltozóból.
- **removeSettler(Settler):** Eltávolítja a paraméterként átvett Settleret az aszteroida felszínéről, kiveszi őt az settlers tagváltozóból.
- **explode ():** Az aszteroida felrobban.
- **drilling ():** Az aszteroida felszínét megfúrják.
- **removeMaterial () : Material :** Eltávolítja az aszteroidia belsejében lévő nyersanyagot. Az eltávolított nyersanyaggal tér vissza.

- **addMaterial (Material m):** Az aszteroida belsejébe kerül a paraméterként átvett nyersanyag.
- **removeNeighbour (Asteroid a):** Kitörli a szomszédjai közül a paraméterként átvett aszteroidát.
- **addNeighbour (Asteroid a):** Hozzáadja szomszédaihoz a paraméterként átvett aszteroidát.
- **addBuilding (Building b):** Eltárolja a paraméterként átvett Building objektumot.
- **removeBuilding (Building b):** Eltávolítja a paraméterként átvett Building objektumot.
- **getNotifiedAboutSunFlare():** Napkitörésről értesül a ReactsToSunFlare interface-t megvalósító objektum.
- **isHollow(): Boolean:** Visszatérési értékében megadja, hogy üreges-e az aszteroida.
- **evaporateMaterial():** Elpárologtat egy nyersanyagot.
- **genSaveString(): String** Visszaadja az objektum szerelzálásához szükséges szöveget.
- **genUIString() : String** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.`initialize()`
- **onTurn()** minden kör elején hívott metódus, ami az autonóm viselkedést valósítja meg.

3.3.2 Base

- **Felelősség**

Ez az osztály felel a bázis kezeléséért, itt tároljuk a bázis helyét illetve ez az osztály dönti el, hogy egyáltalán meg tudjuk-e építeni azt.

- **Ősosztályok**

Printable -> Building -> Base

- **Interfészek**

Controllable, ReactsToSunFlare

- **Attribútumok**

- asteroid : Asteroid : Azt az aszteroidát tárolja amelyiken a bázis található.
- hadActionThisTurn: bool Végzett-e műveletet a jelenlegi játékkörben?
- owner: Game A játékot vezénylő objektum.
- id: String Egy adott objektumot azonosító egyedi jelölés.

- **Metódusok**

- **onPlace(a: Asteroid) :** Ez a függvény inicializálja a bázist abba a pillanatba amikor ráhelyezódik egy aszteroidára, illetve elvégez minden feladatot ami ezzel jár (pl.: a játék megnyerése)
- **destroy() :** Ez a függvény felel a bázis elpusztításáért, mielőtt elpusztul szól az aszteroidának amin tartózkodik arról hogy el fog pusztulni így az is ezt a változást figyelembe tudja venni.
- **genSaveString(): String** Visszaadja az objektum szerelzálásához szükséges szöveget.

- **genUIString() : String** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.`initialize()`
- **getNotifiedAboutSunFlare():** Napkitörésről értesül a ReactsToSunFlare interface-t megvalósító objektum.

3.3.3 Building

- **Felelősség**

Ez az ōsosztálya minden épületnek, felelős azért hogy megvalósítsa az épületek általános funkciót.

- **Ősosztályok**

Printable

- **Interfészek**

Controllable, ReactsToSunFlare

- **Attribútumok**

- **asteroid : Asteroid**: Azt az aszteroidát tárolja amelyiken az épület található.
- **hadActionThisTurn : bool** Végzett-e műveletet a jelenlegi játékkörben.
- **owner : Game**: A játékot vezénylő objektumra mutató hivatkozás.
- **id: String**: Egy adott objektumot azonosító egyedi jelölés.

- **Metódusok**

- **onPlace(a: Asteroid)** : Ez a függvény inicializálja az épületet abba a pillanatba amikor ráhelyeződik egy aszteroidára, illetve elvégez minden feladatot ami ezzel jár.
- **destroy()** : Ez a függvény felel az épület elpusztításáért, mielőtt elpusztul szól az aszteroidának amin tartózkodik arról hogy el fog pusztulni így az is ezt a változást figyelembe tudja venni.

3.3.4 Coal

- **Felelősség**

A szén nyersanyag osztálya.

- **Ősosztályok**

Printable -> Material -> Coal

- **Interfészek**

- **Attribútumok**

- **canEvaporate : boolean**: El tud-e párologni az anyag egy aszteroida belsejéből?
- **exposedCounter : int**: Számolja, hogy hány kör során volt feltárt aszteroidában, miközben az aszteroida napközelben volt.
- **isRadio : boolean**: Radioaktív-e a nyersanyag?
- **name : String**: Megadja a nyersanyag típusának nevét.
- **id : String**: Egy adott objektumot azonosító egyedi jelölés.

- **Metódusok**
 - **exposedAndCloseToSun(Asteroid a):** Körönként egyszer kell hívni, amikor egy nyersanyag egy aszteroida üregében van, az aszteroida kérge teljesen át van fúrva, és az aszteroida napközelben van.
 - **genSaveString(): String:** Visszaadja az objektum szerializálásához szükséges szöveget.
 - **genUIString() : String:** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.

3.3.5 Controllable

- **Felelősség**

Ezt az interfészet olyan osztályok valósítják meg amelyeknek autonóm módon minden körben csinálniuk kell valamit, tehát például a robotok.

- **Ősosztályok**
- **Metódusok**
 - **onTurn() :** Ez a függvény elvégzi az adott dolog körönkénti cselekedetét/cselekedeteit.

3.3.6 Entity

- **Felelősség**

Ez az ősosztálya minden entitásnak. Azért felelős, hogy az entitások általános feladatait megvalósítsa.

- **Ősosztályok**
Printable -> Entity
- **Interfészek**
ReactsToSunFlare, Moving
- **Attribútumok**
 - **asteroid : Asteroid:** Azt tárolja, hogy az adott entitás éppen melyik aszteroidán tartózkodik.
 - **owner : Game:** A játékot vezénylő objektumra mutató hivatkozás.
 - **id : String:** Egy adott objektumot azonosító egyedi jelölés.
- **Metódusok**
 - **asteroidExploded():** Értesíti az entitást, hogy az aszteroida, amin állt megsemmisült. A konkrét implementáció az entitás típusától függ.
 - **die():** Elpusztítja az entitást. A konkrét implementáció az entitás típusától függ.
 - **genSaveString(): String:** Visszaadja az objektum szerializálásához szükséges szöveget.
 - **genUIString() : String:** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.

- **getNotifiedAboutSunflare()**: Hatására az entitás elpusztítja magát, ha nem átfűrt kérő vagy nem üreges belsejű aszteroidán tartózkodik.

3.3.7 Game

- **Felelősség**

Ez az osztály felel a játék körönkénti menetéért, az összes olyan osztálynak, amelynek dolga van a körben a kör elején szól, hogy végezze el a dolgát.

- **Ősosztályok**
- **Interfészek**
- **Attribútumok**
 - **asteroids : List<Asteroid>**: A játékban szereplő aszteroidák listája.
 - **asteroidsToRemove : List<Asteroid>**: Egy körben megsemmisülő aszteroidák listája. Ez a lista minden körben kiürül, amikor a Game végleg törli a hivatkozást az aszteroidáakra.
 - **nonPlayers : List<NonPlayer>**: A játékban szereplő nem játékos által irányított entitások listája.
 - **nonPlayersToRemove : List<NonPlayer>**: Egy körben megsemmisülő nem játékos által irányított entitások listája. Ez a lista minden körben kiürül, amikor a Game végleg törli a hivatkozást az entitásokra.
 - **settlers : List<Settler>**: A játékban szereplő telepesek listája.
 - **settlersToRemove : List<Settler>**: Egy körben megsemmisülő telepesek listája. Ez a lista minden körben kiürül, amikor a Game végleg törli a hivatkozást a telepesekre.
 - **suns : List<Sun>**: A játékban szereplő napok listája.
 - **sunsToRemove : List<Sun>**: Egy körben megsemmisülő napok listája. Ez a lista minden körben kiürül, amikor a Game végleg törli a hivatkozást a napokra.
 - **turnEvents : List<String>**: Egy string lista, ami az adott körben lejátszódott eseményekről tájékoztatja a felhasználót.
 - **ui : UI**: Hivatkozás a felhasználói inputot megvalósító objektumra.
- **Metódusok**
 - **addAsteroid(Asteroid)**: Új aszteroidát ad a megfelelő listához.
 - **addNonPlayer(NonPlayer)**: Új nem játékos által irányított entitást ad a megfelelő listához.
 - **addSettler(Settler)**: Új telepet ad a megfelelő listához.
 - **addSun(Sun)**: Új napot ad a megfelelő listához.
 - **addTurnEvent(String)**: Új bejegyzéssel bővíti a kör eseményeit tároló listát.
 - **cleanup()**: Megszünteti a hivatkozást az összes megsemmisítére kijelölt telepesre, napra, aszteroidára és nem játékos által irányított entitásra.
 - **gameLost()**: Hívására a telepesek elveszítik a játékot.
 - **gameWon()**: Hívására a telepesek megnyerik a játékot.
 - **initGame(isManual : Boolean)**: Inicializálja a játékot. Ha false értékű paraméterrel hívjuk véletlenszerűen generálja a játékmezőt.
 - **isGameLost()**: Akkor tekinti elveszítettnek a játszmát, ha nem maradt telepes.
 - **listAll()**: Struktúrált módon listázza a játékban szereplő összes főbb objektumhoz kapcsolódó információkat.

- **listAllSettlers():** Listázza a telepesekhez kapcsolódó információkat.
- **listTurnEvents():** Listázza egy körben történt eseményeket.
- **loadAlien(Scanner, String, List<Alien>, List<Asteroid>, List<Inventory>):** Az űrlény fájlból való betöltésére szolgál.
- **loadAsteroid(Scanner, List<Asteroid>, List<Settler>, List<Material>, List<Robot>, List<Alien>, List<Base>, List<StarGate>):** Az aszteroida fájlból való betöltésére szolgál.
- **loadBase(Scanner, String, List<Base>, List<Asteroid>):** A bázis fájlból való betöltésére szolgál.
- **loadGame():** Betölti a "game.txt" fájlban tárolt játékállást.
- **loadInventory(Scanner, List<Inventory>, List<Material>, List<StarGate>):** A tároló fájlból való betöltésére szolgál.
- **loadMaterial(Material, String, Scanner, List<Material>):** A nyersanyag fájlból való betöltésére szolgál.
- **loadRobot(Scanner, String, List<Robot>, List<Asteroid>):** A robot fájlból való betöltésére szolgál.
- **loadSettler(Scanner, List<Settler>, List<Asteroid>, List<Inventory>):** A telepes fájlból való betöltésére szolgál.
- **loadStarGate(Scanner, String, List<StarGate>, List<Asteroid>):** A teleportkapu fájlból való betöltésére szolgál.
- **loadSun(Scanner, List<Sun>, List<Asteroid>):** A nap fájlból való betöltésére szolgál.
- **removeAsteroid(Asteroid):** A paraméterként kapott objektumot megsemmisítésre jelöli ki.
- **removeNonPlayer(NonPlayer):** A paraméterként kapott objektumot megsemmisítésre jelöli ki.
- **removeSettler(Settler):** A paraméterként kapott objektumot megsemmisítésre jelöli ki.
- **removeSun(Sun):** A paraméterként kapott objektumot megsemmisítésre jelöli ki.
- **saveGame():** Elmenti az aktuális játékállást a "game.txt" fájlba.
- **startGame():** Az inicializálás után ezzel a metódussal indíthatjuk a játékot.
- **startTurn():** Lezárja az előző kört. Megkezdi az újat. Felelőssége, hogy véleg törölje a hivatkozást az előző körben erre kijelölt objektumokra, és, hogy onTurn() hívást végezzen minden erre alkalmas objektumon.

3.3.8 Ice

- **Felelősség**

A vízjég nyersanyagot jelképező osztály, ha napközelben bányásszák, akkor elpárolog.

- **Ősosztályok**

Printable -> Material -> Ice

- **Interfészek**

- **Attribútumok**

- **canEvaporate : boolean:** El tud-e párologni az anyag egy aszteroida belsejéből?
- **exposedCounter : int:** Számolja, hogy hány kör során volt feltárt aszteroidában, miközben az aszteroida napközelben volt.

- **isRadio : boolean:** Radioaktív-e a nyersanyag?
 - **name : String:** Megadja a nyersanyag típusának nevét.
 - **id : String:** Egy adott objektumot azonosító egyedi jelölés.
 - **Metódusok**
 - **exposedAndCloseToSun(Asteroid a):** Körönként egyszer kell hívni, amikor egy a nyersanyag egy aszteroida üregében van, az aszteroida kérge teljesen át van fúrva, és az aszteroida napközelben van.
 - **genSaveString() : String:** Visszaadja az objektum serializálásához szükséges szöveget.
 - **genUIString() : String:** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.
- ### 3.3.9 Inventory
- **Felelősség**
Az osztály egy telepes hádzsákjában található dolgokért felel, abba a megfelelő nyersanyagokat berakja, ha a telepes kitermeli azokat, illetve onnan kiveszi őket ha a telepes felhasználja azokat.
 - **Ősosztályok**
 - **Interfészek**
 - **Attribútumok**
 - **capacityM : int:** A maximális mennyisége, amennyi nyersanyagot tárolni tud.
 - **capacitySG : int:** A maximális mennyisége, amennyi teleportkaput tárolni tud.
 - **materials : LinkedList<Material>:** A tárolt nyersanyagok.
 - **starGates : LinkedList<StarGate>:** A tárolt teleportkapuk.
 - **id : String:** Egy adott objektumot azonosító egyedi jelölés.
 - **Metódusok**
 - **addMaterial(Material):** Betesz egy nyersanyagot a tárolóba.
 - **addStarGate(StarGate):** Betesz egy teleportkaput a tárolóba.
 - **genSaveString() : String:** Visszaadja az objektum sserializálásához szükséges szöveget.
 - **genUIString() : String:** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.
 - **isFullMaterials() : bool:** Megtelt-e a tároló nyersanyagokkal?
 - **isFullStarGates() : bool:** Megtelt-e a tároló teleportkapukkal?
 - **isHaveStarGate(StarGate) : bool:** Tartalmazza-e a paraméterként kapott teleportkaput?
 - **removeMaterial(Material) : Material:** Kivesz egy ugyanolyan típusú nyersanyagot, mint amilyet paraméterként kapott.
 - **removeMaterialFromList(LinkedList<Material>, Material):** Egy paraméterként kapott listából vesz ki nyersanyagot.
 - **removeStarGate(StarGate):** Kivesz egy paraméterként kapott nyersanyagot a tárolóból.
 - **subSet(LinkedList<Material>) : LinkedList<Material>:** Összehasonlítja a paraméterként kapott és a tárolóban lévő nyersanyag listát. Visszaadja a két lista különbségét.

- **subtraction(LinkedList<Material>) : LinkedList<Material>:** Kivonja a paraméterként kapott listát a tárolóban lévő nyersanyag listából. Visszaadja a két lista különbségét.

3.3.10 Iron

- **Felelősség**

A vas nyersanyagot reprezentáló osztály.

- **Ősosztályok**

Printable -> Material -> Iron

- **Interfészek**

- **Attribútumok**

- **canEvaporate : boolean:** El tud-e párologni az anyag egy aszteroida belsejéből?
- **exposedCounter : int:** Számolja, hogy hány kör során volt feltárt aszteroidában, miközben az aszteroida napközelben volt.
- **isRadio : boolean:** Radioaktív-e a nyersanyag?
- **name : String:** Megadja a nyersanyag típusának nevét.
- **id : String:** Egy adott objektumot azonosító egyedi jelölés.

- **Metódusok**

- **void addInventory (Inventory i):** A paraméterként átvett Inventoryba bekerül az adott nyersanyag.

3.3.11 Material

- **Felelősség**

A játékban lévő nyersanyagok közös, absztrakt ősosztálya.

- **Ősosztályok**

Printable -> Material

- **Interfészek**

- **Attribútumok**

- **canEvaporate : boolean:** El tud-e párologni az anyag egy aszteroida belsejéből?
- **exposedCounter : int:** Számolja, hogy hány kör során volt feltárt aszteroidában, miközben az aszteroida napközelben volt.
- **isRadio : boolean:** Radioaktív-e a nyersanyag?
- **name : String:** Megadja a nyersanyag típusának nevét.
- **id : String:** Egy adott objektumot azonosító egyedi jelölés.

- **Metódusok**

- **exposedAndCloseToSun(Asteroid a):** Körönként egyszer kell hívni, amikor egy a nyersanyag egy aszteroida üregében van, az aszteroida kérge teljesen át van fúrva, és az aszteroida napközelben van.

- **genSaveString(): String:** Visszaadja az objektum serializálásához szükséges szöveget.
- **genUIString() : String:** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.

3.3.12 Mining

- **Felelősség**
Azon osztályok valósítják meg, amelyek képesek bányászni.
- **Ősosztályok**
- **Interfészek**
- **Attribútumok**
- **Metódusok**
- **mine():** Bányászás műveletét kell, hogy megvalósítsa.

3.3.13 Moving

- **Felelősség**
Azon osztályok valósítják meg, amelyek képesek mozogni.
- **Ősosztályok**
- **Interfészek**
- **Attribútumok**
- **Metódusok**
- **move(Asteroid):** Aszteroidára történő mozgást kell, hogy implementálja.
- **move(StarGate):** Teleportkapura történő mozgást kell, hogy implementálja.

3.3.14 NonPlayer

- **Felelősség**
A nem játékos által irányított entitások ősosztálya.
- **Ősosztályok**
- **Printable -> Entity -> NonPlayer**
- **Interfészek**
Controllable, ReactsToSunFlare, Moving
- **Attribútumok**
- **counter : int:** Az önálló viselkedéshez szükséges változó, ami lehetővé teszi, hogy változatos cselekvéseket hajtson végre az entitás.
- **isRandom : boolean:** Véletlenszerű viselkedést hajtson-e végre az entitás?

- **owner: Game:** A játékot vezérylő objektumra mutató hivatkozás.
- **id : String:** Egy adott objektumot azonosító egyedi jelölés.
- **asteroid : Asteroid:** Azt tárolja, hogy az adott entitás éppen melyik aszteroidán tartózkodik.
- **Metódusok**
- **asteroidExploded():** Értesíti az entitást, hogy az aszteroida, amin állt megsemmisült. A konkrét implementáció az entitás típusától függ.
- **die():** Elpusztítja az entitást.
- **genSaveString(): String:** Visszaadja az objektum szerializálásához szükséges szöveget.
- **genUIString() : String:** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.
- **getNotifiedAboutSunflare():** Hatására az entitás elpusztítja magát, ha nem átfúrt kérgű vagy nem üreges belsejű aszteroidán tartózkodik.
- **move(Asteroid):** Az entitás egy szomszédos aszteroidára lép.
- **move(StarGate):** Az entitás egy teleportkapura lép.
- **onTurn():** minden kör elején hívott metódus, ami az autonóm viselkedést valósítja meg.

3.3.15 Planet

- **Felelősség**

A Sun és Asteroid osztályok absztrakt őse. Későbbiekben is, ha bármilyen újabb „bolygó” kerül a játékba, belőle kell származtatni.

- **Ősosztályok**

Printable -> Planet

- **Interfészek**

- **Attribútumok**

- **owner: Game:** A játékot vezérylő objektum.
- **id : String:** Egy adott objektumot azonosító egyedi jelölés.

- **Metódusok**

3.3.16 Printable

Felelősség

Azon osztályok őse, amelyek szerializálhatóak, a felhasználó számára információkat lehet róluk megjeleníteni és egyedi azonosítóval rendelkeznek.

Ősosztályok

Interfészek

Attribútumok

nextId : int: Egy statikus változó, amely megjegyzi, hogy az összes objektum id-jében mi volt a legnagyobb kiosztott index.

id : String: Egy objektum egyedi azonosítója.

Metódusok

genSaveString(): String: Visszaadja az objektum szerializálásához szükséges szöveget.

genUIString() : String: Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.

3.3.17 ReactsToSunFlare**Felelősség**

Ezt az interface-t azok az osztályok valósítják meg, amelyek valamilyen módon reagálni tudnak a napkitörésekre.

Ősosztályok**Interfészek****Attribútumok****Metódusok**

getNotifiedAboutSunflare(): Hatására az entitás elpusztítja magát, ha nem átfűrt kérgű vagy nem üreges belsejű aszteroidán tartózkodik.

3.3.18 Robot

- Felelősség**

Ez az osztály felel a robotokért s azoknak a mozgatásáért, irányításáért.

- Ősosztályok**

Printable -> Entity -> NonPlayer -> Robot

- Interfészek**

Controllable, Drilling, ReactsToSunFlare, Moving

- Attribútumok**

- counter : int:** Az önálló viselkedéshez szükséges változó, ami lehetővé teszi, hogy változatos cselekvéseket hajtson végre az entitás.
- isRandom : boolean:** Véletlenszerű viselkedést hajtson-e végre az entitás?
- owner: Game:** A játékot vezénylő objektumra mutató hivatkozás.
- id : String:** Egy adott objektumot azonosító egyedi jelölés.
- asteroid : Asteroid:** Azt tárolja, hogy az adott entitás éppen melyik aszteroidán tartózkodik.

- **Metódusok**
- **asteroidExploded()**: Értesíti az entitást, hogy az aszteroida, amin állt megsemmisült. A konkrét implementáció az entitás típusától függ.
- **die()**: Elpusztítja az entitást.
- **drill()**: A robot fűr egyet az aszteroidáján.
- **genSaveString() : String**: Visszaadja az objektum serializálásához szükséges szöveget.
- **genUIString() : String**: Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.
- **getNotifiedAboutSunflare()**: Hatására az entitás elpusztítja magát, ha nem átfúrt kérgű vagy nem üreges belsejű aszteroidán tartózkodik.
- **move(Asteroid)**: Az entitás egy szomszédos aszteroidára lép.
- **move(StarGate)**: Az entitás egy teleportkapura lép.
- **onTurn()**: minden kör elején hívott metódus, ami az autonóm viselkedést valósítja meg.

3.3.19 Settler

- **Felelősség**

Ez az osztály felel a telepesekért, azok mozgatásáért s irányításáért.

- **Ősosztályok**

Printable -> Entity -> Settler

- **Interfészek**

Drilling, Mining, ReactsToSunFlare, Moving

- **Attribútumok**

- **BASE_RECIPE : Inventory**: A bázis építésének receptje.
- **hadactionthisturn : boolean**: Volt-e már akció ebben a körben.
- **inventory : Inventory**: A telepes tárolója.
- **ROBOT_RECIPE : Inventory**: A robot építésének receptje.
- **STARGATE_RECIPE : Inventory**: A teleportkapu építésének receptje.
- **isRandom : boolean**: Véletlenszerű viselkedést hajtson-e végre az entitás?
- **owner: Game**: A játékot vezénylő objektumra mutató hivatkozás.
- **id : String**: Egy adott objektumot azonosító egyedi jelölés.
- **asteroid : Asteroid**: Azt tárolja, hogy az adott entitás éppen melyik aszteroidán tartózkodik.

- **Metódusok**

- **asteroidExploded()**: Értesíti az entitást, hogy az aszteroida, amin állt megsemmisült. A konkrét implementáció az entitás típusától függ.
- **buildBase()**: Bázis építése.
- **buildRobot()**: Robot építése.
- **buildStarGate()**: Teleportkapu építése.
- **die()**: Elpusztítja az entitást.
- **drill()**: A robot fűr egyet az aszteroidáján.

- **dropMaterial(Material):** Paraméterként kapott nyersanyag eldobása.
- **genSaveString(): String:** Visszaadja az objektum szerializálásához szükséges szöveget.
- **genUIString() : String:** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.
- **getNotifiedAboutSunflare():** Hatására az entitás elpusztítja magát, ha nem átfűrt kérgű vagy nem üreges belsejű aszteroidán tartózkodik.
- **mine():** Bányászás.
- **move(Asteroid):** Az entitás egy szomszédos aszteroidára lép.
- **move(StarGate):** Az entitás egy teleportkapura lép.
- **placeStarGate(StarGate):** Teleportkapu lehelyezése.

3.3.20 StarGate

Felelősség

A teleportkaput valósítja meg, ami aszteroidára helyezhető. Ha egy kapupár minden két eleme le van helyezve egy-egy aszteroidára, akkor eljuthatunk rajtuk keresztül az egyikről a másik aszteroidára.

Ősosztályok

Printable -> Building -> StarGate

Interfészek

Moving, Controllable, ReactsToSunFlare

Attribútumok

isRandom : boolean: Véletlenszerű viselkedést hajtson-e végre az kapu, ha meghibásodott?

neighbour : StarGate: A kapu szomszédja (párja).

wasInSunFlare : boolean: Volt-e már napviharnak kitéve?

works : boolean: Működik-e?

asteroid : Asteroid: Azt az aszteroidát tárolja amelyiken az épület található.

hadActionThisTurn : boolean: Végzett-e műveletet a jelenlegi játékkörben.

owner : Game: A játékot vezénylő objektumra mutató hivatkozás.

id : String: Egy adott objektumot azonosító egyedi jelölés.

Metódusok

destroy(): Elpusztítja magát.

genSaveString(): String: Visszaadja az objektum szerializálásához szükséges szöveget.

genUIString() : String: Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.

move(Asteroid): A kapu egy szomszédos aszteroidára lép.

move(StarGate): A kapu egy másik teleportkapura lép.

onPlace(): Aszteroidára történő lehelyezéskor hívódik.

onTurn(): minden kör elején hívott metódus, ami az autonóm viselkedést valósítja meg. (Itt csak akkor van szerepe, ha már érte napvihar a kaput.)

3.3.21 Sun

- **Felelősség**

Ő a nap a játékban, ő tudja végrehajtani a napkitörést a játék során.

- **Ősosztályok**

Printable -> Planet -> Sun

- **Interfészek**

Controllable

- **Attribútumok**

asteroids : List<Asteroid>: A naphoz tartozó aszteroidák.

isRandom : boolean: Véletlenszerű viselkedést hajtson-e végre a nap?

owner : Game: A játékot vezénylő objektumra mutató hivatkozás.

id : String: Egy adott objektumot azonosító egyedi jelölés.

- **Metódusok**

- **addAsteroid(Asteroid):** Új aszteroida hozzáadása a naphoz.
- **genSaveString() : String:** Visszaadja az objektum serializálásához szükséges szöveget.
- **genUIString() : String:** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.
- **onTurn():** minden kör elején hívott metódus, ami az autonóm viselkedést valósítja meg.
- **removeAsteroid(Asteroid):** Aszteroida eltávolítása a nap tárolójából.
- **sunFlare():** Napvihar előidézése.

3.3.22 Uran

- **Felelősség**

Radioaktív nyersanyag a játékban.

- **Ősosztályok**

Printable -> Material -> Uran

- **Interfészek**

- **Attribútumok**

canEvaporate : boolean: El tud-e párologni az anyag egy aszteroida belsejéből?

exposedCounter : int: Számolja, hogy hány kör során volt feltárt aszteroidában, miközben az aszteroida napközelben volt.

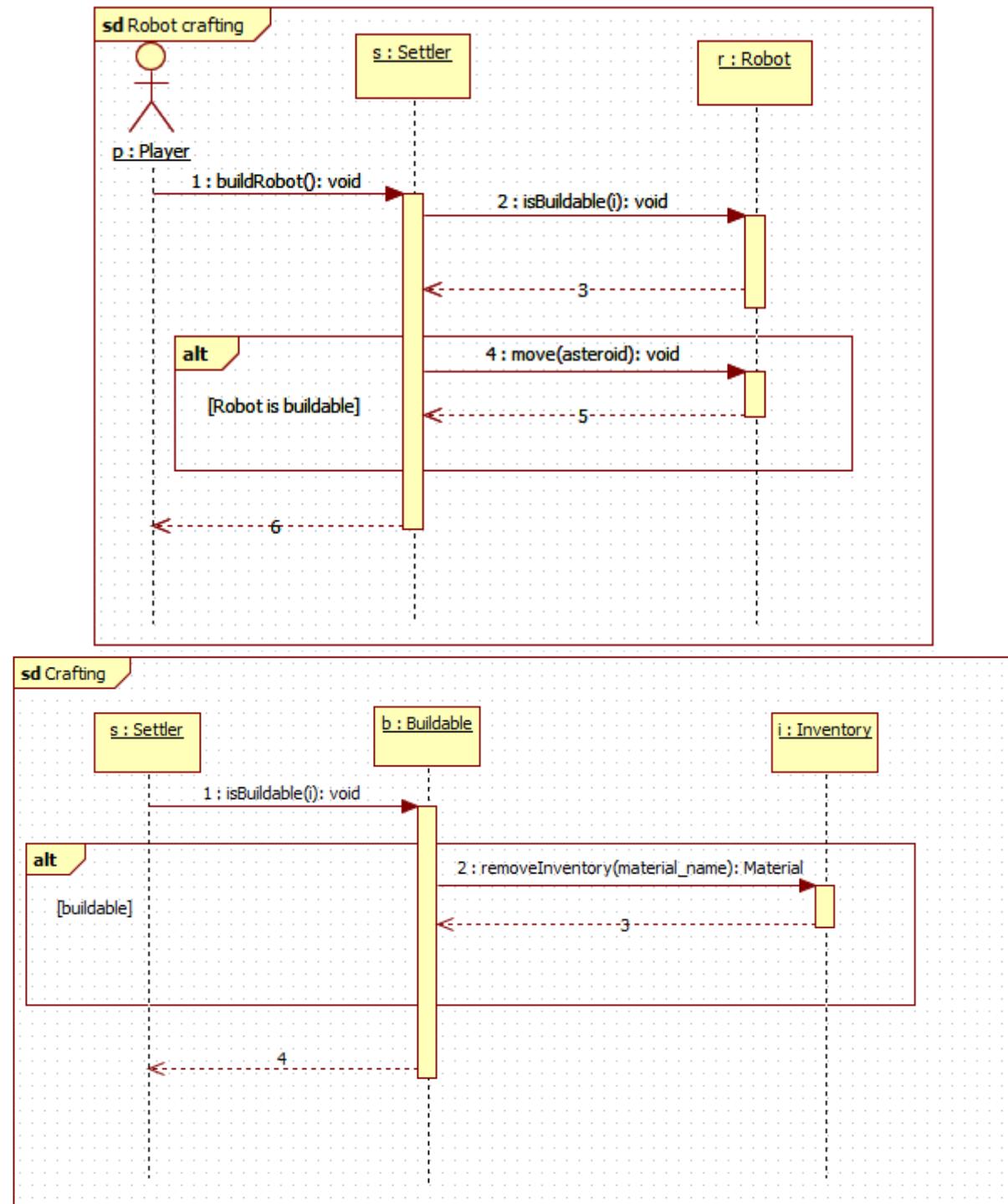
isRadio : boolean: Radioaktív-e a nyersanyag?

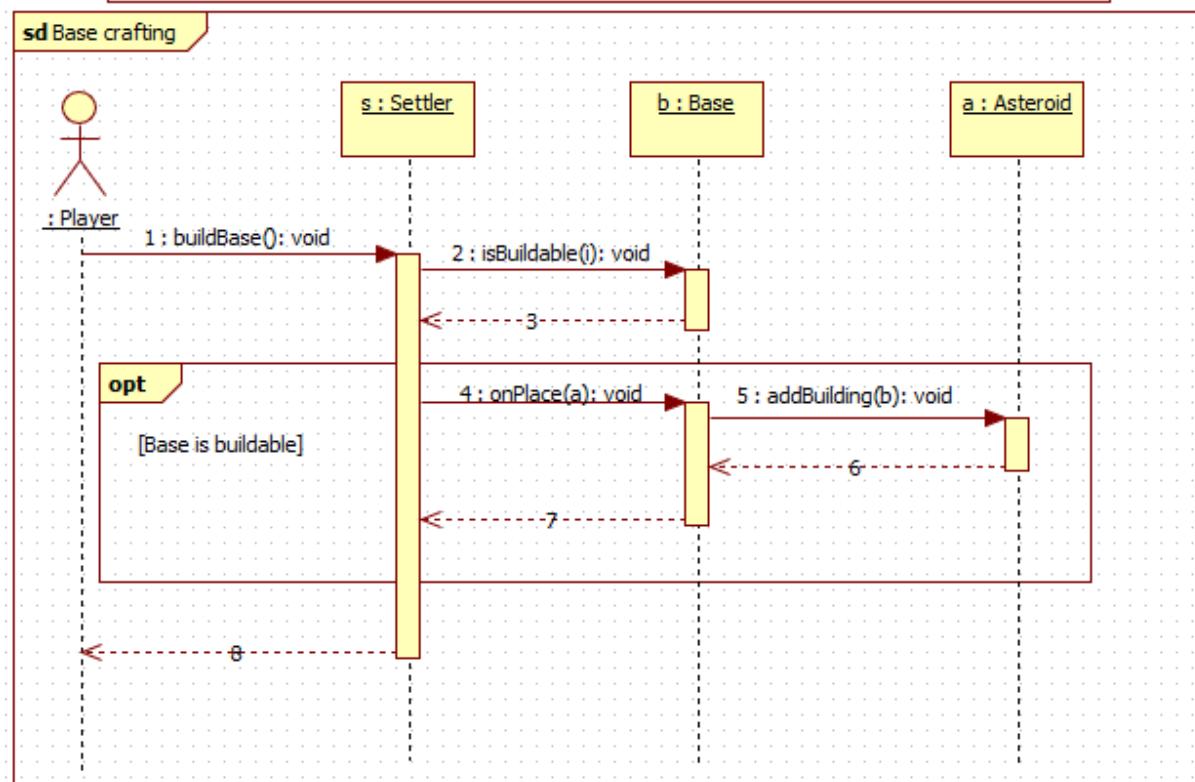
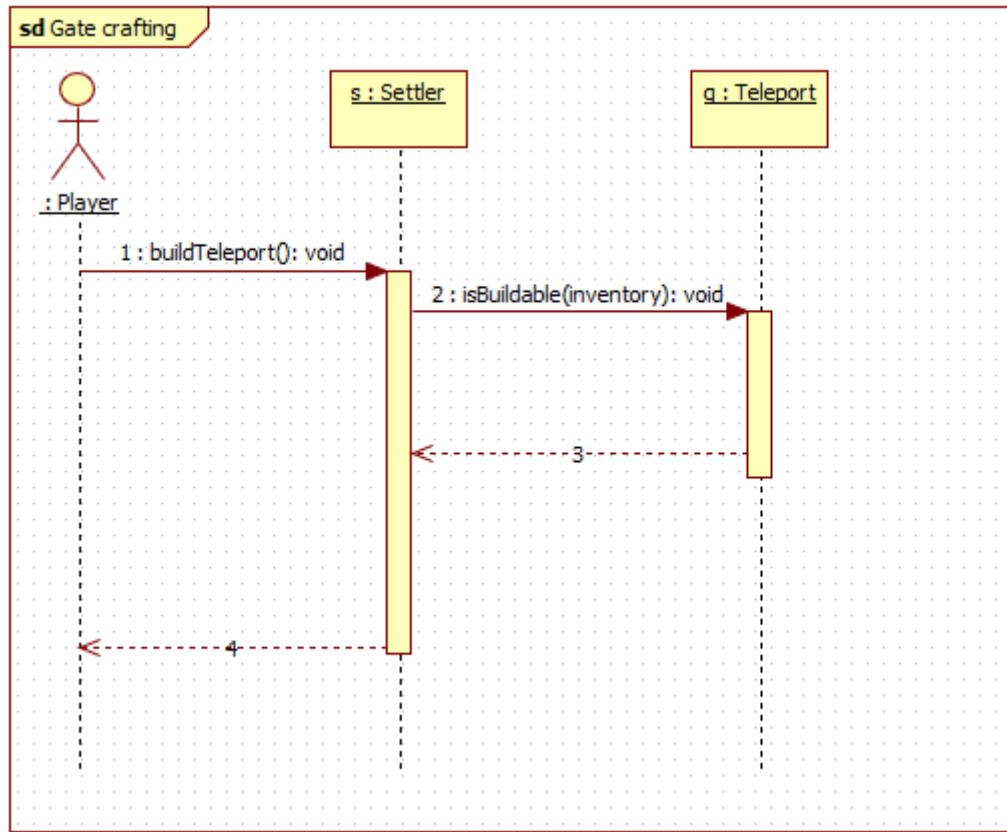
name : String: Megadja a nyersanyag típusának nevét.

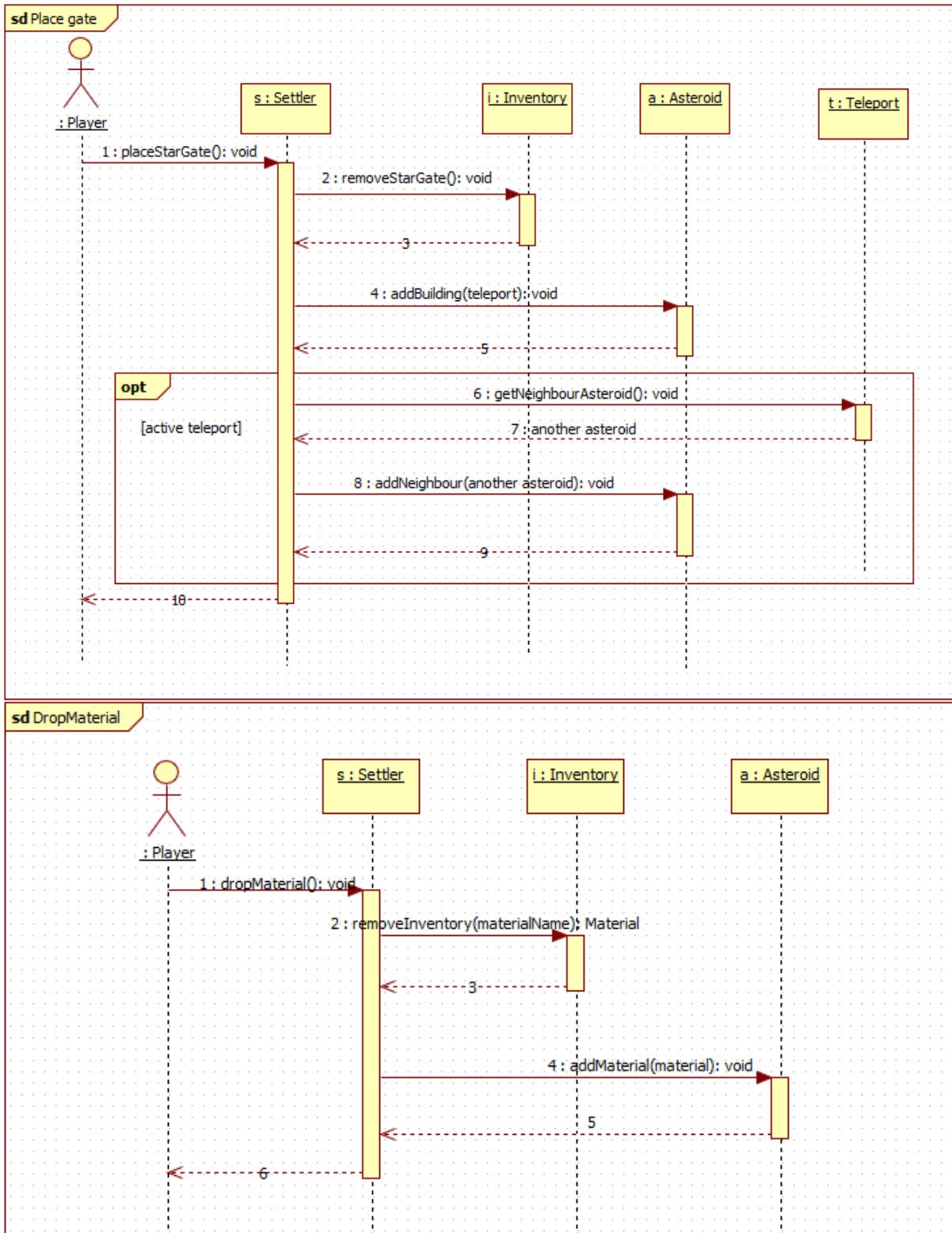
id : String: Egy adott objektumot azonosító egyedi jelölés.

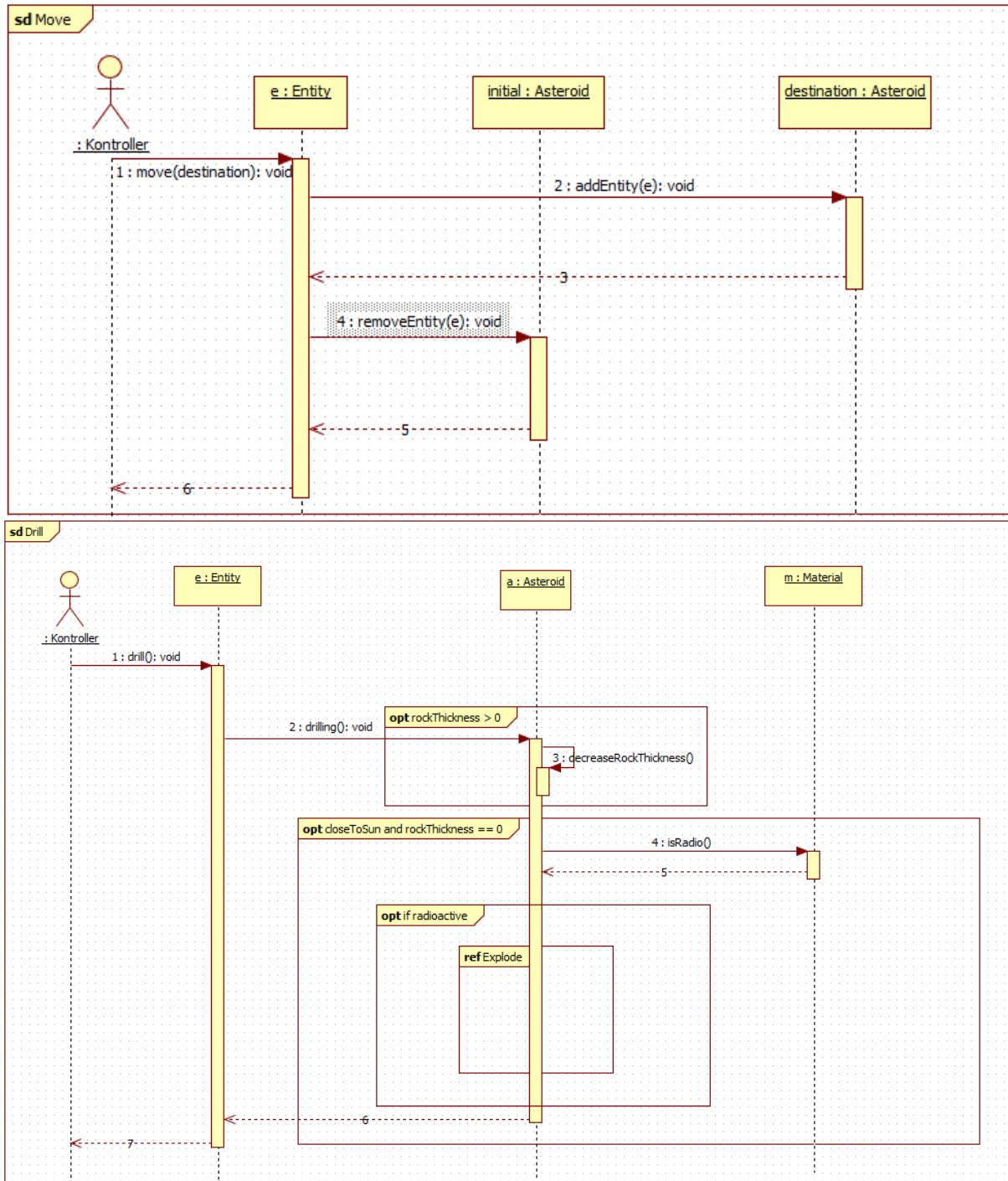
- **Metódusok**
- **exposedAndCloseToSun(Asteroid a):** Körönként egyszer kell hívni, amikor egy a nyersanyag egy aszteroida üregében van, az aszteroida kérge teljesen át van fúrva, és az aszteroida napközelben van.
- **genSaveString(): String:** Visszaadja az objektum serializálásához szükséges szöveget.
- **genUIString() : String:** Visszaadja az objektumról a játékos számára lényeges információkat tartalmazó szöveget.

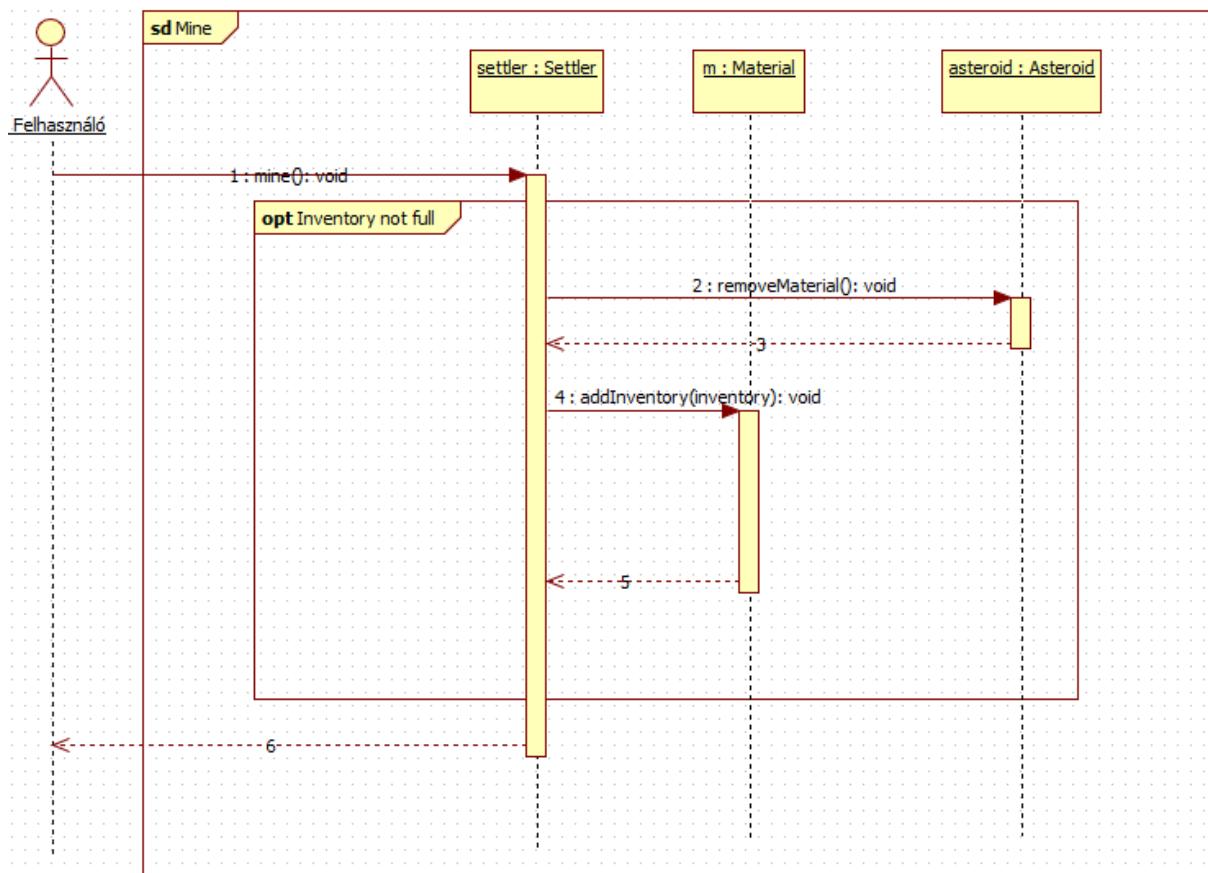
3.4 Szekvencia diagramok

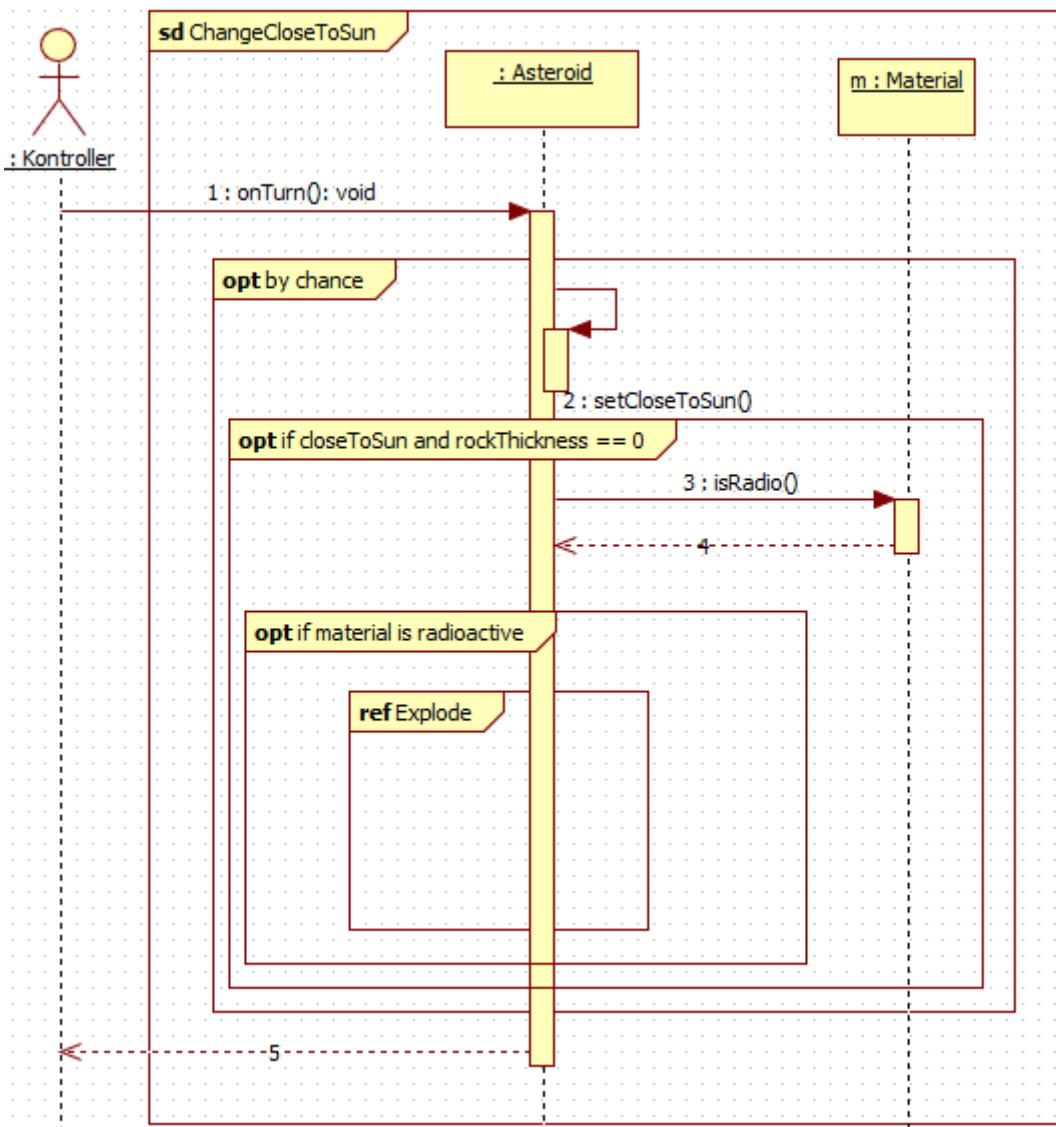


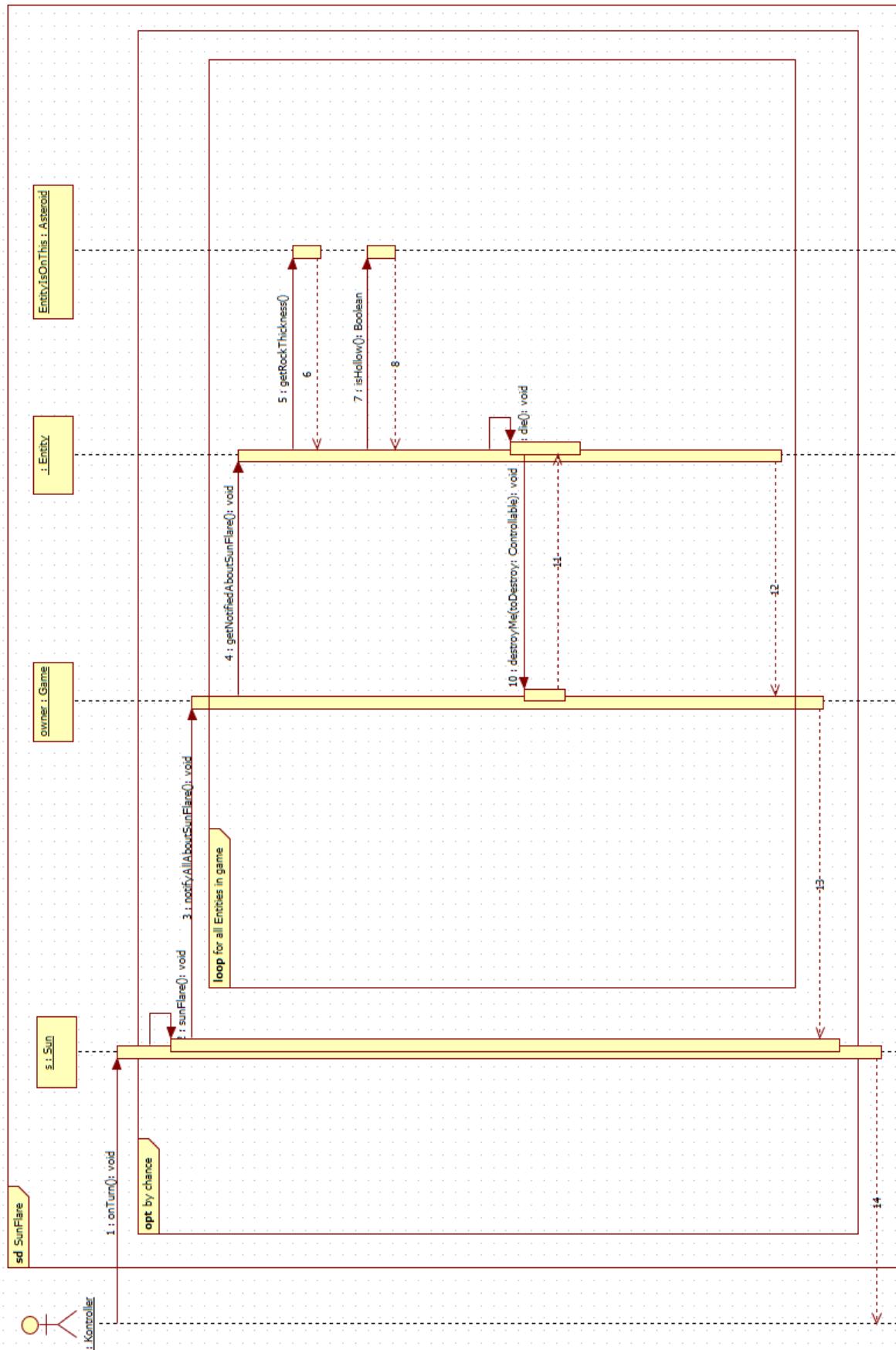


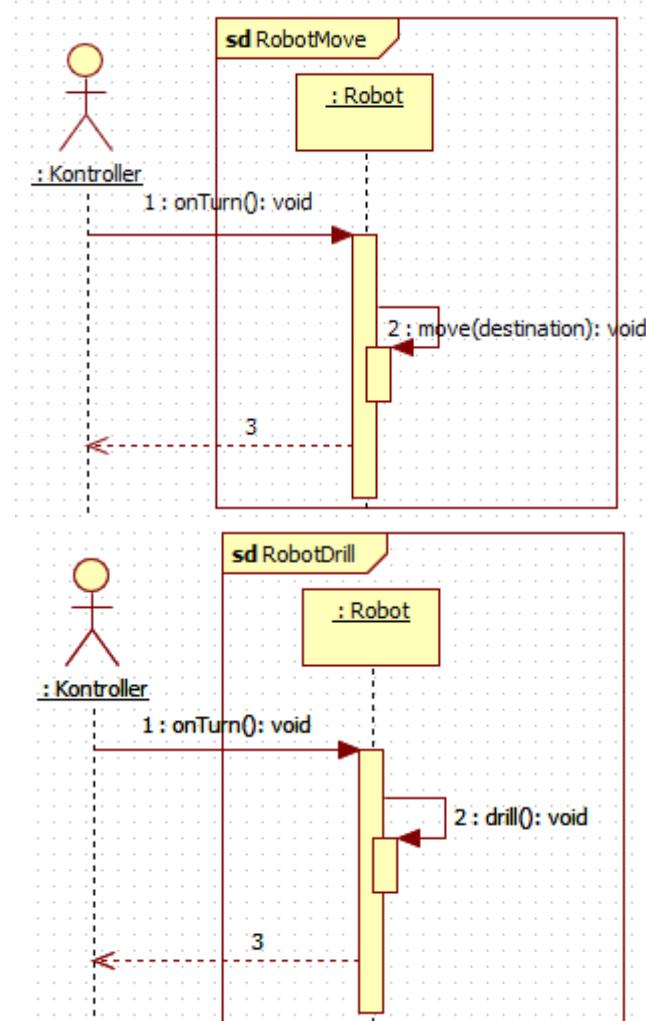


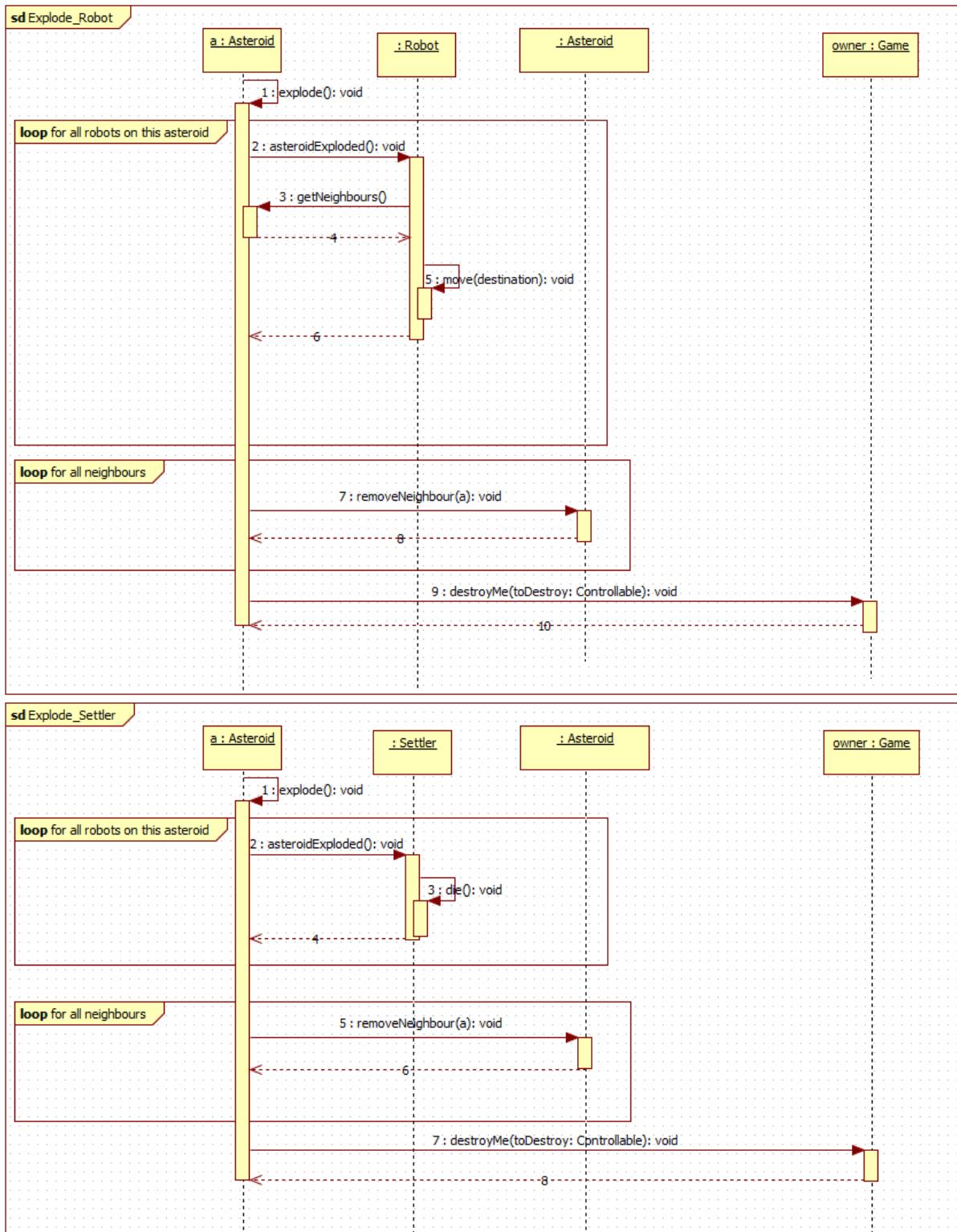












3.5 State-chartok

3.6 Napló

Kezdet	Időtartam	Résznevők	Leírás
2021.02.23. 20:00	1 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet. A csapat közösen létrehozott néhány osztályt.
2021.02.24. 10:15	1 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Konzultáció.
2021.02.26. 15:00	3,5 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet, a csapat eldöntötté az egyes osztályok felelősségeit, metódusait. Bárkányi vállalta, hogy áttekinti az objektumkatalógust, Tóth és Mohácsi vállalta, hogy dokumentálja az osztályokat, Kárpáti, Simon, Bárkányi, pedig a szekvencia diagrammokat készíti el.
2021.02.28 20:15	1 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet, mindenki elhozta az általa készített részeket, ezeket átnéztük és összefeszültük. A dokumentumot lezártuk.

5. Skeleton tervezése

67 – Brainstormers

Konzulens:
Simon Balázs

Csapattagok

Mohácsi Márton

Simon Zoltán

Kárpáti Márk András

Bárkányi Csaba

Tóth Ádám

T8ZT88

HRSNUG

O1BG0Z

BQI7QQ

QEYOW2

moha.mohacsi@gmail.com

simonzolika1999@gmail.com

mark.karpati1999@gmail.com

csaba.barkanyi.csaba@gmail.com

toth99adam@gmail.com

2021-05-10

4. Szkeleton tervezése

4.1 Megjegyzés

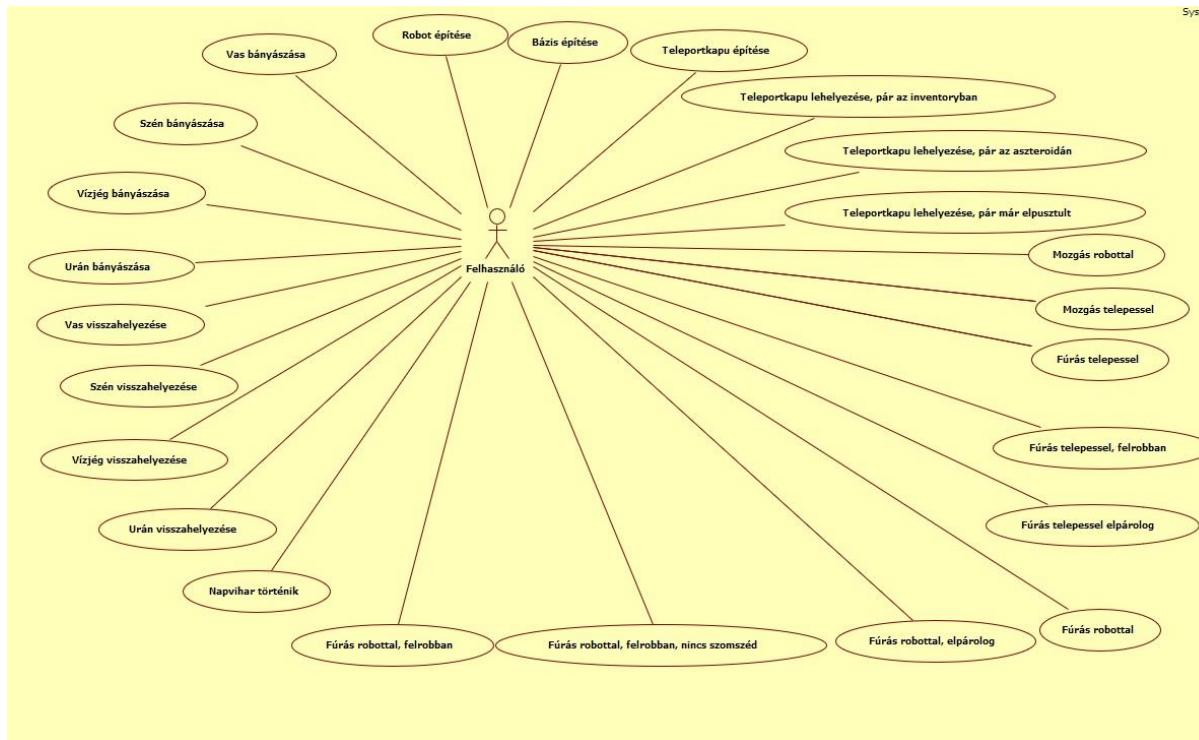
A robot, teleportkapu és bázis építéséhez kapcsolódó diagrammok esetében a megépítésükhoz szükséges nyersanyagok listáját tartalmazó receptek elemei külön nem kerültek feltüntetésre az inventory-val ellentétben. Ennek oka, hogy a diagrammok kezelhetetlen méretűvé nőttek volna, valamint a receptek konstansok, így a játék során tartalmuk végig állandó. A receptekben, ha fel lennének külön tüntetve a nyersanyag példányok ugyan azon típusok szerepelnének, mint jelen esetben az inventory-ban. Ez azért igaz, mivel minden esetben itt csak azon nyersanyagok lettek az inventory-ba helyezve, amelyek kellenek is az adott dolog elkészítéséhez.

A forgatókönyvek megalkotásánál, úgy vettük, hogy a felhasználó nem próbál meg irrationális dolgot véghez vinni vagy ha igen is ezek még az érintett szekvenciák futása előtt meghiúsulnak. Ez úgy értendő, hogy ha nincs nyersanyag nála, akkor nem próbálja azt aszteroidába elhelyezni, ha még is megpróbálná az már a lerakás előtt meghiúsul. minden hasonló eset ezen megállapodás szerint értendő. A robot, teleportkapu és bázis építéséhez kapcsolódó diagrammok esetében a megépítésükhoz szükséges nyersanyagok listáját tartalmazó receptek elemei külön nem kerültek feltüntetésre az inventory-val ellentétben. Ennek oka, hogy a diagrammok kezelhetetlen méretűvé nőttek volna, valamint a receptek konstansok, így a játék során tartalmuk végig állandó. A receptekben, ha fel lennének külön tüntetve a nyersanyag példányok ugyan azon típusok szerepelnének, mint jelen esetben az inventory-ban. Ez azért igaz, mivel minden esetben itt csak azon nyersanyagok lettek az inventory-ba helyezve, amelyek kellenek is az adott dolog elkészítéséhez.

A forgatókönyvek megalkotásánál, úgy vettük, hogy a felhasználó nem próbál meg irrationális dolgot véghez vinni vagy ha igen is ezek még az érintett szekvenciák futása előtt meghiúsulnak. Ez úgy értendő, hogy ha nincs nyersanyag nála, akkor nem próbálja azt aszteroidába elhelyezni, ha még is megpróbálná az már a lerakás előtt meghiúsul. minden hasonló eset ezen megállapodás szerint értendő.

4.2 A szkeleton modell valóságos use-case-ai

4.2.1 Use-case diagram



4.2.2 Use-case leírások

Use-case neve	Vas bányászása
Rövid leírás	A telepessel vasat bányászunk az aszteroidából.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> 1. Kiadjuk a parancsot, hogy bányásszunk a telepessel. 2. A telepes szól az aszteroidának, hogy vegye ki a vasat a magjából. 3. A telepes hozzáadja a kivett vasat az inventory-jához.
Alternatív forgatókönyv	2.A.1. Ha az aszteroida kérgének vastagsága nagyobb mint 0 akkor nem veszi ki a telepes a vasat.
Alternatív forgatókönyv	2.B.1. Ha a telepes inventory-ja teli van akkor sem veszi ki a vasat.

Use-case neve	Szén bányászása
Rövid leírás	A telepessel szenet bányászunk az aszteroidából.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> 1. Kiadjuk a parancsot, hogy bányásszunk a telepessel. 2. A telepes szól az aszteroidának, hogy vegye ki a szenét a magjából. 3. A telepes hozzáadja a kivett szenet az inventory-jához.
Alternatív forgatókönyv	2.A.1. Ha az aszteroida kérgének vastagsága nagyobb mint 0 akkor nem veszi ki a telepes a szenet.
Alternatív forgatókönyv	2.B.1. Ha a telepes inventory-ja teli van akkor sem veszi ki a szenet.

Use-case neve	Vízjég bányászása
Rövid leírás	A telepessel vízjeget bányászunk az aszteroidából.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> 1. Kiadjuk a parancsot, hogy bányásszunk a telepessel. 2. A telepes szól az aszteroidának, hogy vegye ki a vízjegét a magjából. 3. A telepes hozzáadja a kivett vízjeget az inventory-jához.
Alternatív forgatókönyv	2.A.1. Ha az aszteroida kérgének vastagsága nagyobb mint 0 akkor nem veszi ki a telepes a vízjeget.
Alternatív forgatókönyv	2.B.1. Ha a telepes inventory-ja teli van akkor sem veszi ki a vízjeget.

Use-case neve	Urán bányászása
Rövid leírás	A telepessel uránt bányászunk az aszteroidából.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> 1. Kiadjuk a parancsot, hogy bányásszunk a telepessel. 2. A telepes szól az aszteroidának, hogy vegye ki az uránját a magjából. 3. A telepes hozzáadja a kivett uránt az inventory-jához.
Alternatív forgatókönyv	2.A.1. Ha az aszteroida kérgének vastagsága nagyobb mint 0 akkor nem veszi ki a telepes az uránt.

Alternatív forgatókönyv	2.B.1. Ha a telepes inventory -ja teli van akkor sem veszi ki az uránt.
--------------------------------	---

Use-case neve	Vas visszahelyezése
Rövid leírás	A telepessel vasat helyezünk vissza az aszteroidába.
Aktorok	Felhasználó
Forgatókönyv	<p>1. Kiadjuk a parancsot, hogy helyezzünk vissza egy vasat az aszteroidába.</p> <p>2. Az aszteroidába bekerül egy vas s a telepes inventory -jából kikerül egy vas.</p>
Alternatív forgatókönyv	2.A.1. Ha az aszteroida nem üreges akkor nem történik semmi.

Use-case neve	Szén visszahelyezése
Rövid leírás	A a telepessel szenet helyezünk vissza az aszteroidába.
Aktorok	Felhasználó
Forgatókönyv	<p>1. Kiadjuk a parancsot, hogy helyezzünk vissza egy szenet az aszteroidába.</p> <p>2. Az aszteroidába bekerül egy szén s a telepes inventory -jából kikerül egy szén.</p>
Alternatív forgatókönyv	2.A.1. Ha az aszteroida nem üreges akkor nem történik semmi.

Use-case neve	Vízjég visszahelyezése
Rövid leírás	A a telepessel vízjeget helyezünk vissza az aszteroidába.
Aktorok	Felhasználó
Forgatókönyv	<p>1. Kiadjuk a parancsot, hogy helyezzünk vissza egy vízjeget az aszteroidába.</p> <p>2. Az aszteroidába bekerül egy vízjeg s a telepes inventory -jából kikerül egy vízjég.</p>
Alternatív forgatókönyv	2.A.1. Ha az aszteroida nem üreges akkor nem történik semmi.
Alternatív forgatókönyv	2.B.1. Ha az aszteroida napközelben van, akkor visszahelyezés után a vízjég rögtön elpárolog.

Use-case neve	Urán visszahelyezése
Rövid leírás	A telepessel uránt helyezünk vissza az aszteroidába.
Aktorok	Felhasználó
Forgatókönyv	<p>1. Kiadjuk a parancsot, hogy helyezzünk vissza egy uránt az aszteroidába.</p> <p>2. Az aszteroidába bekerül egy urán s a telepes inventory -jából kikerül egy urán.</p>
Alternatív forgatókönyv	2.A.1. Ha az aszteroida nem üreges akkor nem történik semmi.
Alternatív forgatókönyv	<p>2.B.1. Ha az aszteroida napközelben van, akkor egy belső számlálója növekszik. Ha a számláló elérte a 3-at, akkor a nyersanyag felrobban.</p> <p>2.B.2. Ha a nyersanyag felrobbant, az aszteroida elpusztul.</p>

	2.B.3. Az aszteroidán lévő összes entitások reagálnak az aszteroida megsemmisülésére. (A robotok kivételével az összes objektum megsemmisül.)
--	---

Use-case neve	Napvíhar történik
Rövid leírás	Az egyik nap napvihart idéz elő.
Aktorok	Felhasználó
Forgatókönyv	<p>1. A nap amelyik kitör szól a saját aszteroidáinak, hogy napvíhar történik.</p> <p>2. Az értesített aszteroidák minden rajtuk található entitást értesítenek a napviharról.</p> <p>3a. Ha nincs fedezékhely, az entitások megsemmisülnek.</p> <p>4. A teleportkapuk "napvíhar érte" állapotba kerülnek, ami hatására a további körökben véletleszerűen fognak mozogni az elérhető aszteroidák közt.</p> <p>Alternatív forgatókönyv</p> <p>3b. Ha az aszteroida üreges és kérgének vastagsága 0 akkor a rajta található entitások megmenekülnek. (A kapu ilyenkor is meghibásodik.)</p>
Alternatív forgatókönyv	3.B. Ha az aszteroida üreges, s kérgének vastagsága 0 akkor nem csinál semmit a rajta található entitásokkal.

Use-case neve	Fúrás robottal, elpárolog
Rövid leírás	Egy olyan aszteroidát fúrunk robottal, melynek belsejében olyan anyag van, ami elpárolgatható.
Aktorok	Felhasználó
Forgatókönyv	<p>1. Kiadjuk a parancsot, hogy a robot fúrjon.</p> <p>2. Az aszteroida kérgének vastagsága csökken 1-el.</p>
Alternatív forgatókönyv	<p>2.A.1. Ha az aszteroida vastagsága 1 akkor 0-ra csökken.</p> <p>2.A.2. Ha az aszteroida napközelben van akkor a benne található nyersanyag elpárolog.</p>
Alternatív forgatókönyv	2.B.1. Ha az aszteroida vastagsága 0 akkor nem történik semmi.

Use-case neve	Fúrás telepessel, elpárolog
Rövid leírás	Egy olyan aszteroidát fúrunk telepessel melynek belsejében olyan anyag van ami elpárolgatható.
Aktorok	Felhasználó
Forgatókönyv	<p>1. Kiadjuk a parancsot, hogy a telepes fúrjon.</p> <p>2. Az aszteroida kérgének vastagsága csökken 1-el.</p>
Alternatív forgatókönyv	<p>2.A.1. Ha az aszteroida vastagsága 1 akkor 0-ra csökken.</p> <p>2.A.2. Ha az aszteroida napközelben van akkor a benne található nyersanyag elpárolog.</p>
Alternatív forgatókönyv	2.B.1. Ha az aszteroida vastagsága 0 akkor nem történik semmi.

Use-case neve	Fúrás robottal, robban, van szomszéd
----------------------	--------------------------------------

Rövid leírás	Egy olyan aszteroidát fúrunk robottal melynek belsejében olyan anyag van ami felrobbantható, ezen kívül az aszteroidának van szomszédja.
Aktorok	Felhasználó
Forgatókönyv	1. Kiadjuk a parancsot, hogy a robot fúrjon. 2. Az aszteroida kérgének vastagsága csökken 1 -el.
Alternatív forgatókönyv	2.B.1. Ha az aszteroida napközelben van és a kérge átfúrt, akkor egy belső számlálója növekszik. Ha a számláló elérte a 3-at, akkor a nyersanyag felrobban. 2.B.2. Ha a nyersanyag felrobbant, az aszteroida elpusztul. 2.B.3. Az aszteroidán lévő robot egy szomszédos aszteroidára lép.

Use-case neve	Fúrás robottal, robban, nincs szomszéd
Rövid leírás	Egy olyan aszteroidát fúrunk robottal melynek belsejében olyan anyag van ami felrobbantható, ezen kívül az aszteroidának nincsen szomszédja.
Aktorok	Felhasználó
Forgatókönyv	1. Kiadjuk a parancsot, hogy a robot fúrjon. 2. Az aszteroida kérgének vastagsága csökken 1 -el.
Alternatív forgatókönyv	2.B.1. Ha az aszteroida napközelben van és a kérge átfúrt, akkor egy belső számlálója növekszik. Ha a számláló elérte a 3-at, akkor a nyersanyag felrobban. 2.B.2. Ha a nyersanyag felrobbant, az aszteroida elpusztul. 2.B.3. Az aszteroidán lévő robot megsemmisül, mert nincs szomszédos aszteroida.
Alternatív forgatókönyv	2.B.1. Ha az aszteroida vastagsága 0 akkor nem történik semmi.

Use-case neve	Fúrás telepessel, robban
Rövid leírás	Egy olyan aszteroidát fúrunk telepessel melynek belsejében olyan anyag van ami felrobbantható.
Aktorok	Felhasználó
Forgatókönyv	1. Kiadjuk a parancsot, hogy a telepes fúrjon. 2. Az aszteroida kérgének vastagsága csökken 1 -el.
Alternatív forgatókönyv	2.B.1. Ha az aszteroida napközelben van és a kérge átfúrt, akkor egy belső számlálója növekszik. Ha a számláló elérte a 3-at, akkor a nyersanyag felrobban. 2.B.2. Ha a nyersanyag felrobbant, az aszteroida elpusztul. 2.B.3. Az aszteroidán lévő telepes megsemmisül.
Alternatív forgatókönyv	2.B.1. Ha az aszteroida vastagsága 0 akkor nem történik semmi.

Use-case neve	Fúrás robottal
Rövid leírás	Egy robottal fúrunk egy aszteroidát.
Aktorok	Felhasználó
Forgatókönyv	1. Kiadjuk a parancsot, hogy a robot fúrjon. 2. Az aszteroida kérgének vastagsága csökken 1 -el.

Alternatív forgatókönyv	2.A.1. Ha az aszteroida vastagsága 0 akkor nem történik semmi.
--------------------------------	--

Use-case neve	Fúrás telepessel
Rövid leírás	Egy telepessel fúrunk egy aszteroidát.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> Kiadjuk a parancsot, hogy a telep fúrjon. Az aszteroida kérgének vastagsága csökken 1 -el.
Alternatív forgatókönyv	2.A.1. Ha az aszteroida vastagsága 0 akkor nem történik semmi.

Use-case neve	Mozgás robottal
Rövid leírás	Mozog egy robot.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> A robotnak kiadjuk a mozgás parancsot. A robot átmegegy a célaszteroidára s elmegegy az eddigi aszteroidájáról.

Use-case neve	Mozgás telepessel
Rövid leírás	Mozog egy telepes.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> A telepesnek kiadjuk a mozgás parancsot. A telepes átmegegy a célaszteroidára s elmegegy az eddigi aszteroidájáról.

Use-case neve	Teleportkapu lehelyezése, pár egy másik aszteroidán.
Rövid leírás	Egy telepes lehelyez egy teleportkaput, úgy, hogy a kapu párja már le van helyezve egy aszteroidán.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> A telepesnek kiadjuk a parancsot, hogy helyezze le a teleportkaput. A telepes kiveszi a teleportkaput az inventoryjából, s rárakja az aszteroidájára. A teleportkapu megnézi, hogy melyik aszteroidán van a vele összekötött teleportkapu A teleportkapu megmondja a két megfelelő aszteroidának, hogy mostmár szomszédosak.

Use-case neve	Teleportkapu lehelyezése, pár a telepes inventory-jában.
Rövid leírás	Egy telepes lehelyez egy teleportkaput, úgy hogy a párja még a telepes hátizsákjában van.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> A telepesnek kiadjuk a parancsot, hogy helyezze le a teleportkaput. A telepes kiveszi a teleportkaput az inventory-jából, s rárakja az aszteroidájára.

Use-case neve	Teleportkapu lehelyezése, pár már elpusztult.
----------------------	---

Rövid leírás	Egy telepes lehelyez egy teleportkaput, úgy hogy a teleportkapu párja már elpusztult.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> 1. A telepesnek kiadjuk a parancsot, hogy helyezze le a teleportkaput. 2. A telepes kiveszi a teleportkaput az inventoryjából, s rárakja az aszteroidájára.

Use-case neve	Teleportkapu építése
Rövid leírás	Egy telepessel építünk egy teleportkaput.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> 1. Kiadjuk a telepesnek a parancsot, hogy építsen meg egy teleportkaput. 2. Kivesszük a szükséges nyersanyagokat a telepes inventoryjából. 3. Hozzáadunk egy teleportkapu párt a telepes inventoryjához
Alternatív forgatókönyv	2.A.1. Ha a telepesnek nincsenek meg a megfelelő nyersanyagai akkor nem történik semmi.

Use-case neve	Bázis építése
Rövid leírás	Egy telepessel építünk egy bázist.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> 1. Kiadjuk a telepesnek a parancsot, hogy építsen meg egy bázist. 2. Kivesszük a szükséges nyersanyagokat a telepes inventoryjából. 3. Ráhelyezünk a telepes aszteroidájára egy bázist. 4. Szólunk a game osztálynak, hogy a játékos megyerte a játékot.
Alternatív forgatókönyv	2.A.1. Ha a telepesnek nincsenek meg a megfelelő nyersanyagai akkor nem történik semmi.

Use-case neve	Robot építése
Rövid leírás	Egy telepessel építünk egy robotot.
Aktorok	Felhasználó
Forgatókönyv	<ol style="list-style-type: none"> 1. Kiadjuk a telepesnek a parancsot, hogy építsen meg egy robotot. 2. Kivesszük a szükséges nyersanyagokat a telepes inventoryjából. 3. Ráhelyezünk a telepes aszteroidájára egy robotot.
Alternatív forgatókönyv	2.A.1. Ha a telepesnek nincsenek meg a megfelelő nyersanyagai akkor nem történik semmi.

4.3 A szkeleton kezelői felületének terve, dialógusok

A kezelőfelület konzolos alkalmazásként kerül megvalósításra. A dokumentumban feltüntetett szekvencia és kommunikációs diagrammoknak páronként megfelelhetők a kezelőfelületen található menü pontok. A több lefutásból álló esetekben egy gyűjtő menüpont alá lettek sorolva a különálló, de hasonló témakörű esetek.

A kezelőfelületen lévő menüpontok között az előttünk lévő szám beírásával és enter ütésével lehet választani. Azon lefutások során, ahol választani lehet esetek közül szintén számok leütésével lehet az esetek között a menühöz hasonlóan választani. Egyes esetekben számot is megadható paraméterként.

4.3.1 Főmenük

A főmenü tervezett kinézete.

4.3.2 Almenük

Az almenük tervezett kinézete.

Aszteroida bányászat

1. Bányászás
2. Visszahelyezés
3. Napkitörés
4. Fúrás
5. Mozgás
6. Teleportkapu lehelyezése
7. Építés

Aszteroida bányászat Bányászás

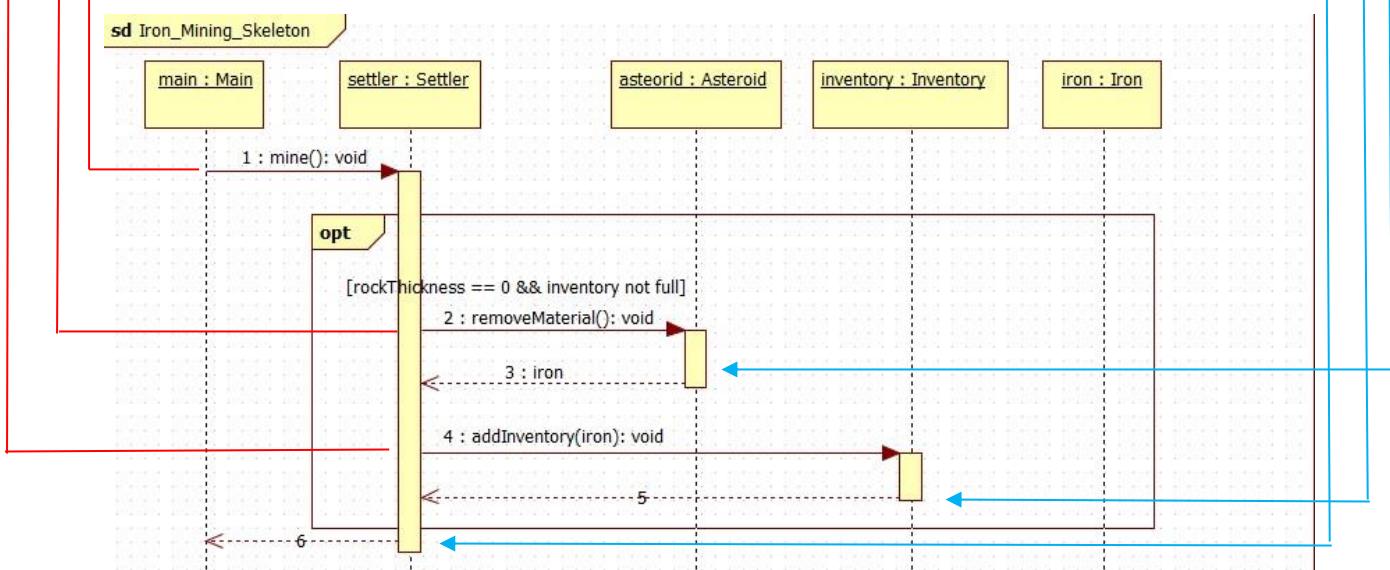
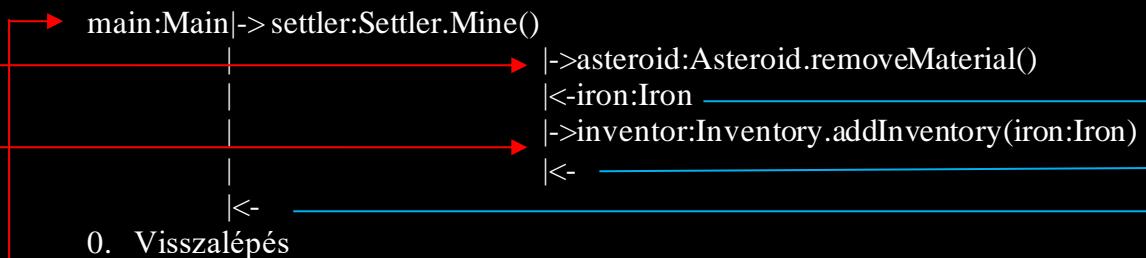
1. Vas bányászása
2. Urán bányászása
3. Szén bányászása
4. Vízjég bányászása
0. Visszalépés

4.3.3 Kimenet formátuma, értelmezése

Aszteroida bányászat Vas bányászása

Nincs beállítható egyéni paraméter.

Lefutás:

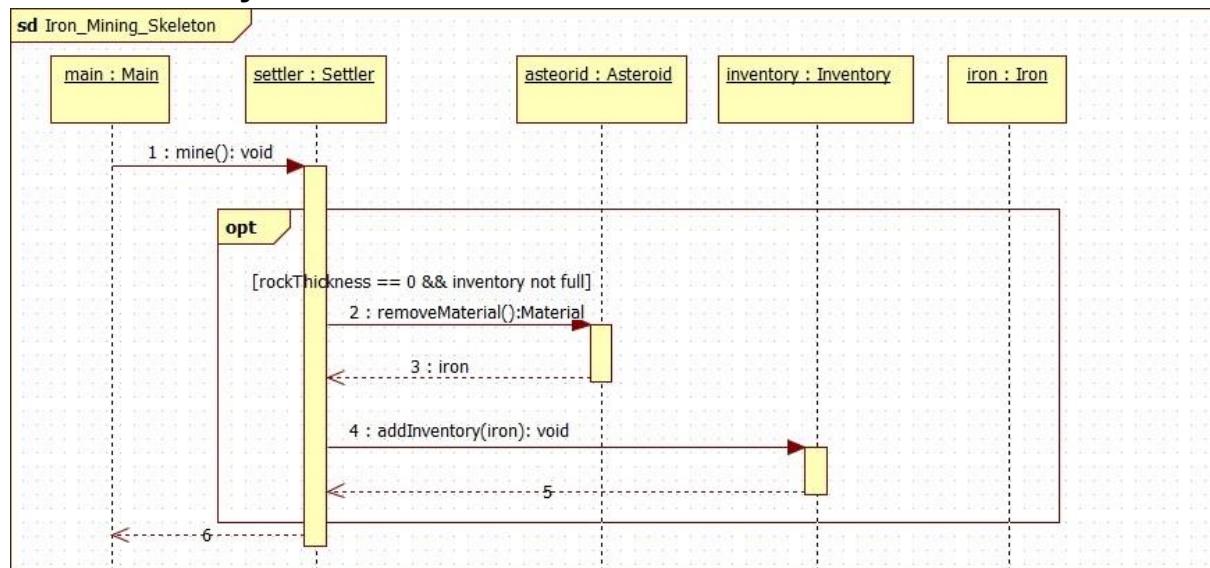


A kimeneti formátum tervezete és értelmezése.

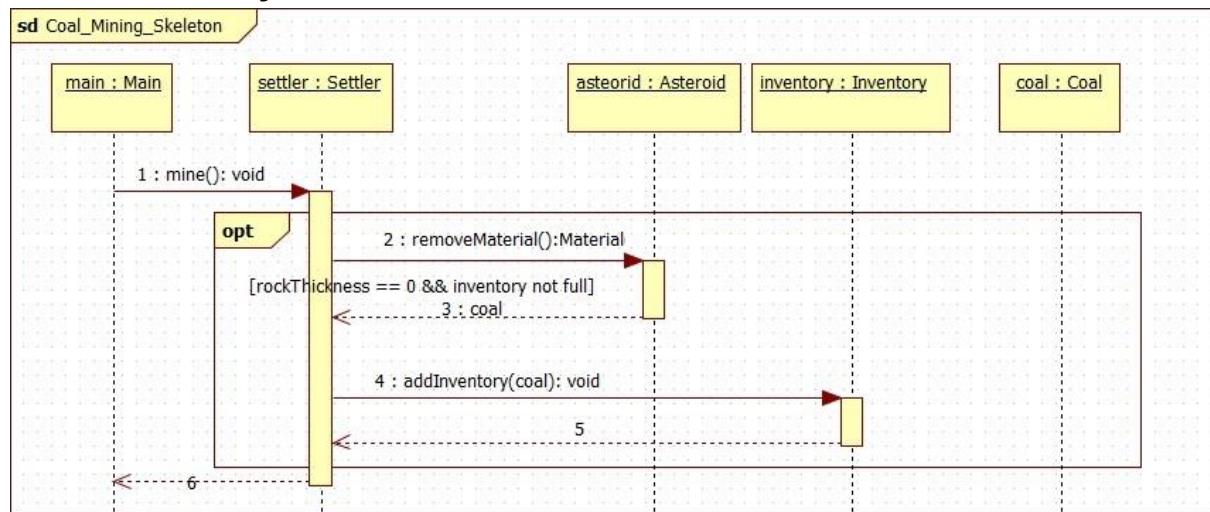
Jelölések a szekvenciával való egyeztetéshez:
Piros nyíl: függvényhívás
Kék nyíl: visszatérés

4.4 Szekvencia diagramok a belső működésre

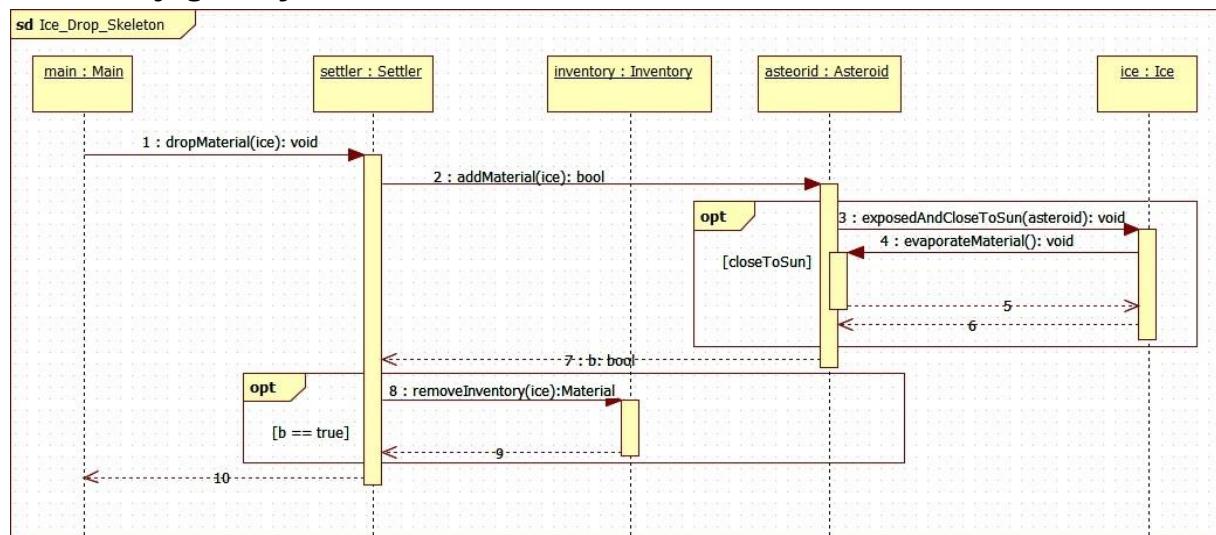
4.4.1 Vas bányászása



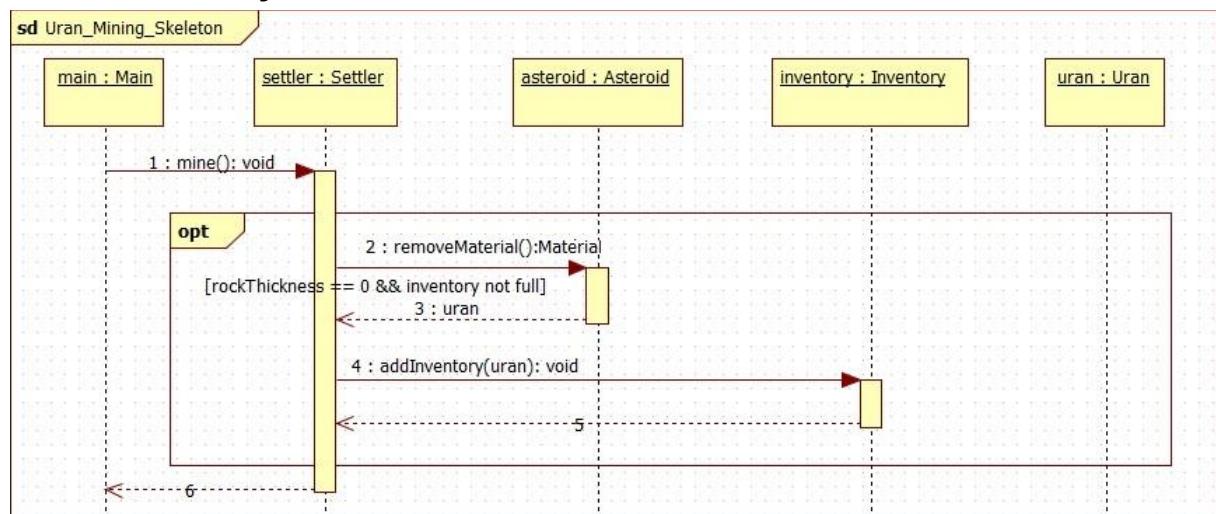
4.4.2 Szén bányászása



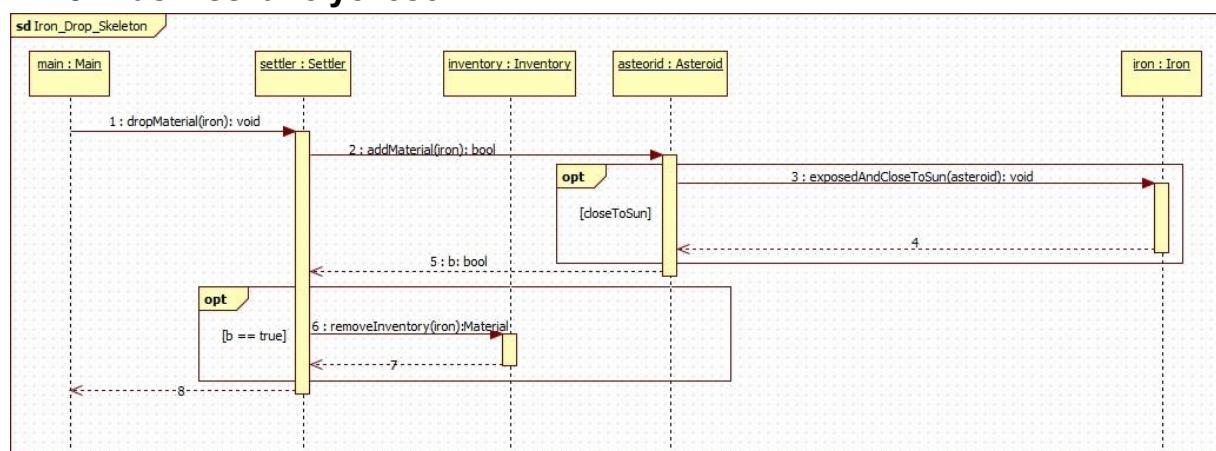
4.4.3 Vízjég bányászása



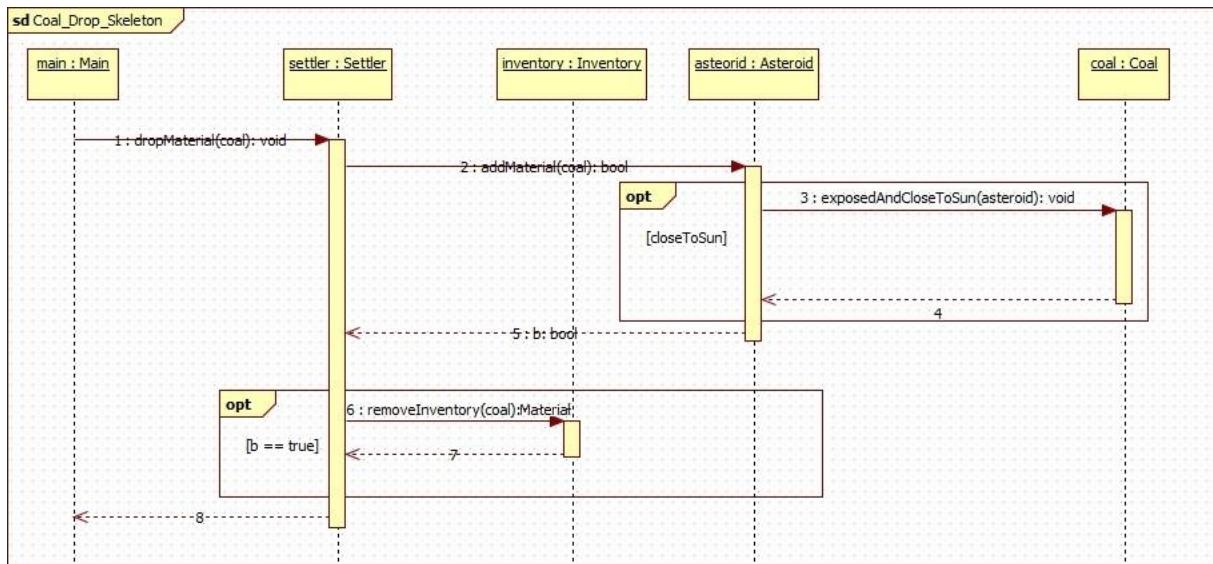
4.4.4 Urán bányászása



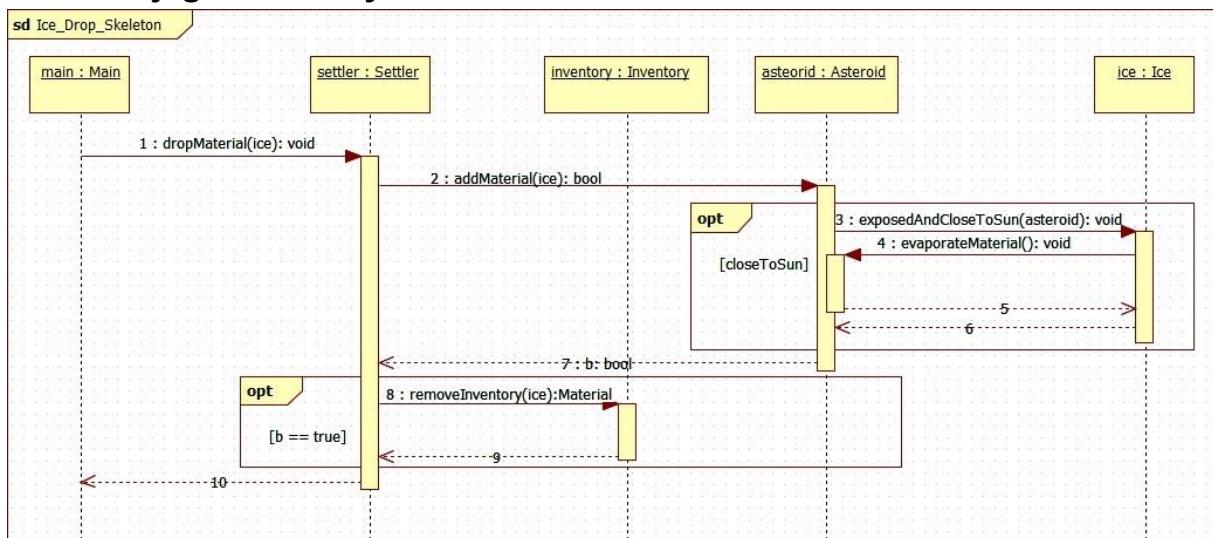
4.4.5 Vas visszahelyezése



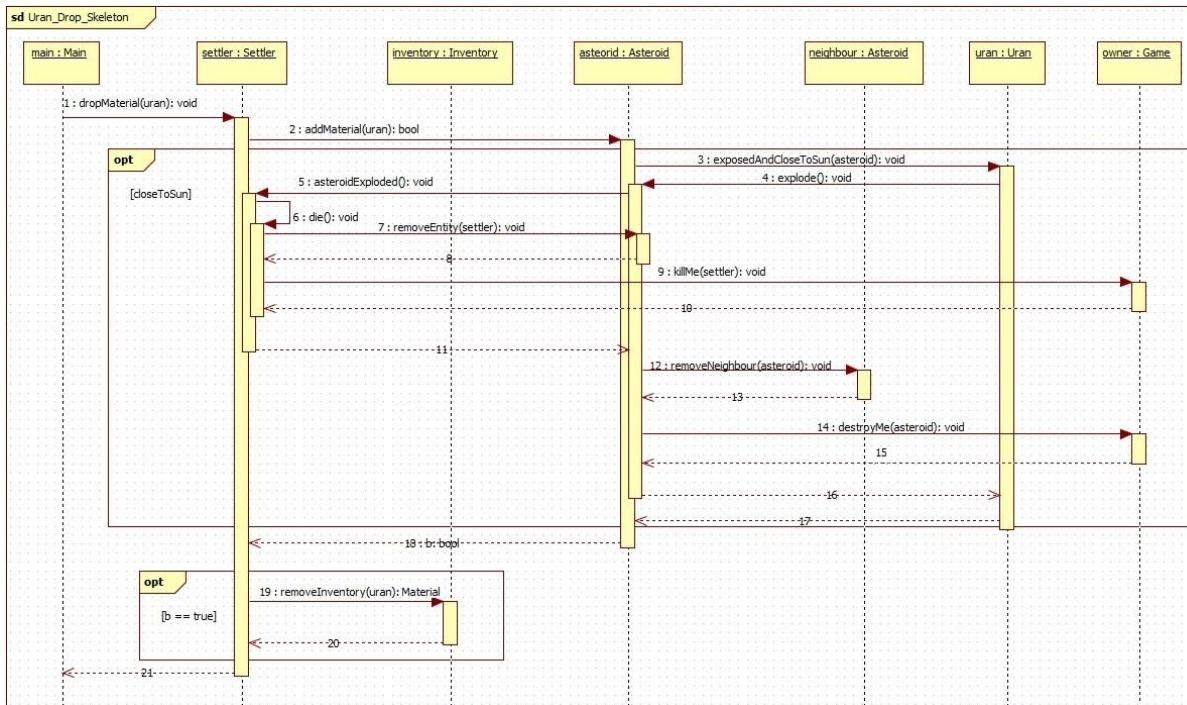
4.4.6 Szén visszahelyezése



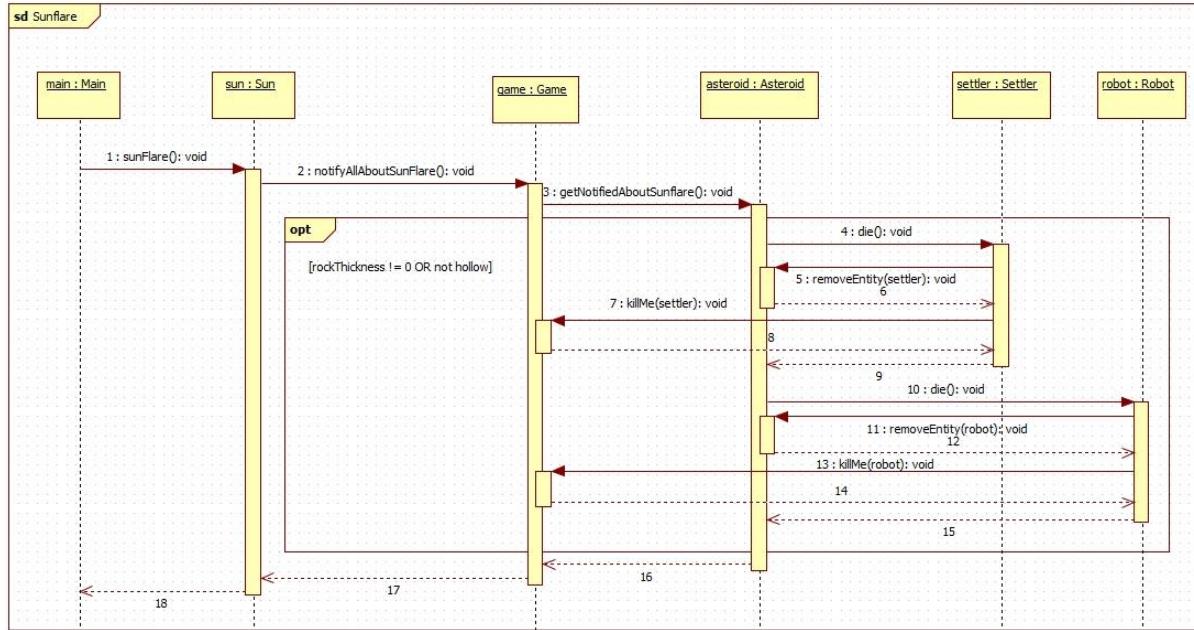
4.4.7 Vízjég visszahelyezése



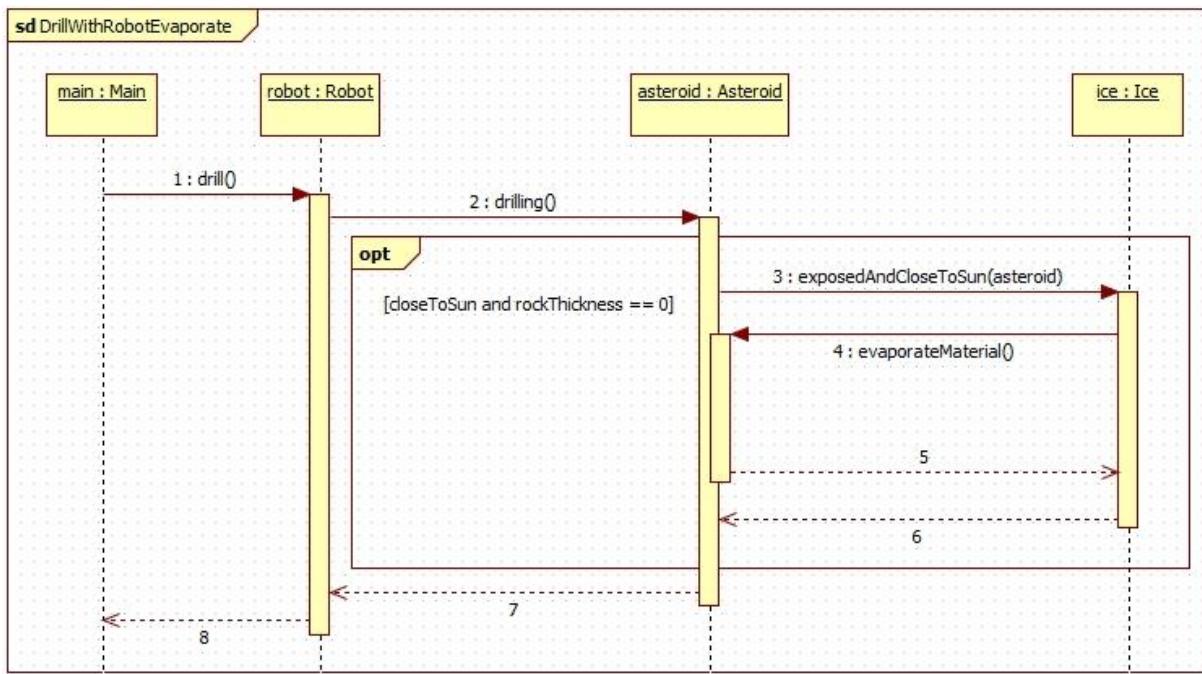
4.4.8 Urán visszahelyezése



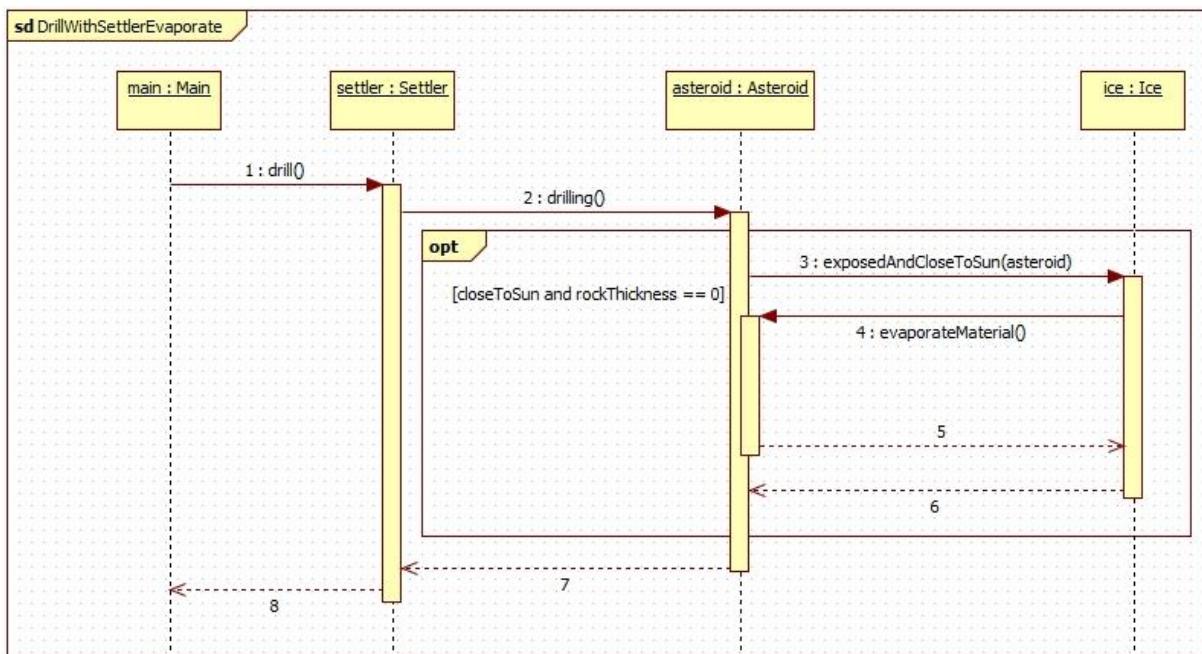
4.4.9 Napvihar történik



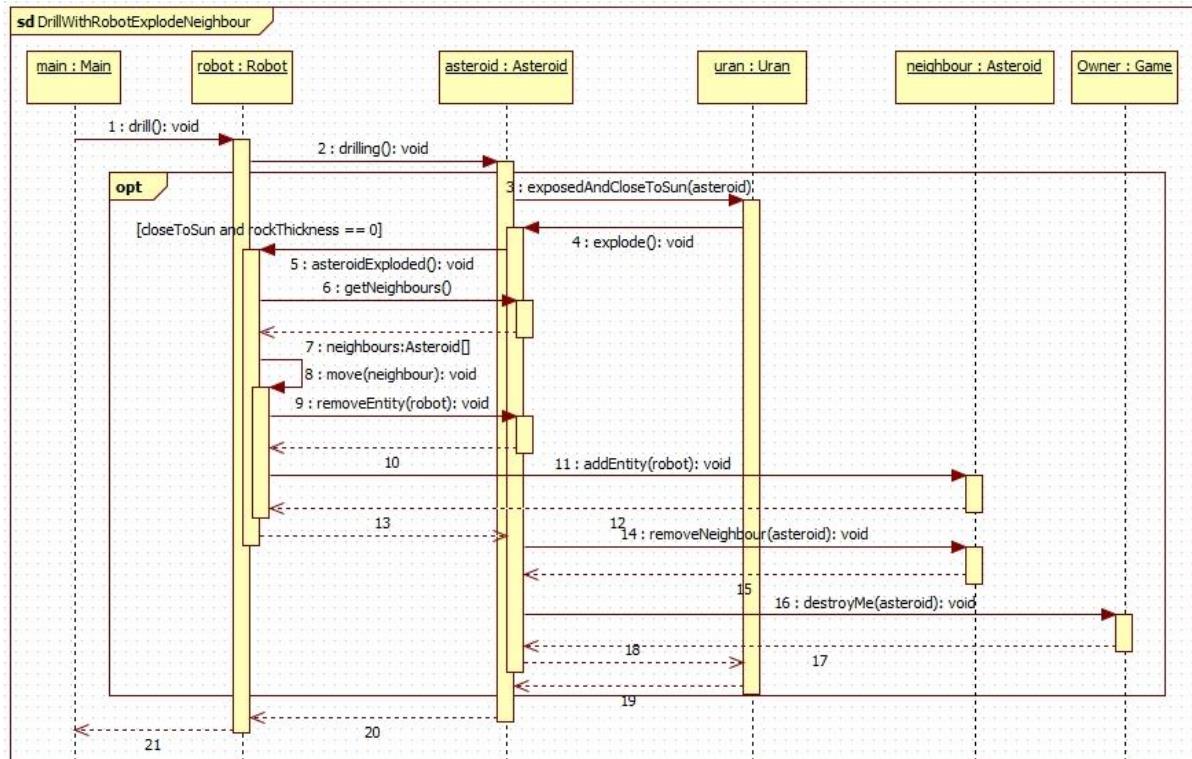
4.4.10 Fúrás robottal, elpárolog



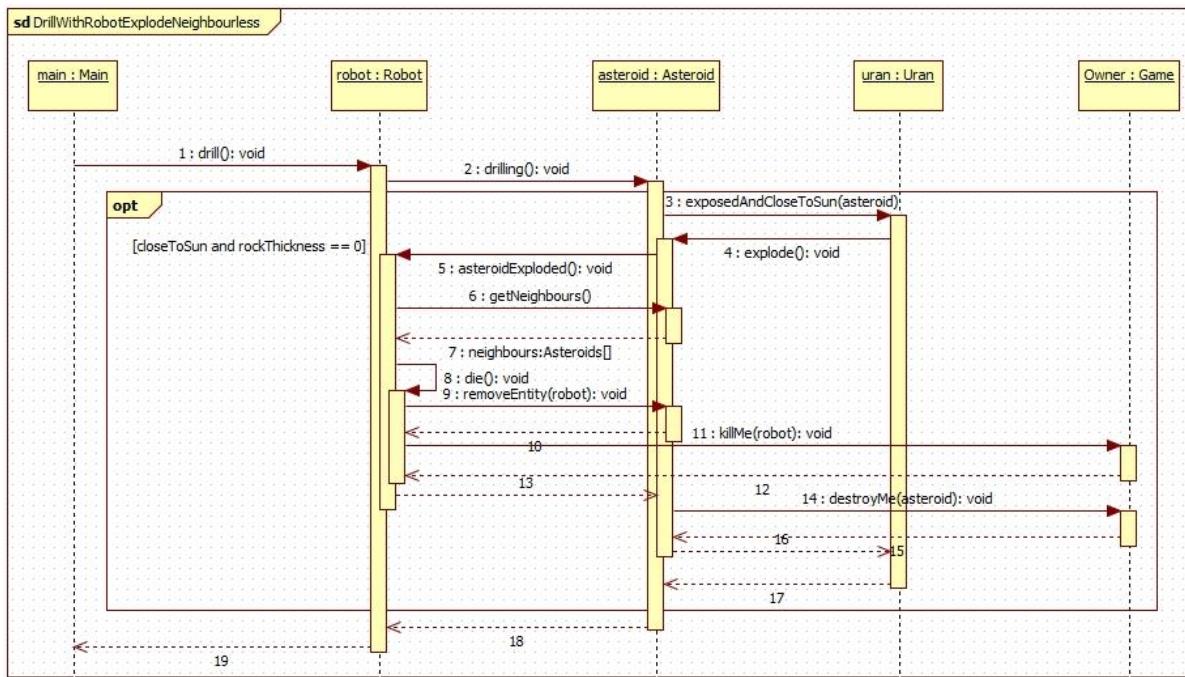
4.4.11 Fúrás telepessel, elpárolog



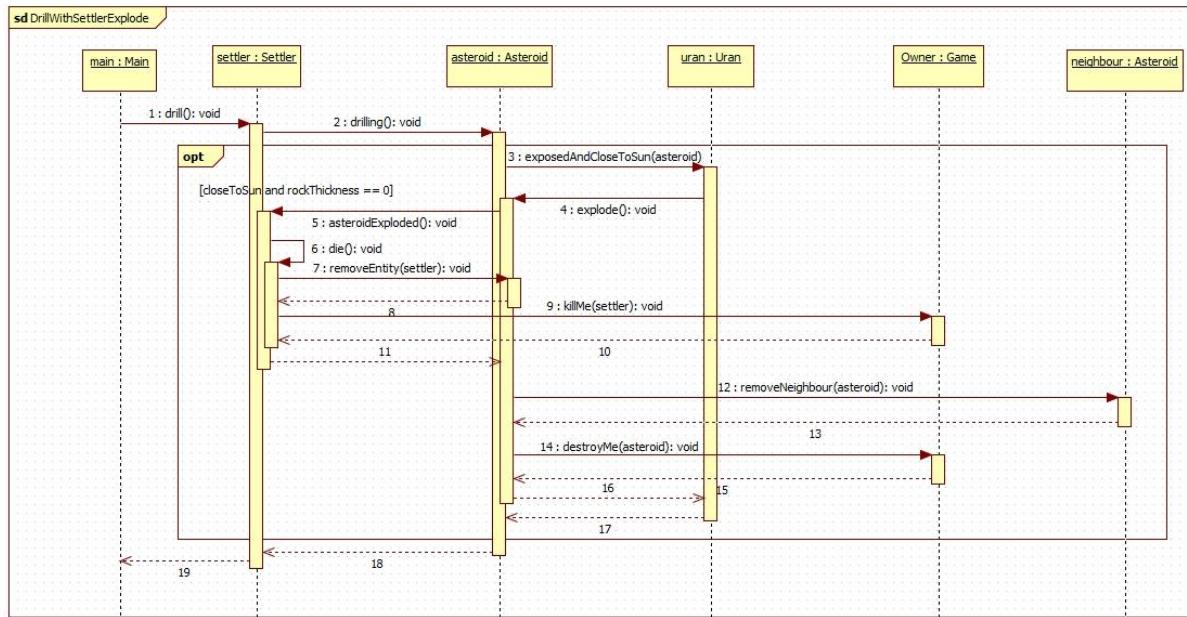
4.4.12 Fúrás robottal, robban, van szomszéd



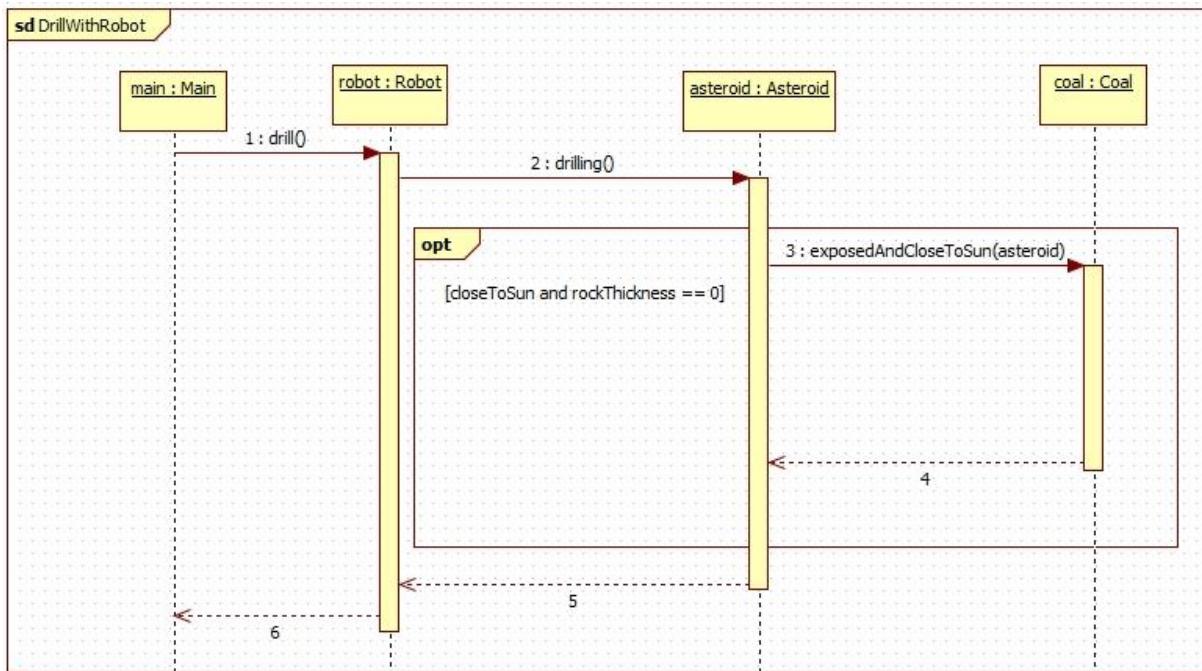
4.4.13 Fúrás robottal, robban, nincs szomszéd



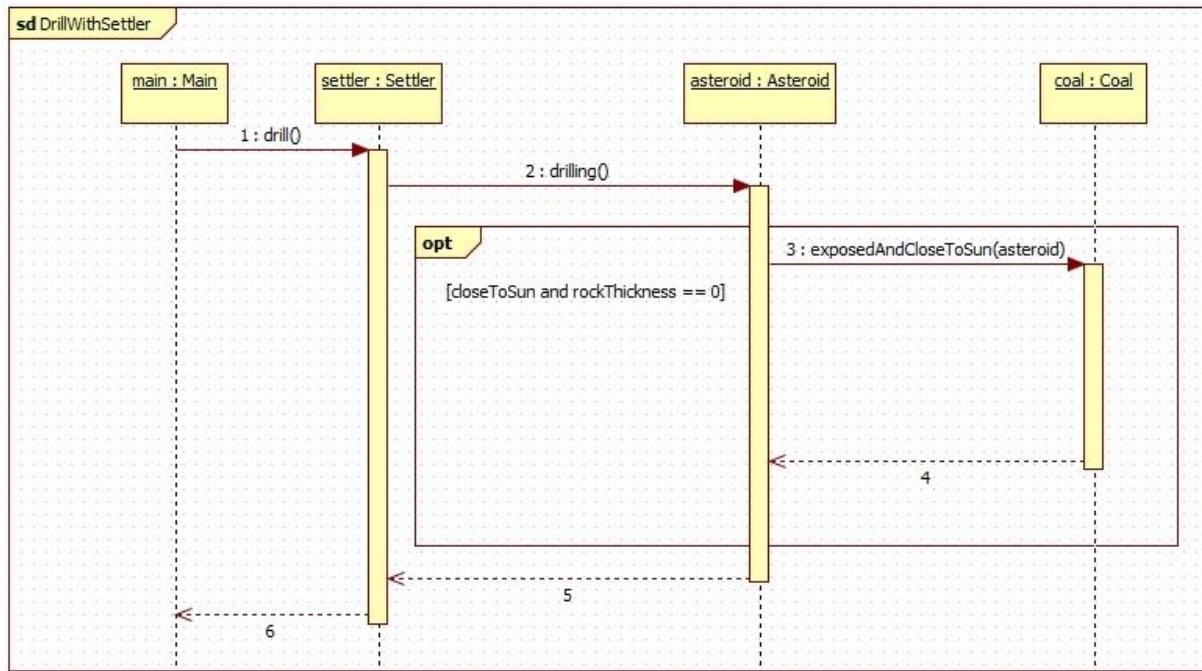
4.4.14 Fúrás telepessel, robban



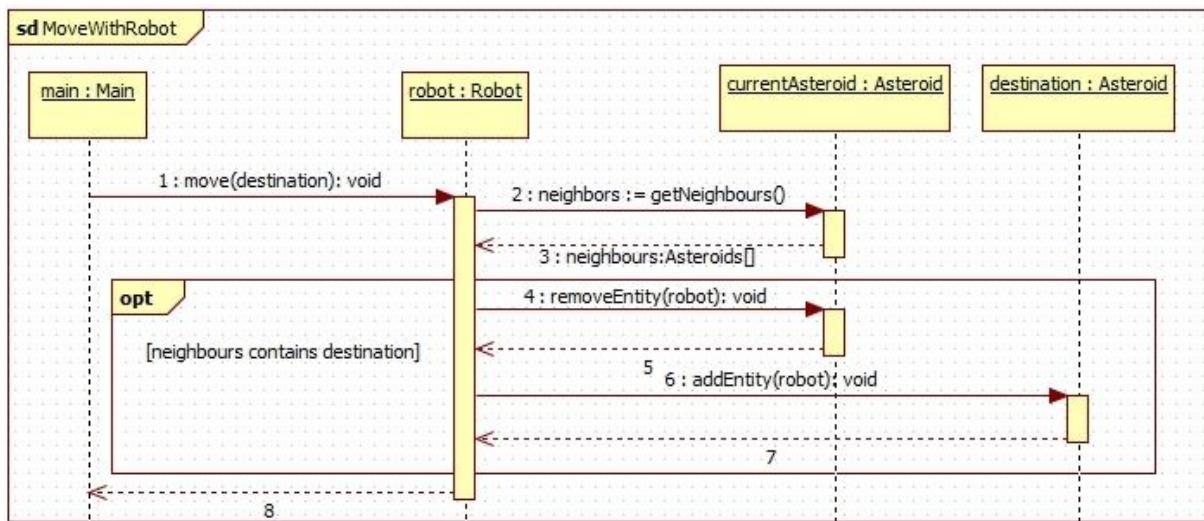
4.4.15 Fúrás robottal



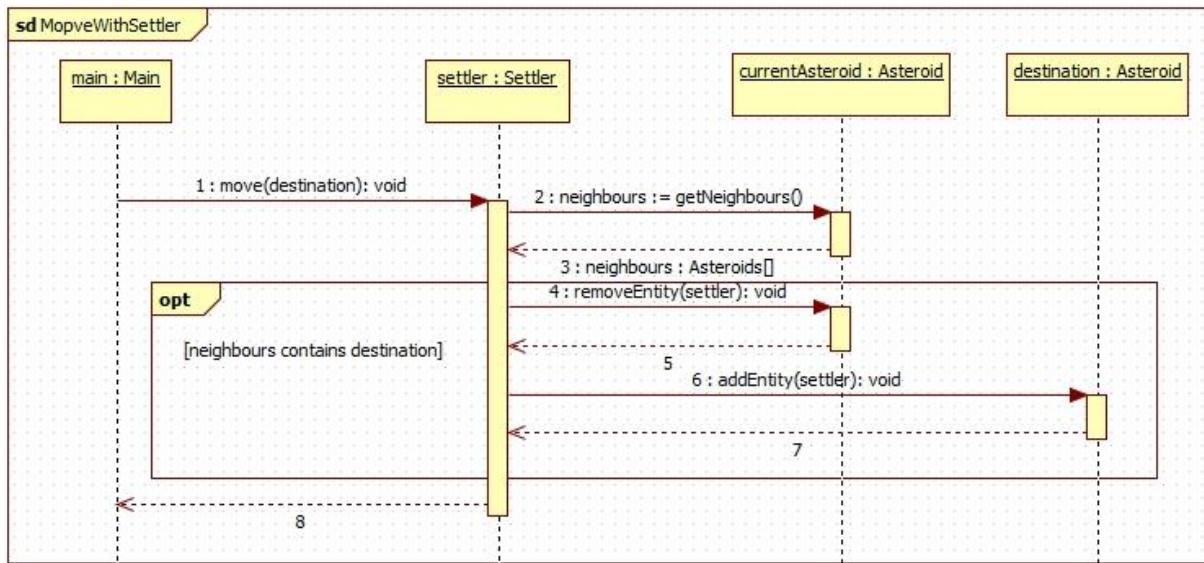
4.4.16 Fúrás telepessel



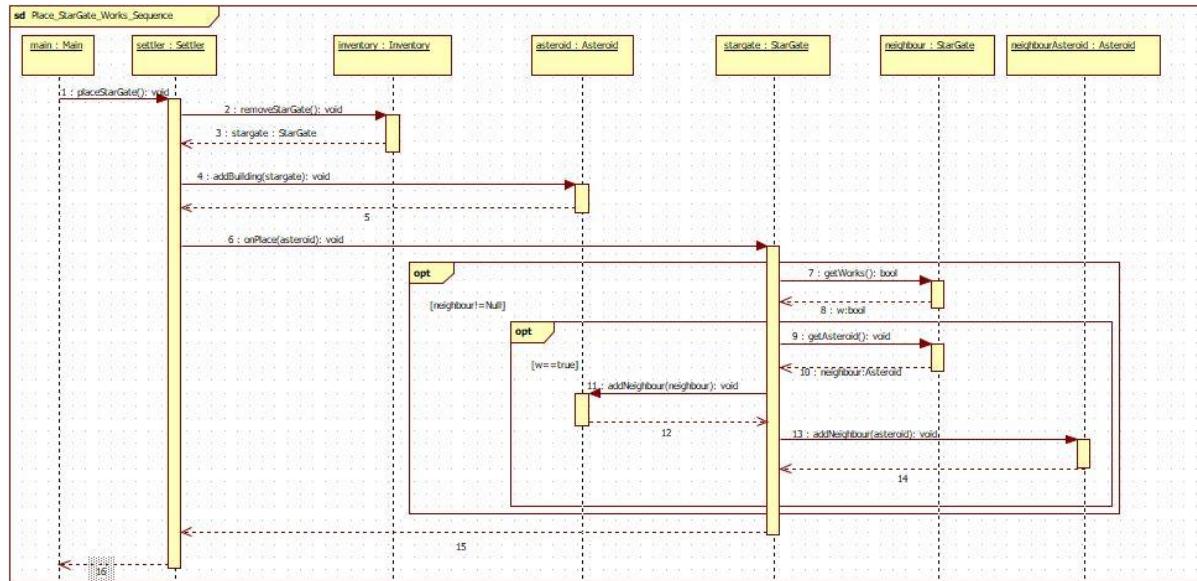
4.4.17 Mozgás robottal



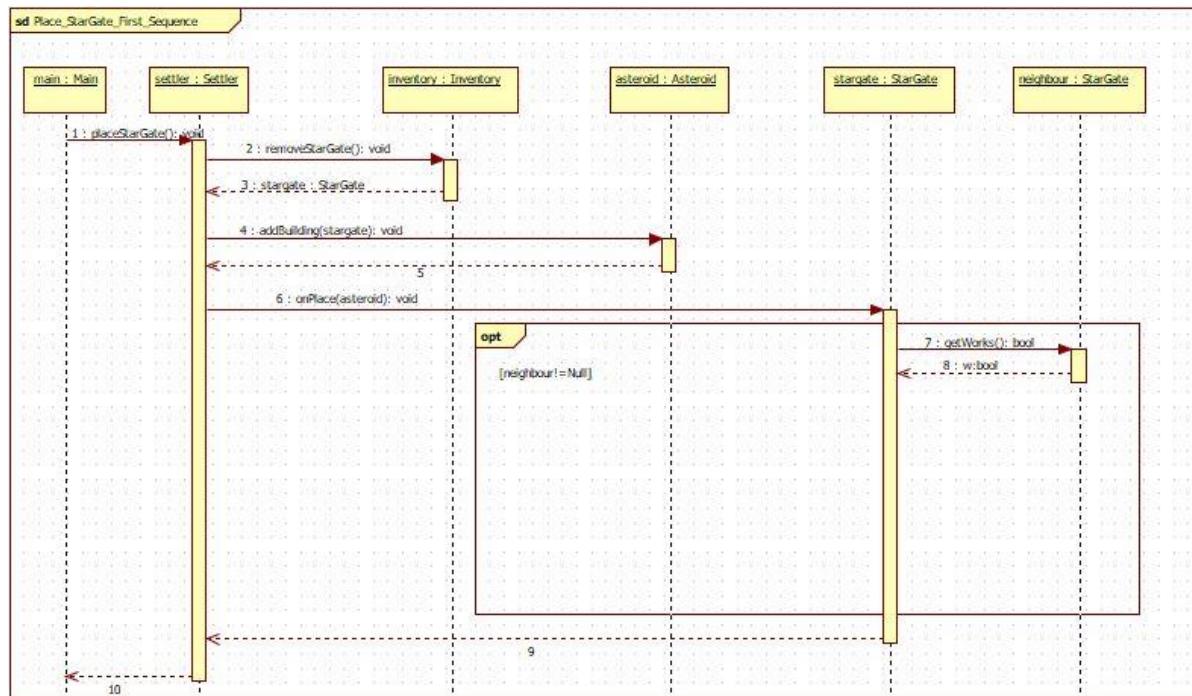
4.4.18 Mozgás telepessel



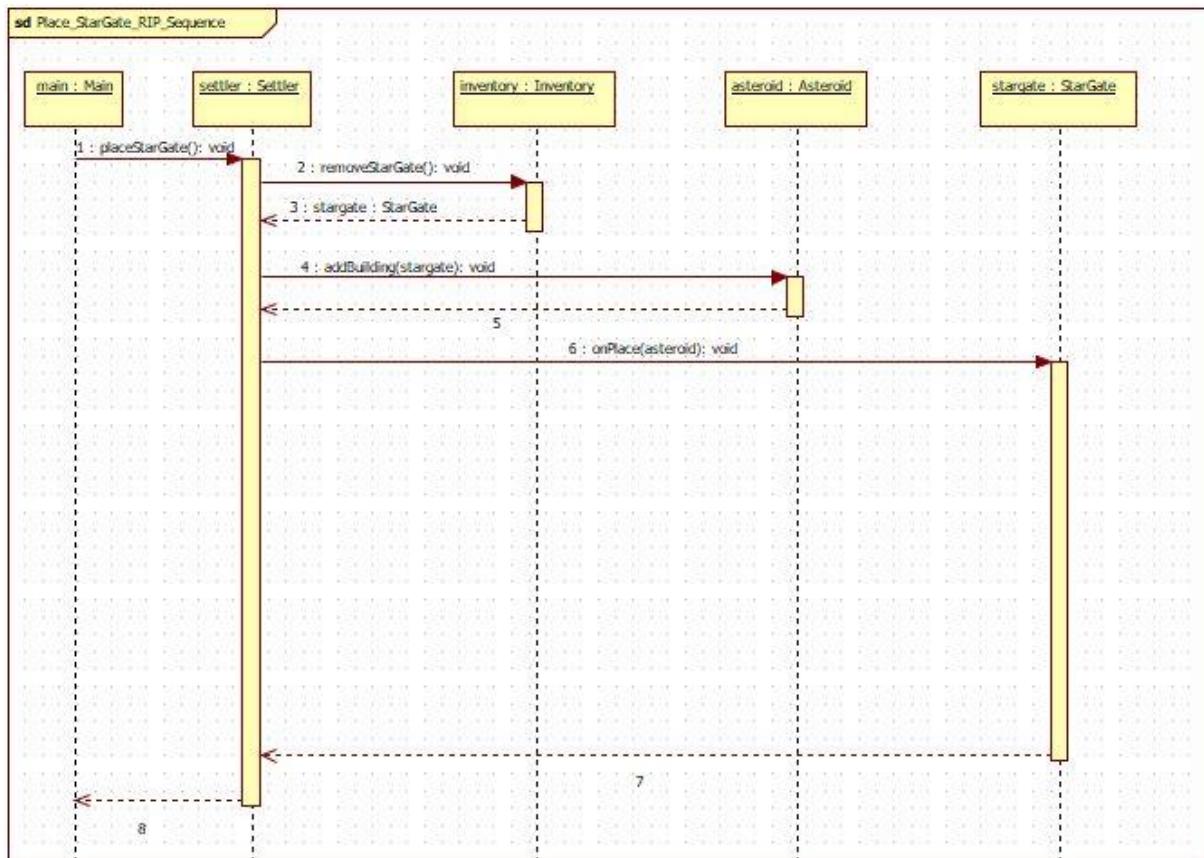
4.4.19 Teleportkapu lehelyezése, pár egy másik aszteroidán.



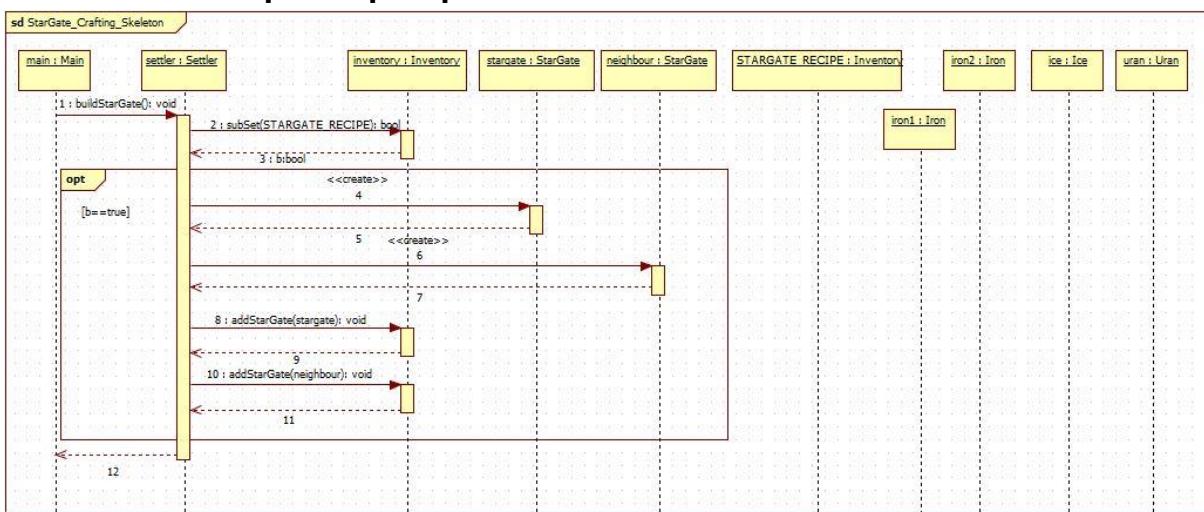
4.4.20 Teleportkapu lehelyezése, pár a telepes inventory-jában.



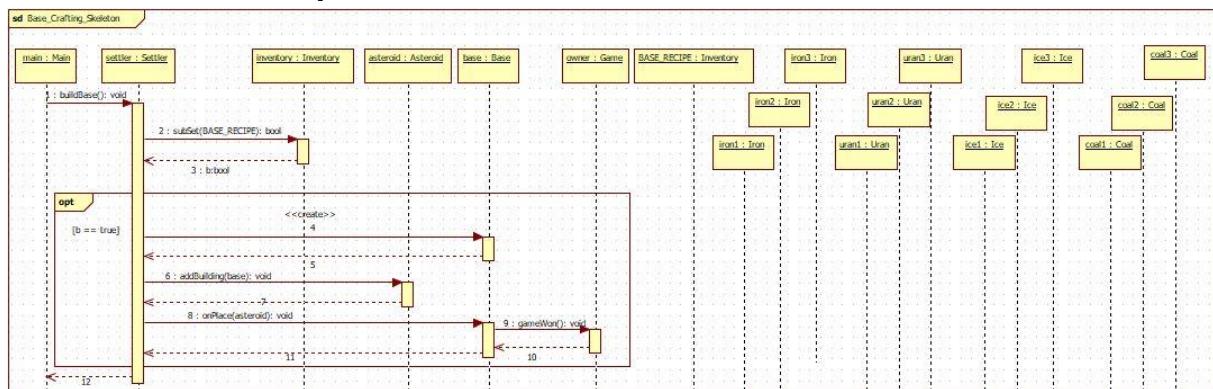
4.4.21 Teleportkapu lehelyezése, pár már elpusztult.



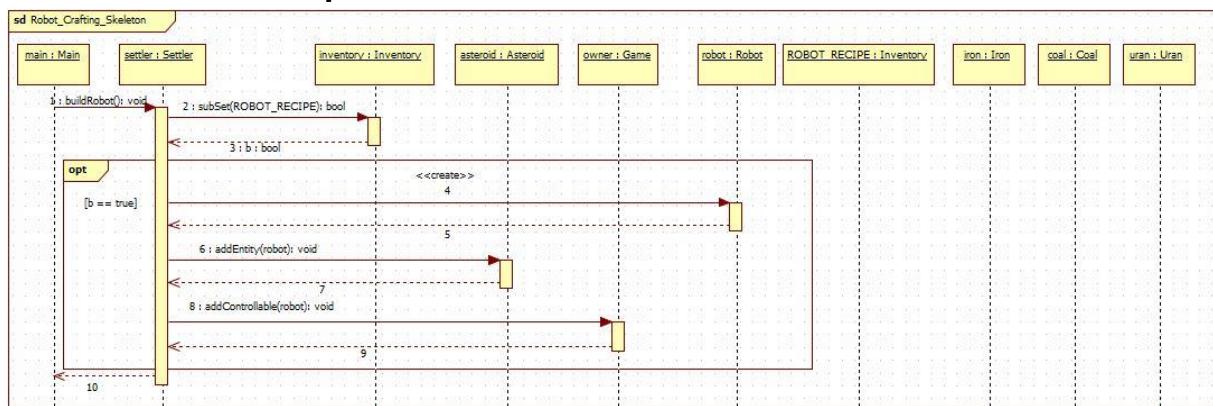
4.4.22 Teleportkapu építése



4.4.23 Bázis építése

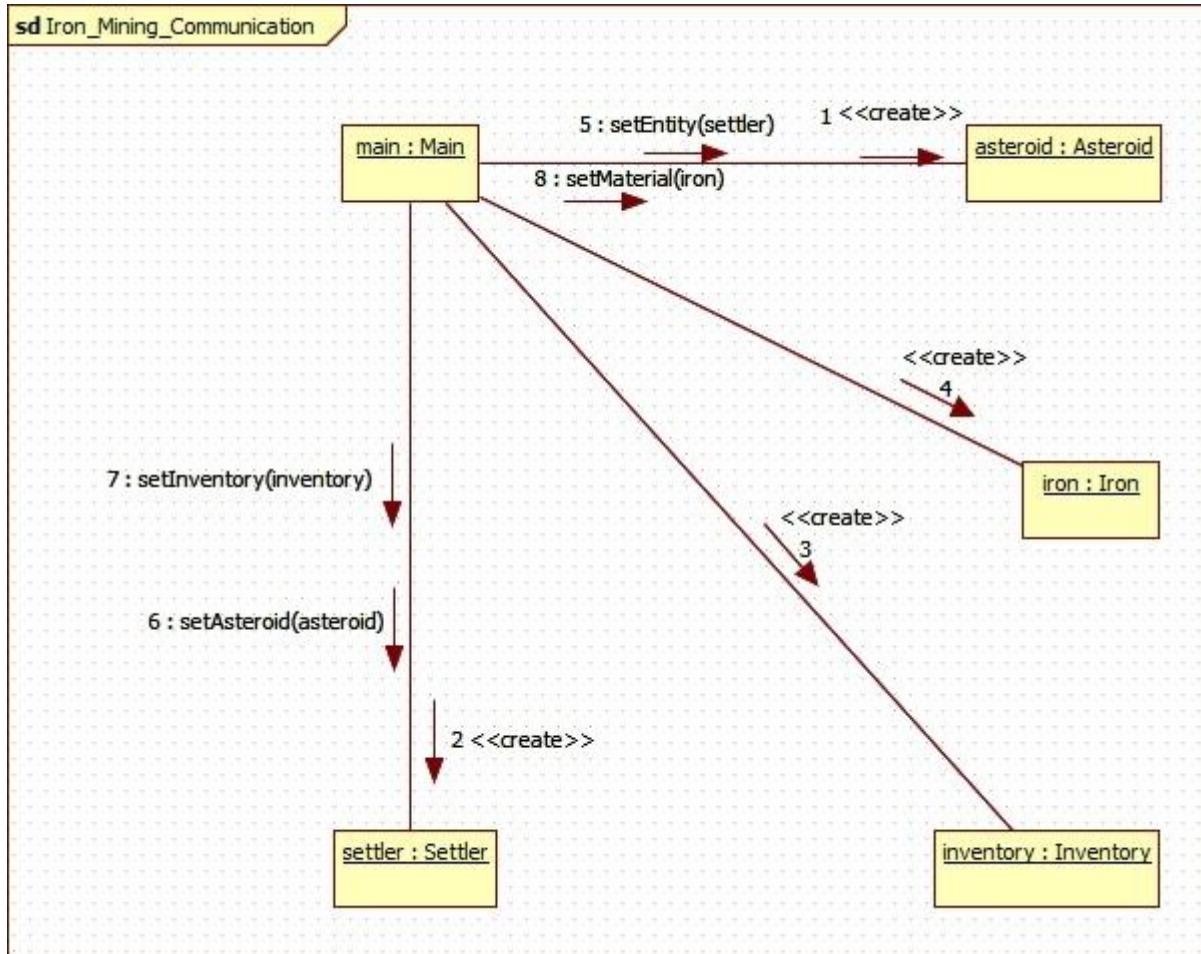


4.4.24 Robot építése

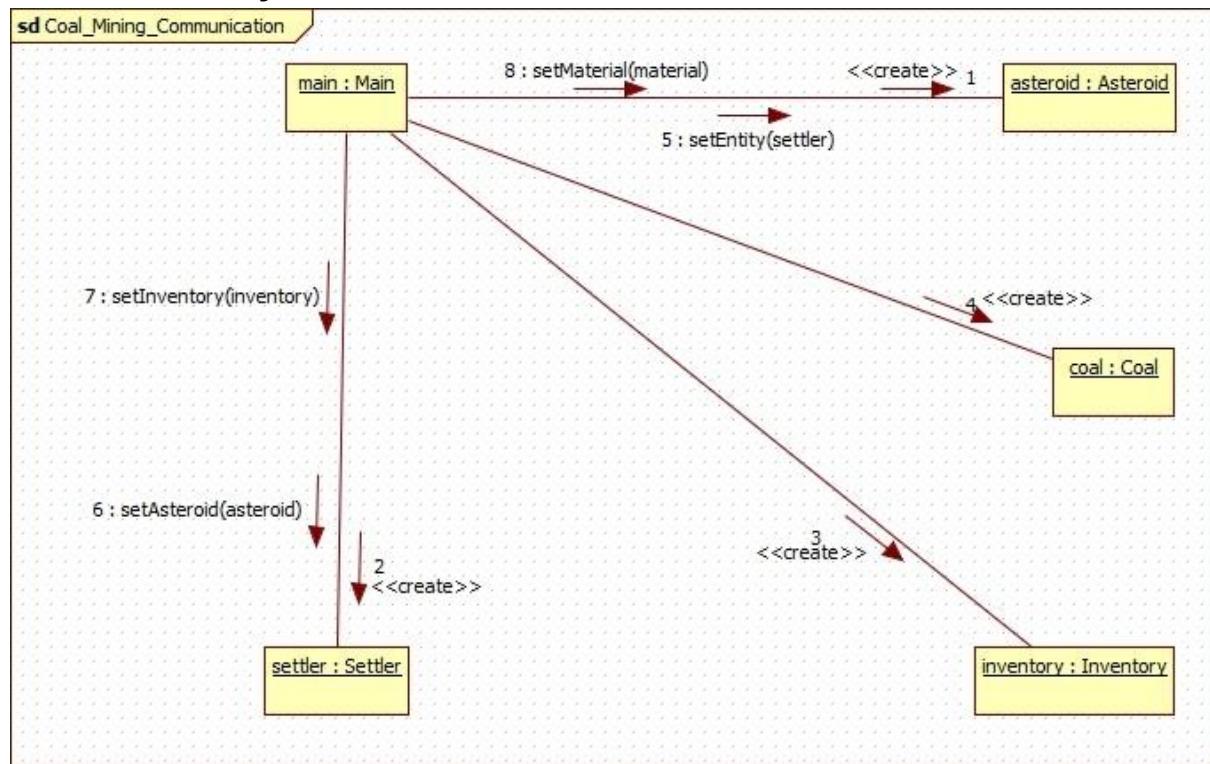


4.5 Kommunikációs diagramok

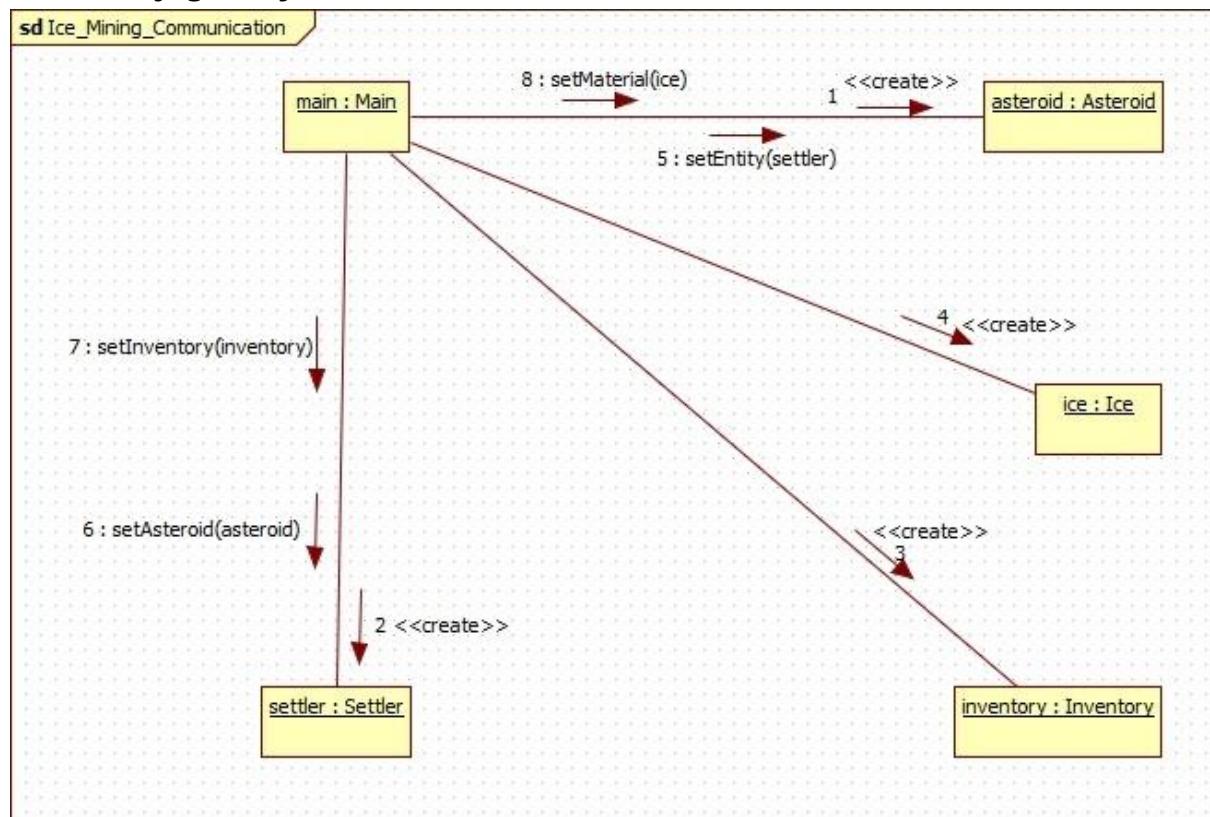
4.5.1 Vas bányászása



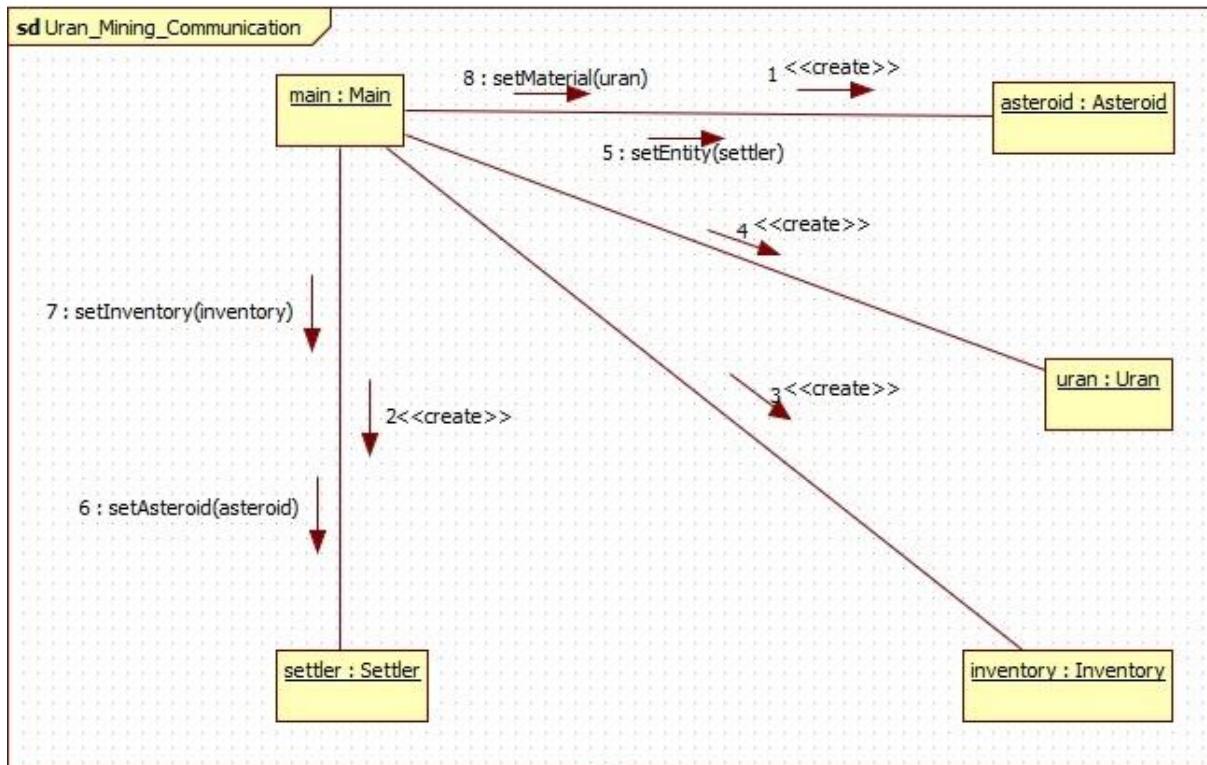
4.5.2 Szén bányászása



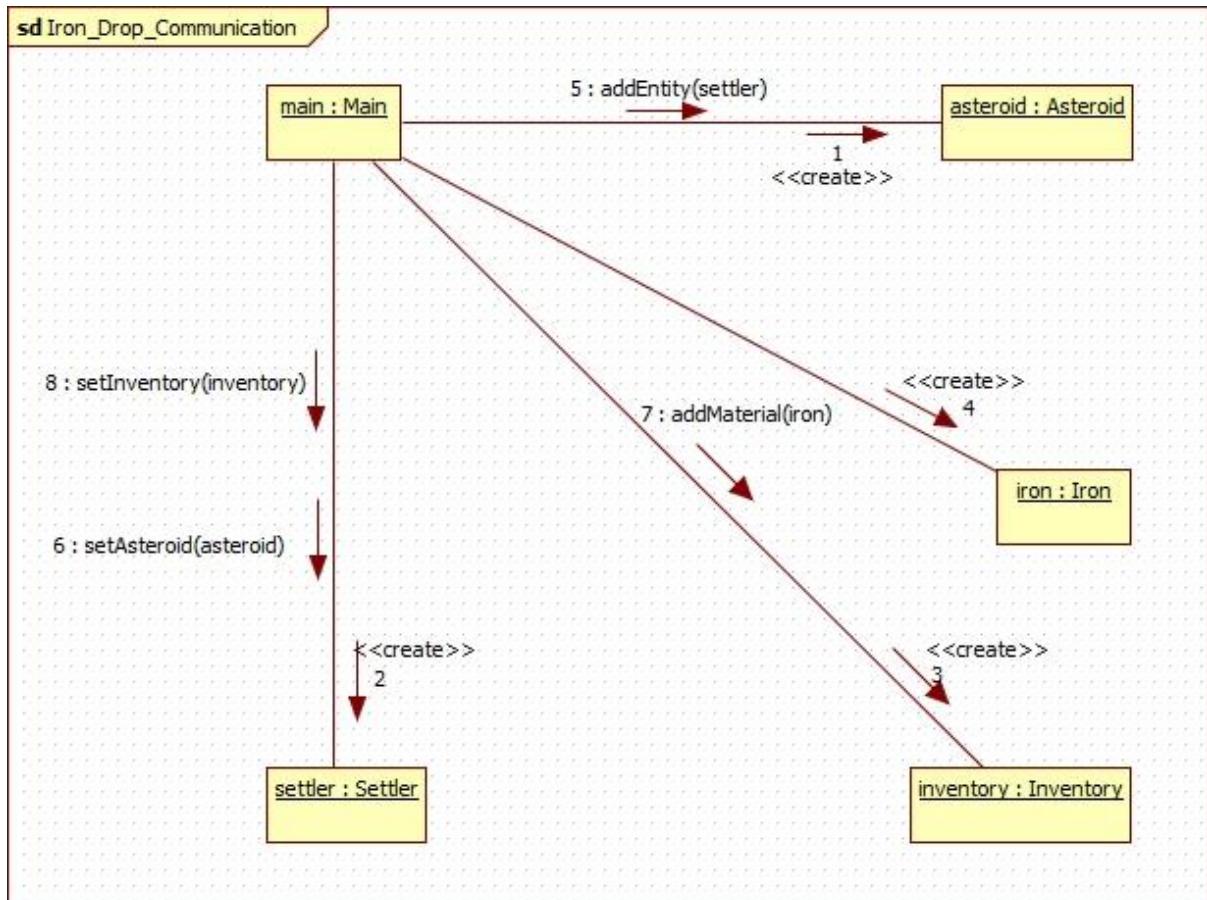
4.5.3 Vízjég bányászása



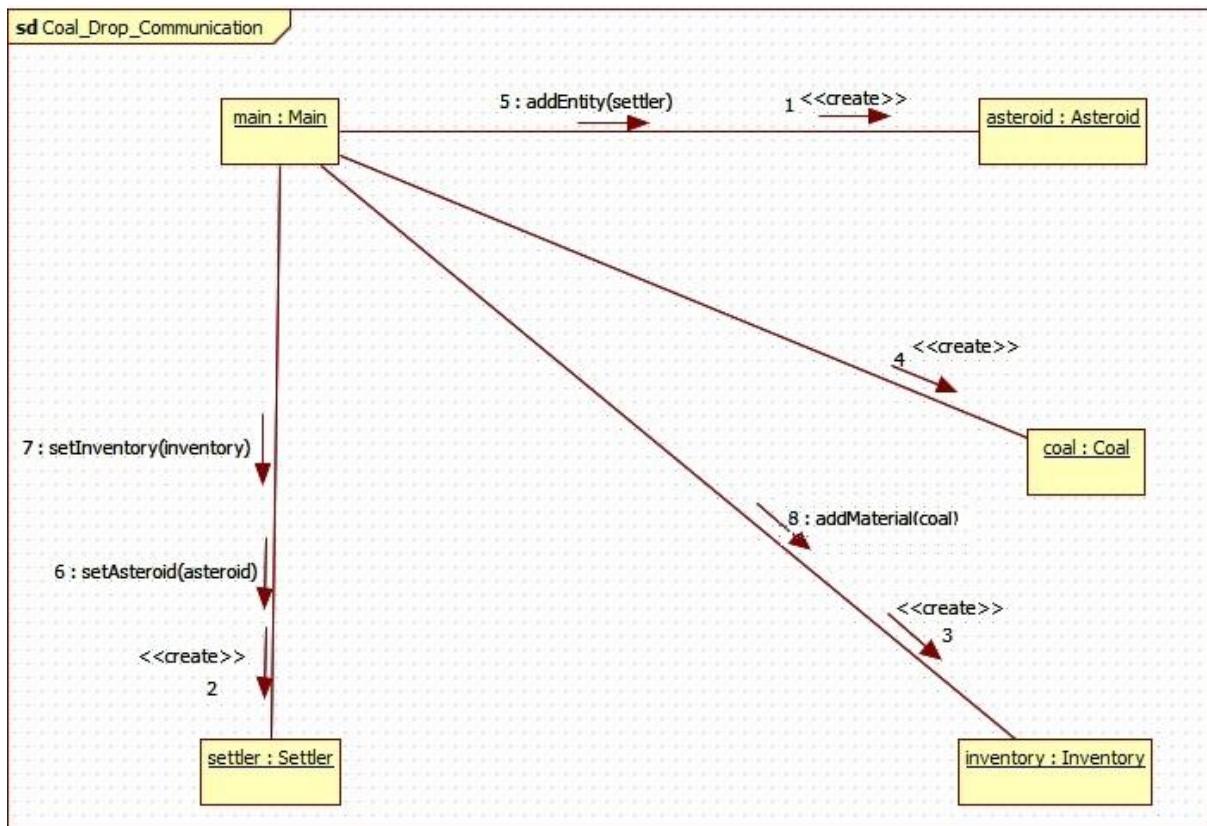
4.5.4 Urán bányászása



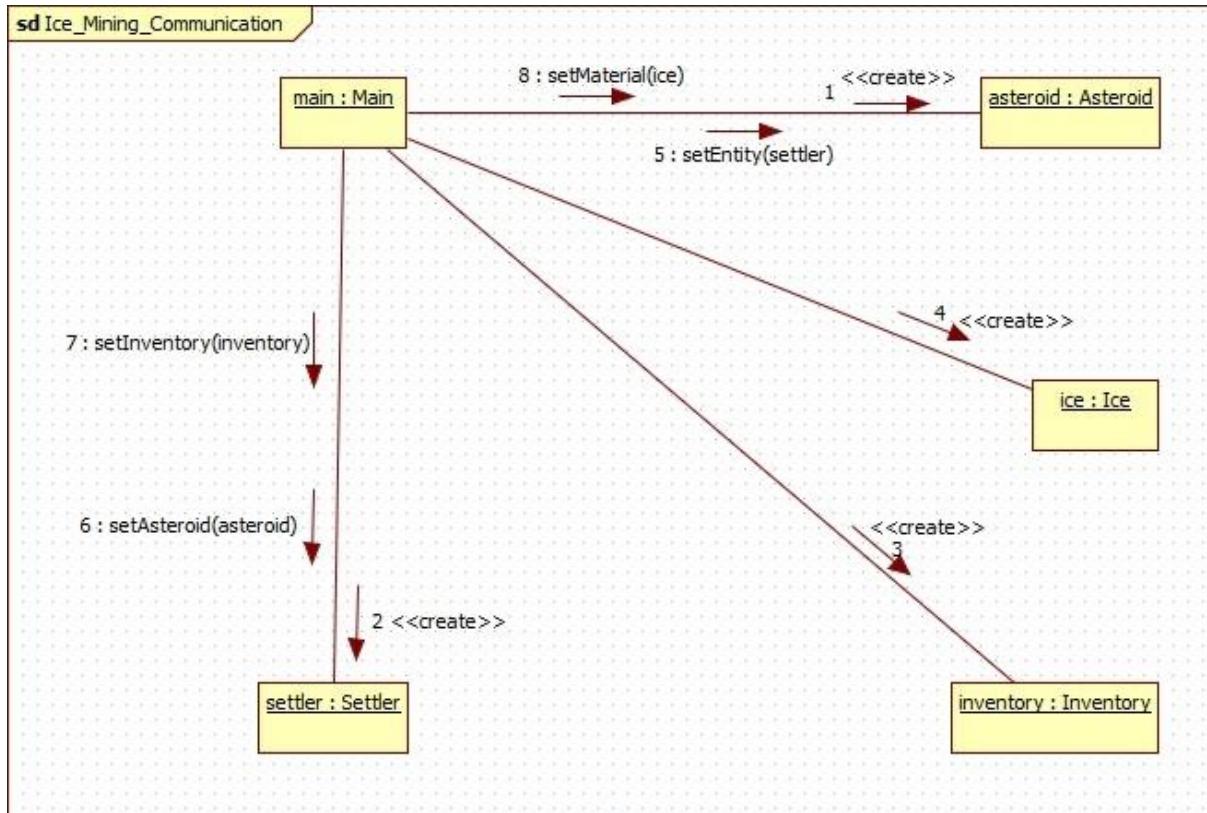
4.5.5 Vas visszahelyezése



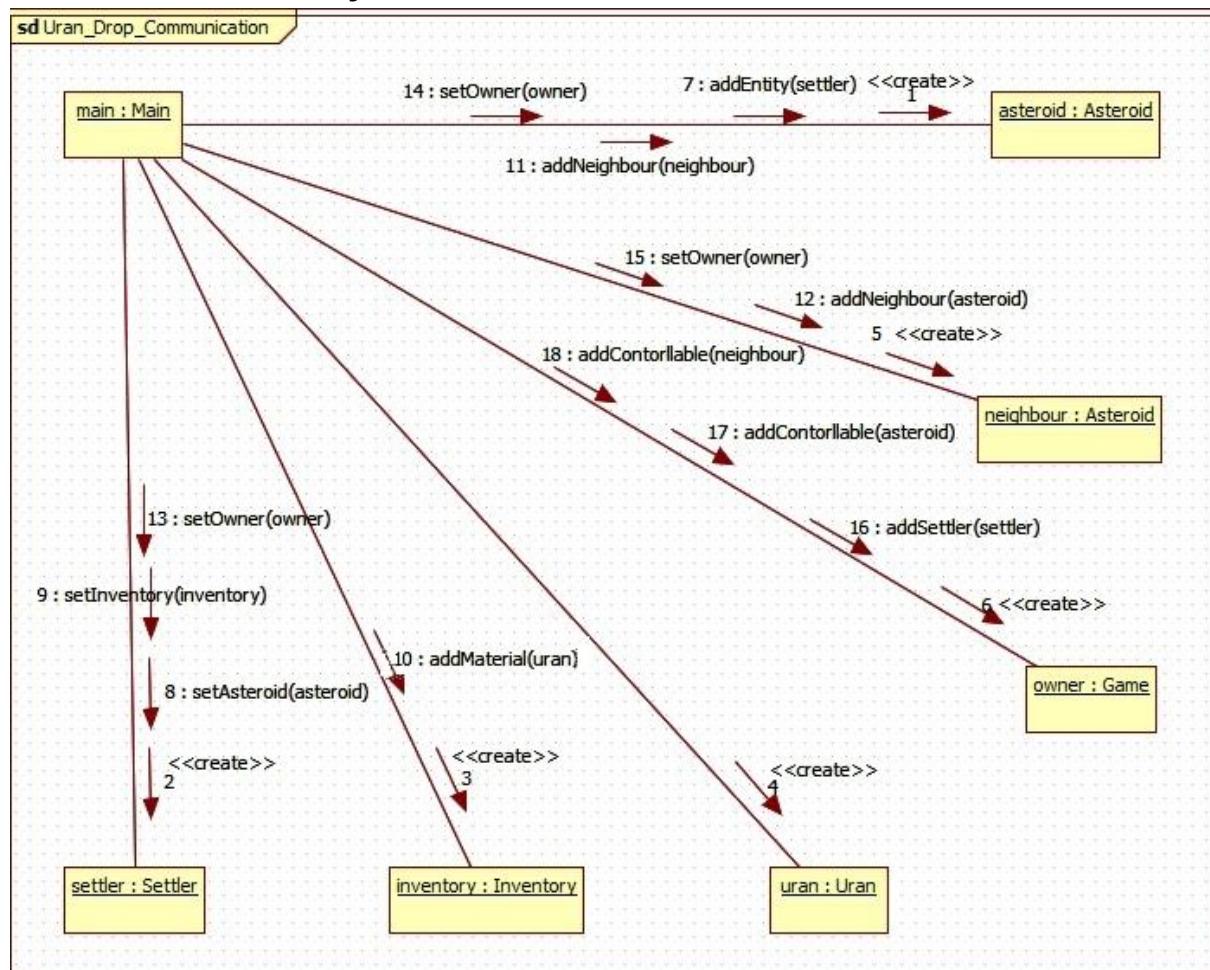
4.5.6 Szén visszahelyezése



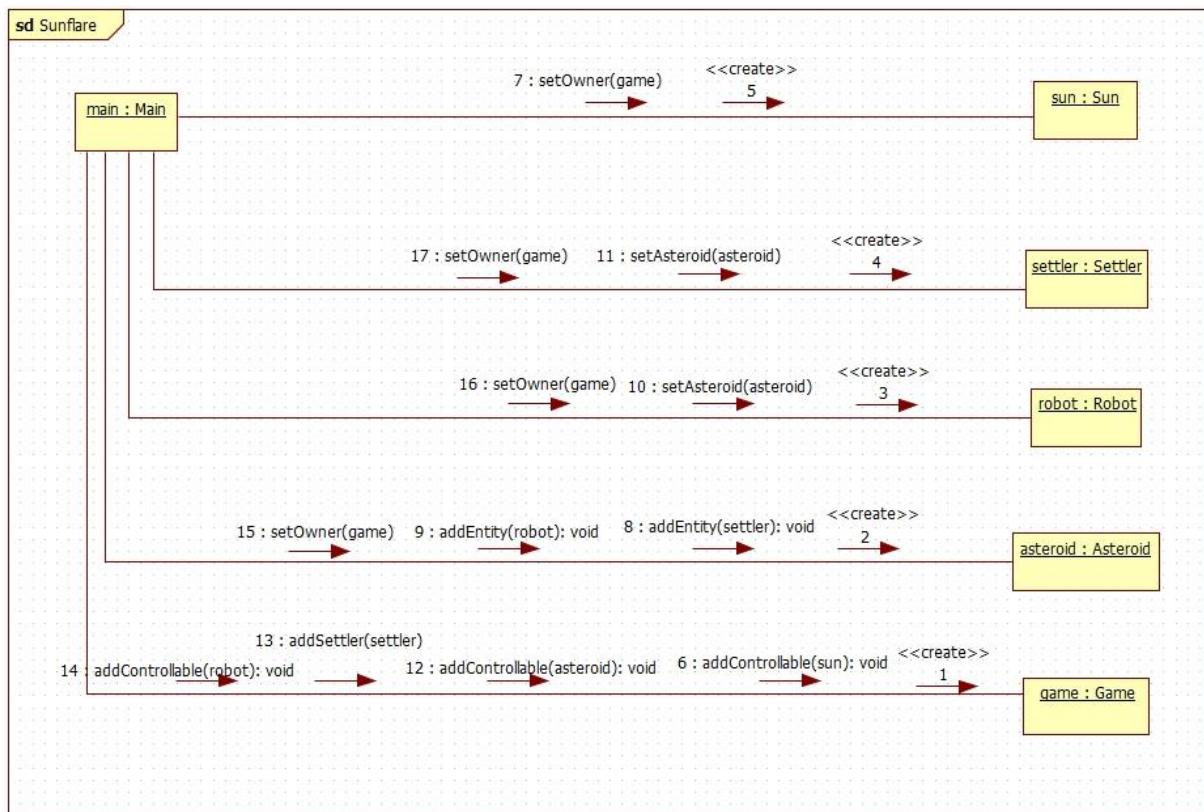
4.5.7 Vízjég visszahelyezése



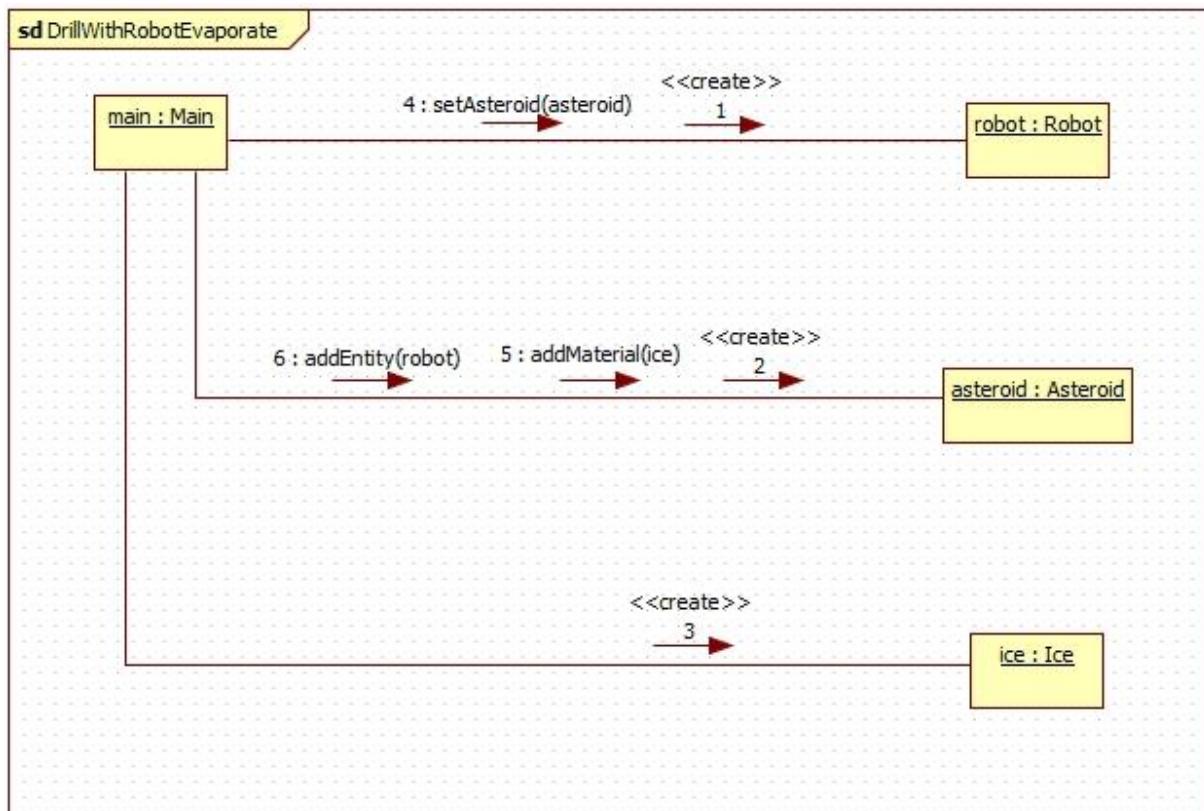
4.5.8 Urán visszahelyezése



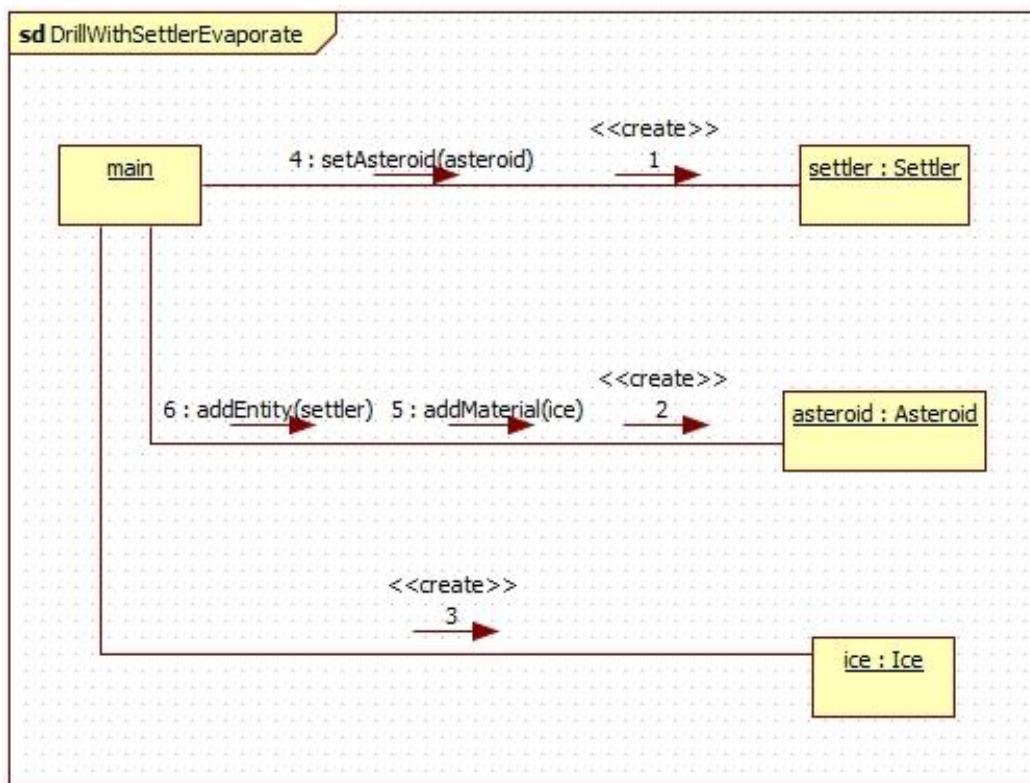
4.5.9 Napvihar történik



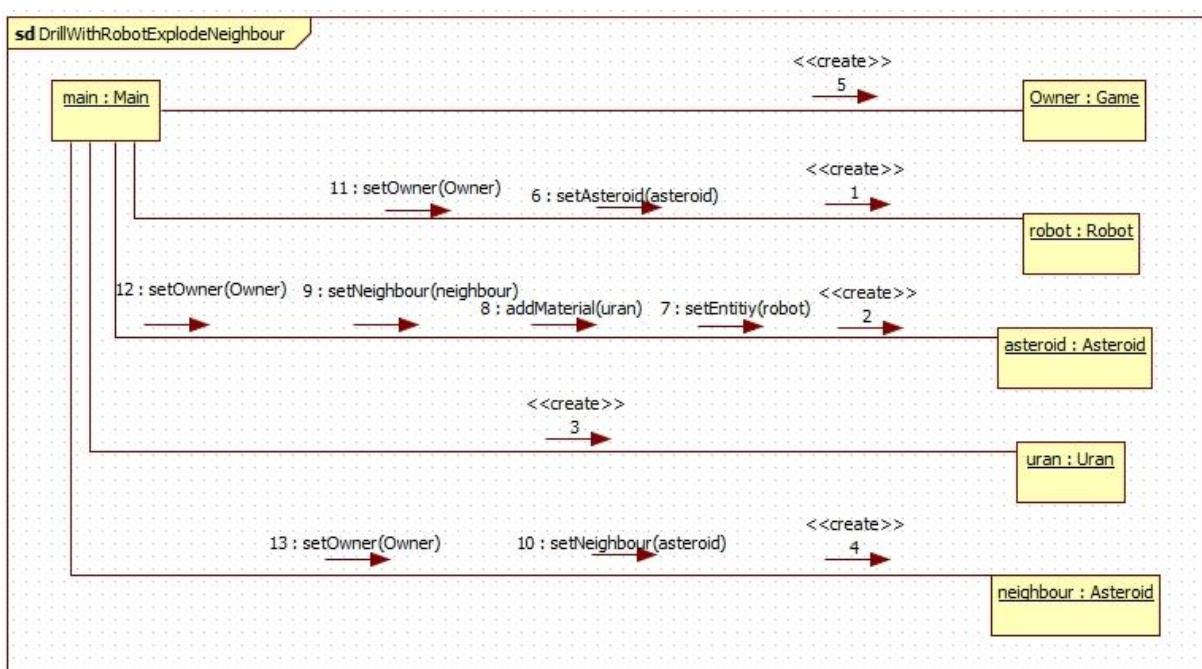
4.5.10 Fúrás robottal, elpárolog



4.5.11 Fúrás telepessel, elpárolog



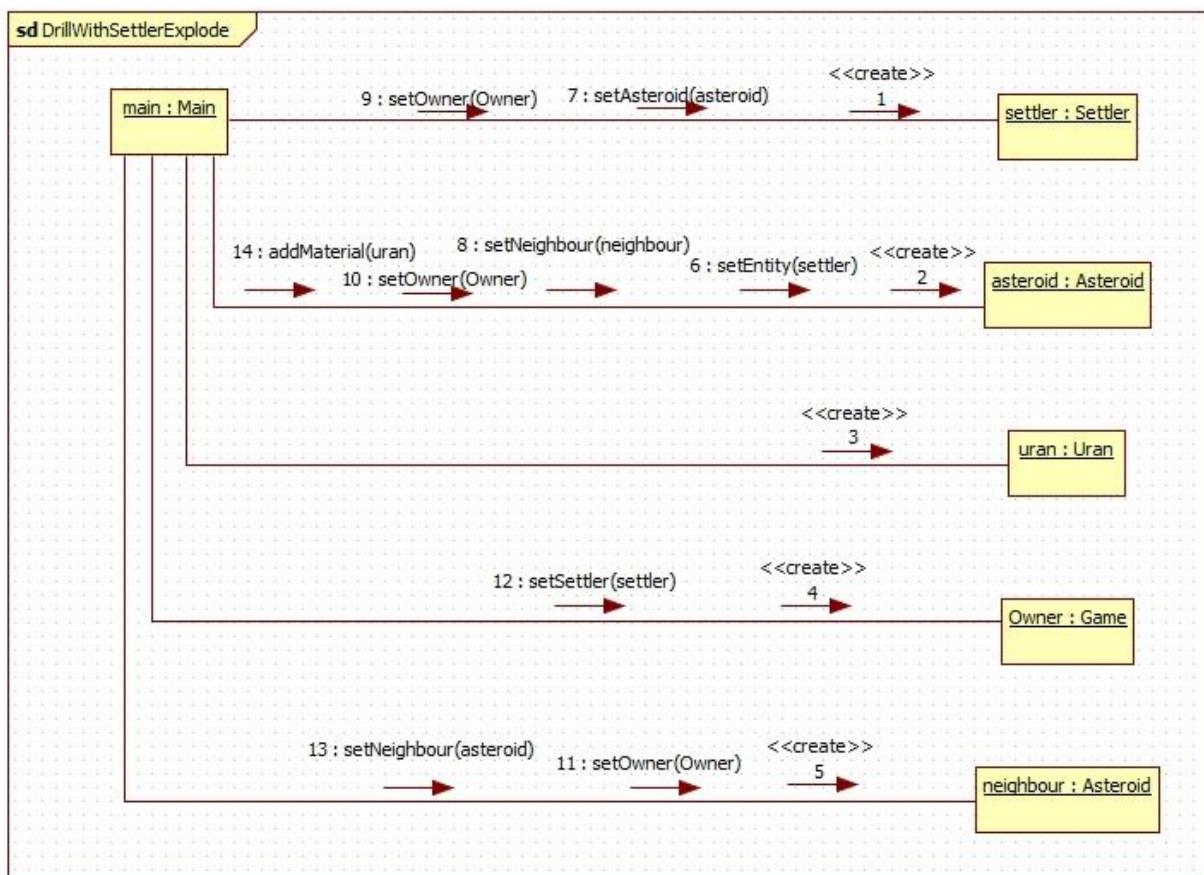
4.5.12 Fúrás robottal, robban, van szomszéd



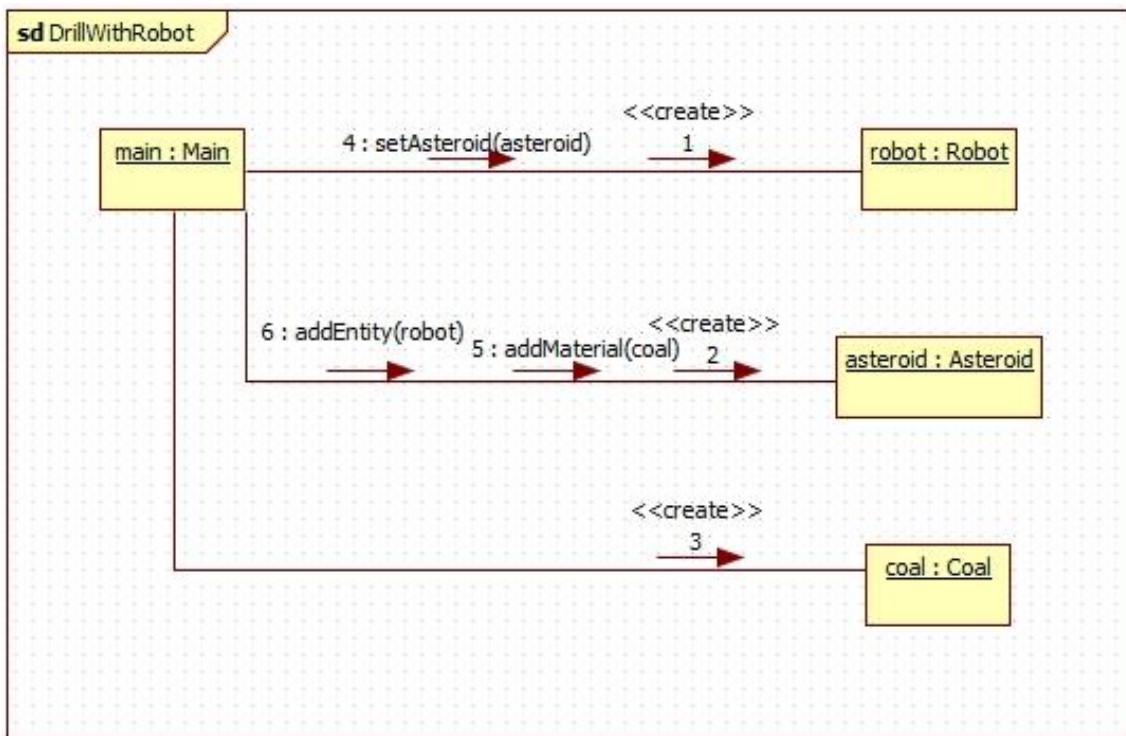
4.5.13 Fúrás robottal, robban, nincs szomszéd



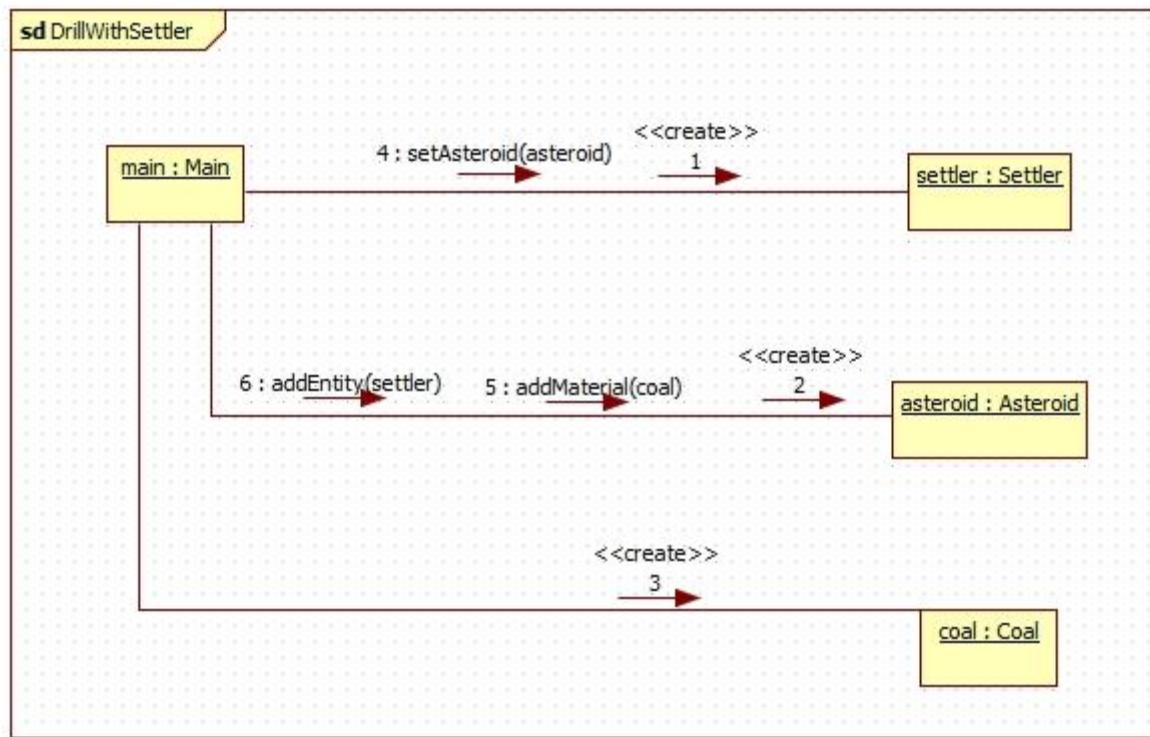
4.5.14 Fúrás telepessel, robban



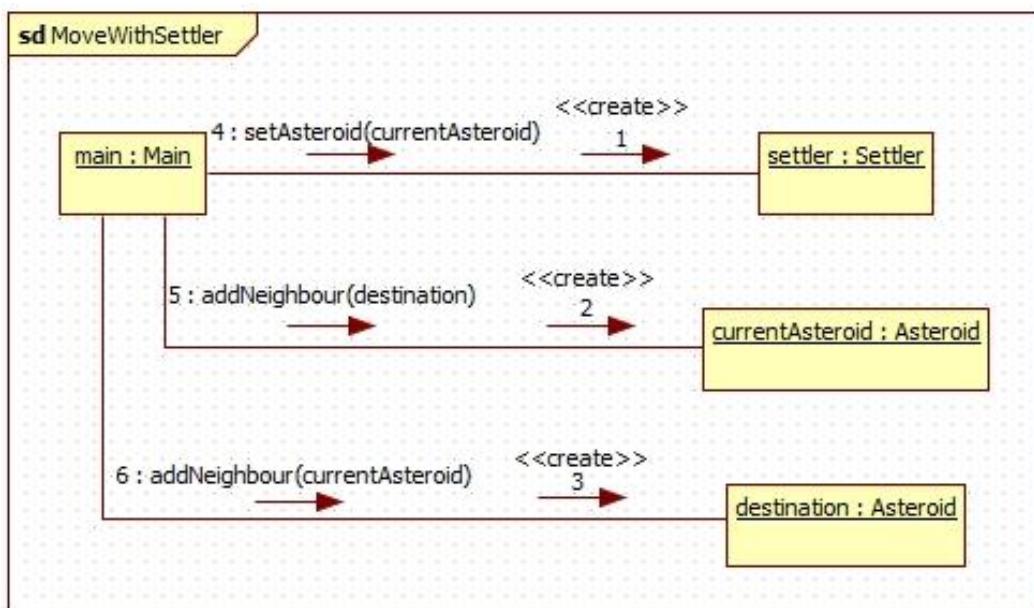
4.5.15 Fúrás robottal



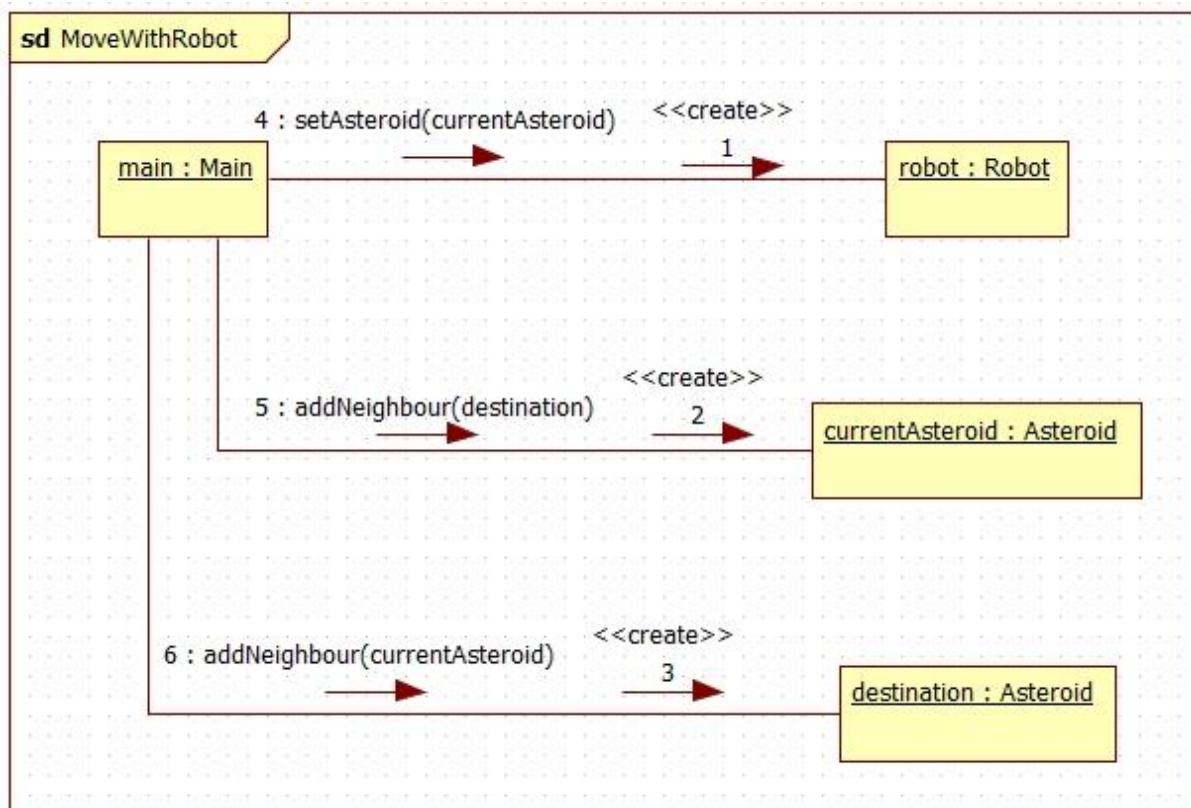
4.5.16 Fúrás telepessel



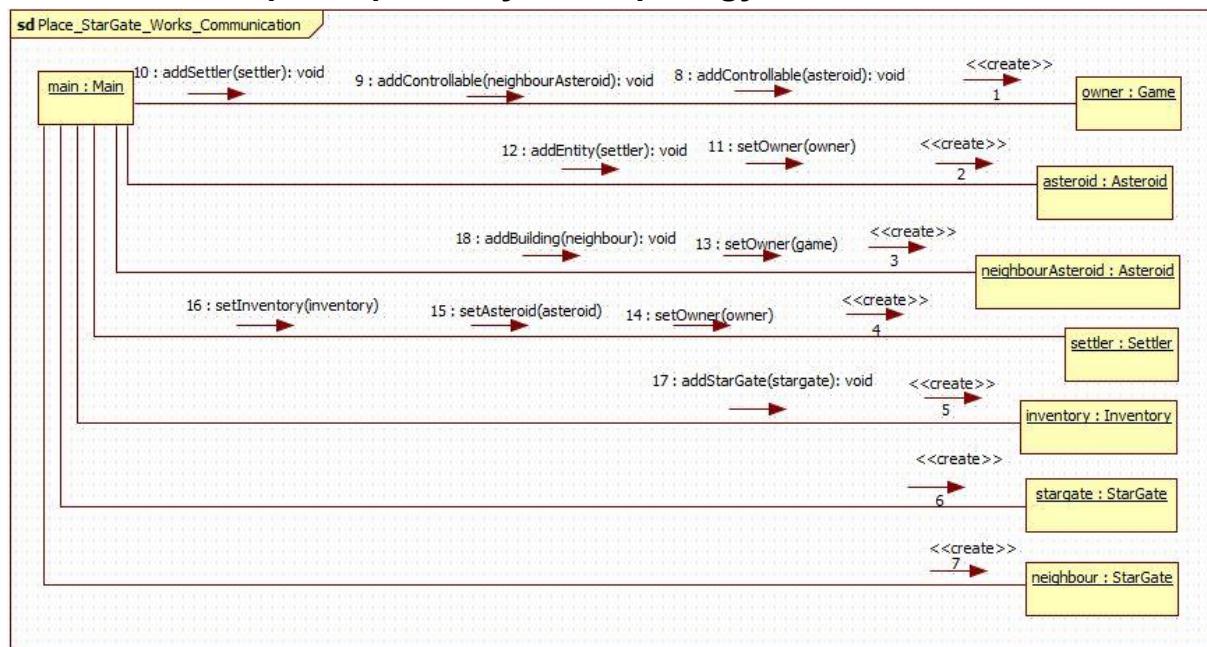
4.5.17 Mozgás telepessel



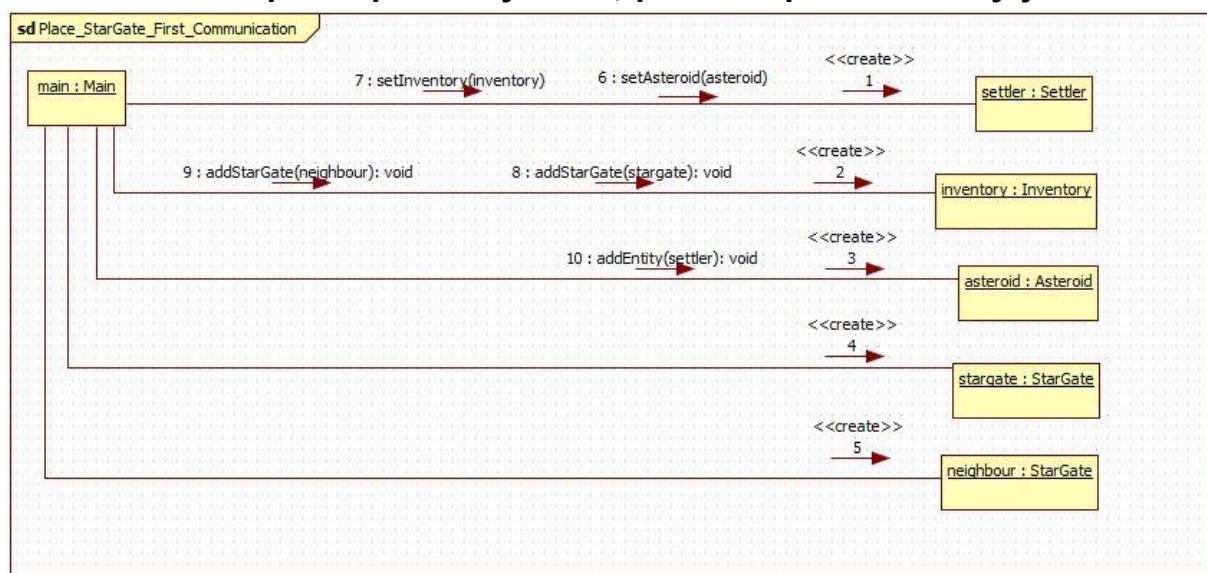
4.5.18 Mozgás robottal



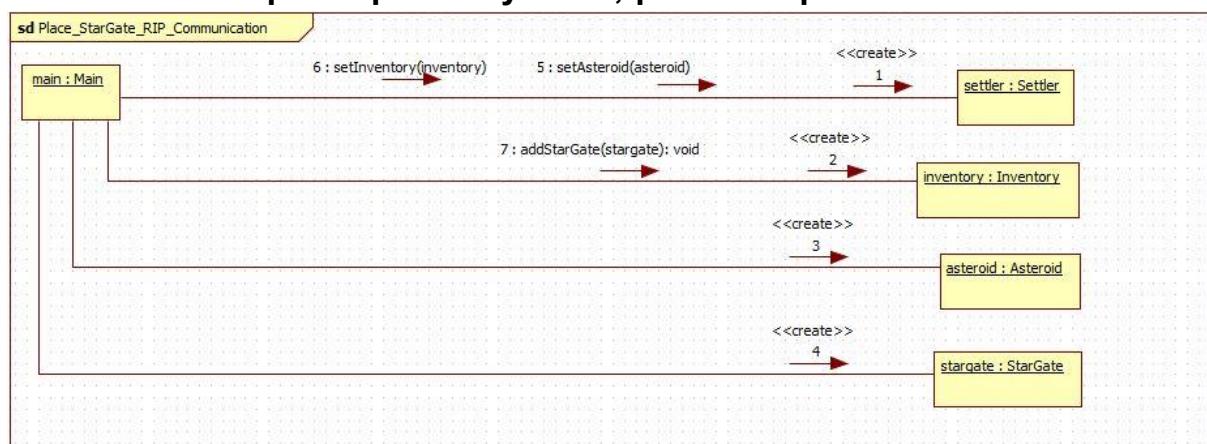
4.5.19 Teleportkapu lehelyezése, pár egy másik aszteroidán.



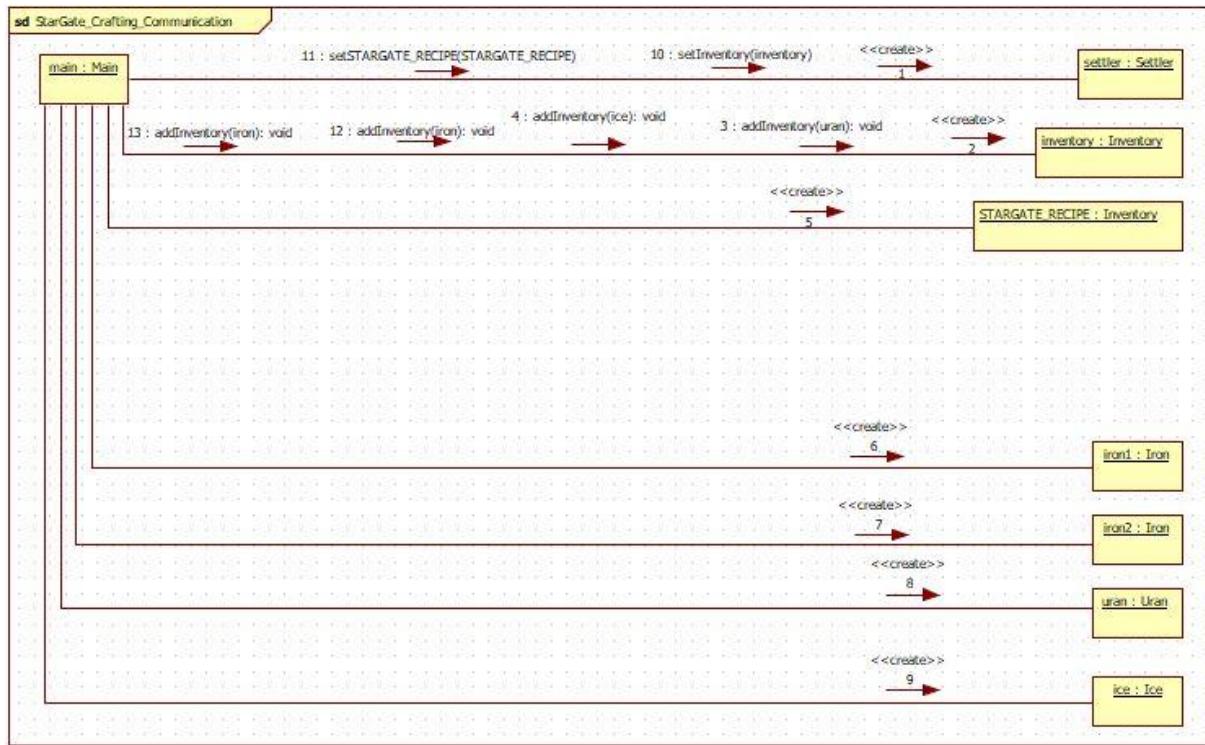
4.5.20 Teleportkapu lehelyezése, pár a telepes inventory-jában.



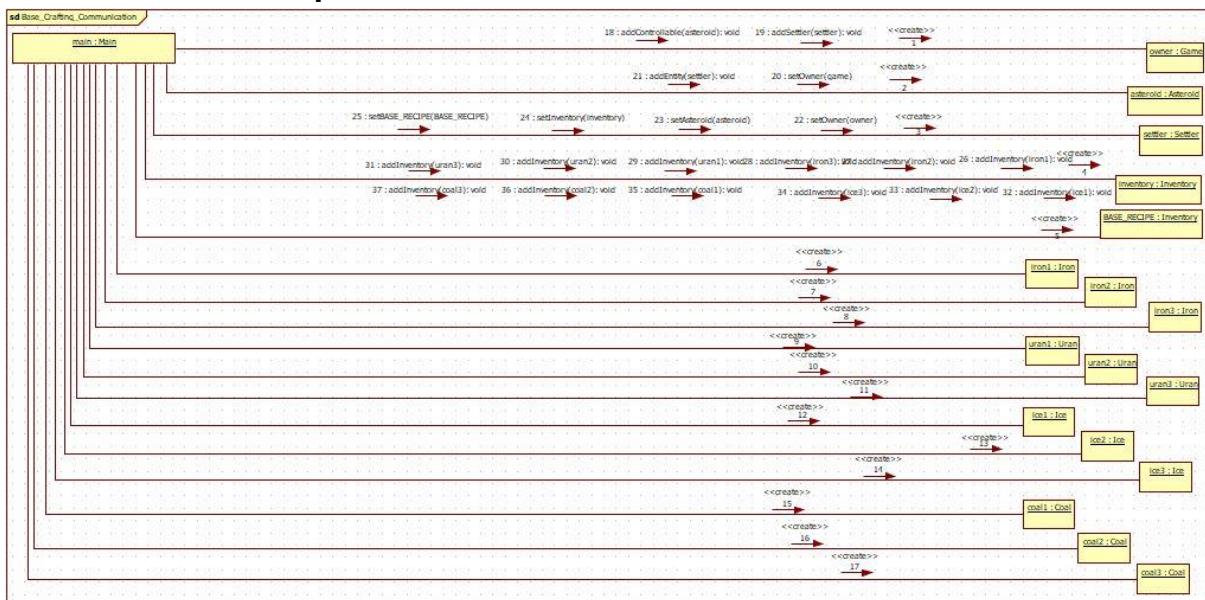
4.5.21 Teleportkapu lehelyezése, pár már elpusztult.



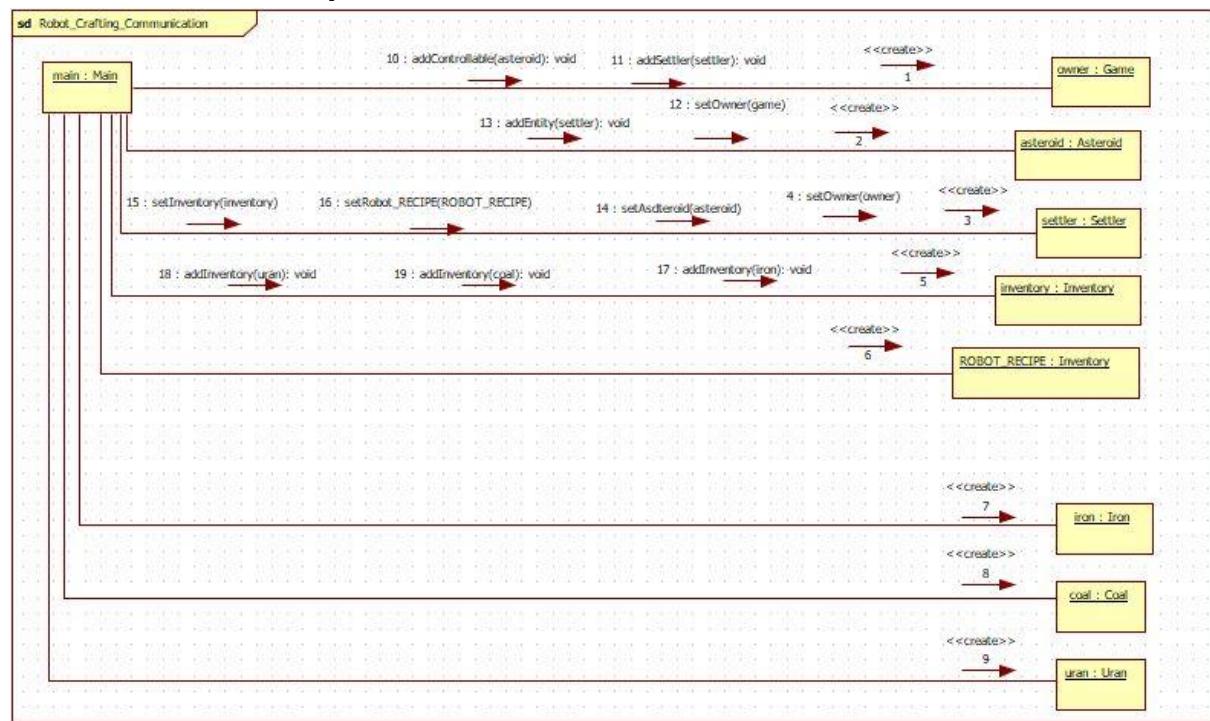
4.5.22 Teleportkapu építése



4.5.23 Bázis építése



4.5.24 Robot építése



4.6 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2021.03.9. 20:00	2 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet. Döntés: Tóth, Bárkányi, Simon, Kárpáti, elkészítí szekvenciadiagramokat és kommunikációsdiagramokat, Mohácsi a use-case leírásokat.
2021.03.12. 15:00	4 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet: A csapat megbeszélte a felmerülő problémákat a kiadót részek elkezdésével, elkészítésével. WhiteStarUML a git merge-el szét esik, ezért meg egyeztünk, hogy egyszerre csak 1 ember dolgozik vele.
2021.03.21. 20:00	4 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet: Az elkészült szekvenciadiagramokat és kommunikációs diagramokat átbeszéltük, és ezeket egyeztettük a use-case-ekkel, és egymással.
2021.03.22. 20:30	5 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Csapat munka: képek elkészítése a dokumentációhoz, kohézió ellenőrzésé, kisebb elnevezési hibák javítása.

6. Szkeleton beadása

67 – brainstormers

Konzulens:
Simon Balázs

Csapattagok

Mohácsi Márton	T8ZT88	moha.mohacsi@gmail.com
Simon Zoltán	HRSNUG	simonzolika1999@gmail.com
Kárpáti Márk András	O1BG0Z	mark.karpati1999@gmail.com
Bárkányi Csaba	BQI7QQ	csaba.barkanyi.csaba@gmail.com
Tóth Ádám	QEYOW2	toth99adam@gmail.com

2021-03-22

5. Szkeleton beadás

5.1 Fordítási és futtatási útmutató

5.1.1 Fájllista

Fájl neve	Méret	Keletkezés ideje	Tartalom
Asteroid.java	9974 byte	2021.03.17. 12:50:47	Az Asteroid osztályt tartalmazza.
Base.java	805 byte	2021.03.17. 12:50:47	A Base osztályt tartalmazza, amely megépítése a játék megnyerését eredményezi.
Building.java	1107	2021.03.17. 12:50:47	A Building osztályt tartalmazza, ami a Base és StarGate osztályok őse.
Coal.java	277 byte	2021.03.17. 12:50:47	A Coal osztályt tartalmazza, ami a Material leszármazottja.
Controllable.java	263 byte	2021.03.17. 12:50:47	A Controllable interfészst tartalmazza.
Entity.java	2058 byte	2021.03.17. 12:50:47	Az Entity ősosztályt tartalmazza.
Game.java	4845 byte	2021.03.17. 12:50:47	A Game osztályt tartalmazza, ami a játék fő osztálya.
Ice.java	633 byte	2021.03.17. 12:50:47	Az Ice osztályt tartalmazza, ami a Material leszármazottja.
Inventory.java	5105 byte	2021.03.17. 12:50:47	Az Inventory osztályt tartalmazza, amely nyersanyagokat tárol.
Iron.java	324 byte	2021.03.17. 12:50:47	Az Iron osztályt tartalmazza, ami a Material leszármazottja.
Main.java	50278 byte	2021.03.17. 12:50:47	A Main osztályt tartalmazza, amely a skeleton tesztelésért felelős.
Material.java	2589 byte	2021.03.17. 12:50:47	A Material osztályt tartalmazza, ami az egyes nyersanyag típusok őse.
Planet.java	890 byte	2021.03.17. 12:50:47	A Planet osztályt tartalmazza, ami az Asteroid és Sun őse.
Robot.java	1863 byte	2021.03.17. 12:50:47	A Robot osztályt tartalmazza, ami az Entity leszármazottja.
Settler.java	5807 byte	2021.03.17. 12:50:47	A Settler osztályt tartalmazza, ami az Entity leszármazottja.
StarGate.java	2268 byte	2021.03.17. 12:50:47	Az StarGate osztályt tartalmazza.
SkeletonBuildScript.bat	90 byte	2021.03.22. 10:52:02	mkdir ./bin javac -d ./bin src/main/*.java jar cfve Skeleton.jar main.Main -C ./bin/ .

SkeletonRunScript.bat	22 byte	2021.03.22. 10:35:39	java -jar Skeleton.jar
-----------------------	---------	----------------------	------------------------

5.1.2 Fordítás

A fordításhoz szükséges Java Development Kit (JDK).

Csomagoljuk ki a Brainstormers_skeleton.zip-et.

Futtassuk a SkeletonBuildScript.bat fájlt.

Ez létrehoz egy bin könyvtárat a root könyvtárban és a src mappa tartalmát lefordítva a bin/-be írja.

(Ehhez a binben létrejön egy main alkönyvtár is.)

Végül a lefordított classokat egy Skeleton.jar-ba csomagolja. A Main osztályt határozza meg, mint főosztály.

5.1.3 Futtatás

A futtastáshoz a SkeletonRunScript.bat fájlt futtassuk!

Ez a korábban létrehozott Skeleton.jar csomagot futtatja.

A megnyílt parancssorban használhatjuk az elindult skeleton programot.

5.1.4 Program használata

A program konzolban használható az előző pontokban leírt futtatás után. A megjelenő menüben a kijelzett menüpontok között a kiválasztott sorszámának a beírásával és enter ütésével választhatunk. A visszalépés hasonló módon történik, amely minden esetben a 0-ás szám választásával történik.

A program futása során a menü által megjelenített opciónkon kívüli választás esetén a program futása megszakadhat. A hiba a program újraindításával szüntethető meg.

A program csak az elvárt és kijelzett számokat fogadja el bemenetként.

5.2 Értékelés

Tag neve	Tag neptun	Munka százalékban
Tóth	QEYOW2	20%
Bárkányi	BQI7QQ	20%
Simon	HRSNUG	20%
Mohácsi	T8ZT88	20%
Kárpáti	O1BG0Z	20%

5.3 Napló

Kezdet	Időtartam	Résznevők	Leírás
2021.03.17. 11:00	1 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet. Döntés: Eldöntötök, hogy ki mit csinál. Tóth: robot, settler, entity Mohácsi: Main, buildings, teleport, vase Simon: Game,sun,planet, Bárkányi: Inventory, materials, ice, coal, uranium, iron Kárpáti: Aszteroida controlable
2021.03.19. 15:00	2 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet: Elkezdett feladatokban felvetett hibákat megbeszéltük.
2021.03.20. 15:00	3 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet: Felvetett hibák javítás megbeszélése.
2021.03.21. 16:00	6 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Csapat munka: use case-k tesztelés, felmerülő program béli hibák javítás, bázis elkészítésének közös megcsinálása.
2021.03.22. 9:00	4 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Csapat megbeszélte és adminisztrálta a dokumentum tartalmát.

7. Prototípus koncepciója

67 – brainstormers

Konzulens:
Simon Balázs

Csapattagok

Mohácsi Márton

Simon Zoltán

Kárpáti Márk András

Bárkányi Csaba

Tóth Ádám

T8ZT88

HRSNUG

O1BG0Z

BQI7QQ

QEYOW2

moha.mohacsi@gmail.com

simonzolika1999@gmail.com

mark.karpati1999@gmail.com

csaba.barkanyi.csaba@gmail.com

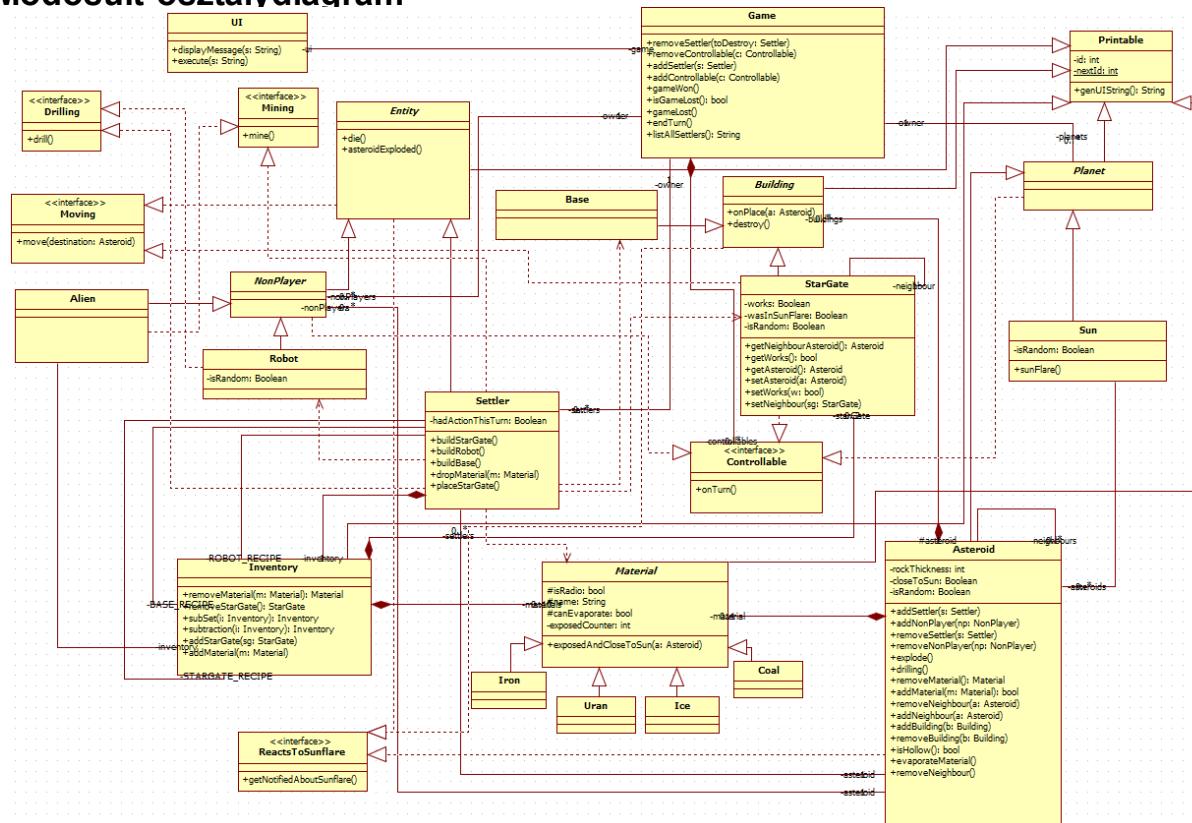
toth99adam@gmail.com

2021-03-29

7. Prototípus koncepciója

7.1 Változás hatása a modellre

7.1.1 Módosult osztálydiagram



7.1.2 Új vagy megváltozó metódusok

Az alábbiakban felsoroljuk az összes olyan osztályt, amelyekben megváltozott vagy hozzáadódott függvény. minden ilyen osztályhoz felsoroljuk külön a megváltozott függvényeket, illetve az új függvényeket.

Asteroid:

Új függvények:

addSettler(Settler: s): Hozzáadja az aszteroida settlers listájához az s Settlert. (az előző tervhez képest az aszteroida nem controllableket tárol, hanem settlereket s NonPlayereket)

addNonPlayer(NonPlayer: np): Hozzáadja az aszteroida nonPlayers listájához az np NonPlayert. (az előző tervhez képest az aszteroida nem controllableket tárol, hanem settlereket s NonPlayereket)

removeSettler(Settler: s): Kiveszi az aszteroida settlers listájából az s Settlert. (az előző tervhez képest az aszteroida nem controllableket tárol, hanem settlereket s NonPlayereket)

removeNonPlayer(NonPlayer: np): Kiveszi az aszteroida nonPlayers listájából az np NonPlayert. (az előző tervhez képest az aszteroida nem controllableket tárol, hanem settlereket s NonPlayereket)

s: string genUIString(): Generál egy stringet mely kiírja az aszteroida fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. A kiírt tulajdonságok a következők: id, kéreg vastagság, aszteroidában található nyersanyag, aszteroidán található épületek, aszteroidán található entitások, az aszteroida szomszédjainak id-je.

Base:

Új függvények:

s: string genUIString(): Generál egy stringet mely kiírja a bázis fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. A kiírt tulajdonságok a következők: id, illetve az aszteroida id-je, amin a bázis van.

getNotifiedAboutSunflare(): Ez a függvény akkor hívódik meg ha éppen napvihar történik, ez egy absztrakt függvény.

Building:

Új függvények:

s: string genUIString(): Generál egy stringet mely kiírja az épület fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. Ez egy absztrakt függvény a konkrét stringekért az egyes épületek felelnek.

getNotifiedAboutSunflare(): Ez a függvény akkor hívódik meg ha éppen napvihar történik, a bázissal napviharkor nem történik semmi.

Coal:

Új függvények:

s: string genUIString(): Generál egy stringet mely kiírja a szén fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. A kiírt tulajdonságok: a szén neve, id-je, illetve öregedési változója.

Entity:

Új függvények:

s: string genUIString(): Generál egy stringet mely kiírja az entitás fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. Ez egy absztrakt függvény a konkrét stringekért az egyes entitások felelnek.

Game:**Új függvények:**

removeControllable(c: Controllable): Kiveszi a game controllables listából a c Controllablet.

removeSettler(s: Settler): Kiveszi a game settlers listából az s Settlert.

addSettler(s: Settler): Hozzáadja a game settlers listájához az s settlert.

b: Boolean isGameLost(): ez a függvény ellenőrzi, hogy veszített-e a játékos.

gameLost(): Ez a függvény elvégzi a játék elvesztésével kapcsolatos feladatokat. Pl. a játékosnak üzenet kiírása, mindenféle listák elengedése.

startTurn(): Ez a függvény akkor hívódik amikor a játékos befejezi a körét, ilyenkor minden controllable onTurn() függvénye meghívódik, hogy minden controllable elvégezhesse a dolgát.

listAllSettlers(): String: Ez a függvény visszaad egy stringet az összes settler releváns információjával, ezt az UI használja fel a játékossal való kommunikációhoz.

Ice:**Új függvények:**

s: string genUIString(): Generál egy stringet mely kiírja a jég fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. A kiírt tulajdonságok: a jég neve, id-je, illetve öregedési változója.

Inventory:**Új függvények:**

addMaterial(m: Material): Ez a függvény berakja az m materialt az inventoryba. (Az eddigi tervezetben a materialok inventorybe pakolását az adott materialok végezték.)

i: Inventory subtraction(i: Inventory): Ez a függvény az adott inventoryból kivon egy másik inventoryt, illetve ha nem tudja teljesen kivonni a másik inventoryt, akkor kivon annyi nyersanyagot amennyit csak tud. Kivonás során a nyersanyagok eltűnnek az adott inventoryból. Visszatérni egy inventoryval fog, mely tartalmazza azokat a nyersanyagokat, amelyeket nem tudott kivonni az első inventoryból. Ezt a függvényt bázis építésekor használjuk fel rendesen, mikoris egy settlernek nem lehet elég nyersanyaga egy bázishoz, így több settler inventoryjából kell azokat eltűntetnünk.

s: string genUIString(): Generál egy stringet mely kiírja az Inventory fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. A kiírt tulajdonságok a következők: az inventoryban található nyersanyagok, illetve az egyes nyersanyagok öregedési szintje.

Megváltozott függvények:

I: Inventory subSet(i: Inventory): Ez a függvény megadja, hogy az adott inventoryból melyen nyersanyagok kellenének még ahhoz, hogy tartalmazza az i inventoryt. Ezt a függvényt bázis építésekor használjuk fel rendesen, mikoris egy settlernek nem lehet elég nyersanyaga egy bázishoz, így több settler inventoryjában kell azokat keresnünk.

Iron:**Új függvények:**

s: string genUIString(): Generál egy stringet mely kiírja a vas fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. A kiírt tulajdonságok: a vas neve, id-je, illetve öregedési változója.

Material:

Új függvények:

s: string genUIString(): Generál egy stringet mely kiírja a nyersanyag fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. Ez egy absztrakt függvény a konkrét stringekért az egyes nyersanyagok felelnek.

Planet:

Új függvények:

s: string genUIString(): Generál egy stringet mely kiírja az égitest fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. Ez egy absztrakt függvény a konkrét stringekért az egyes égitestek felelnek.

Robot:

Új függvények:

s: string genUIString(): Generál egy stringet mely kiírja a robot fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. A kiírt tulajdonságok: a robot id-je, illetve az aszteroidája.

Settler:

Új függvények:

getNotifiedAboutSunflare(): Ez a függvény akkor hívódik meg ha éppen napvihar történik, ilyenkor a telepes meghal.

s: string genUIString(): Generál egy stringet mely kiírja a telepes fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. A kiírt tulajdonságok: id, inventoryban található StarGate-k id-jei, az inventoryban található nyersanyagok nevei, id-jei s öregségei, illetve az aszteroida id-je s annak szomszédainak id-je.

StarGate:

Új függvények:

move(destination: Asteroid): Ez a függvény mozgatja a teleportkaput az aszteroidájáról aszteroidáról a destination aszteroidára s ennek következményeit kezeli, mint például a megfelelő aszteroidának elveszi bizonyos szomszédait, illetve a másikhoz hozzáad bizonyos szomszédokat.

onTurn(): Ez a függvény a teleportkapu körben elvégzendő feladatait végzi el. Ha a teleportkaput napvihar éri, akkor minden körben egy random szomszédos aszteroidára mozog át.

getNotifiedAboutSunflare(): Ez a függvény akkor hívódik meg ha éppen napvihar történik, beállítja a wasInSunFlare flaget igazra, innentől a teleportkapu elkezd mozogni.

s: string genUIString(): Generál egy stringet mely kiírja a StarGate fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. A kiírt tulajdonságok a következők: az id, a szomszédos StarGate id-je és szomszédos StarGate aszteroidájának id-je.

Sun:

Megváltozott függvények:

sunFlare(): Ez a függvény minden a naphoz tartozó s napközelben lévő aszteroidának szól, hogy napkitörés történik.

Uran:

Új függvények:

s: string genUIString(): Generál egy stringet mely kiírja az urán fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. A kiírt tulajdonságok: az urán neve, id-je illetve öregedési változója.

Az alábbiakban az előző tervhez képest új interfaceket s osztályokat soroljuk fel:
Interfacek:

Drilling: Ezt az interfacet olyan dolgok valósítják meg melyek képesek aszteroida felületét fúrni, benne egyetlen absztrakt drill() függvény található.

Mining: Ezt az interfacet olyan dolgok valósítják meg melyek képesek aszteroidából bányászni, benne egyetlen absztrakt mine() függvény található.

Moving: Ezt az interfacet olyan dolgok valósítják meg melyek egy aszteroidáról egy másik aszteroidára képesek átmozogni, benne egyetlen absztrakt move() függvény található.

ReactsToSunflare: Ezt az interfacet olyan dolgok valósítják meg melyek reagálnak napviharra, benne egyetlen absztrakt getNotifiedAboutSunflare() függvény található.

Osztályok:

Printable: Ez az osztály egy absztrakt űosztály mely azért felel, hogy minden doognak a játékban melynek külön id-re van szüksége megkapja azt s az egyedi legyen, illetve, hogy azok az osztályok melyek ebből származnak le tudjanak generálni egy Stringet amit az UI ki tud majd írni.

Függvények:

s: string genUIString(): Generál egy stringet mely kiírja az adott printable objektum fontos tulajdonságait, ezt az UI osztály fogja felhasználni, hogy kommunikáljon a játékossal. Ez egy absztrakt függvény a konkrét stringekért az egyes printable objektumok felelnek.

UI: Ez az osztály a játékos s a játék között kommunikál, ez az osztály írja ki a játék üzeneteit, illetve ez az osztály dolgozza fel a játékos parancsait.

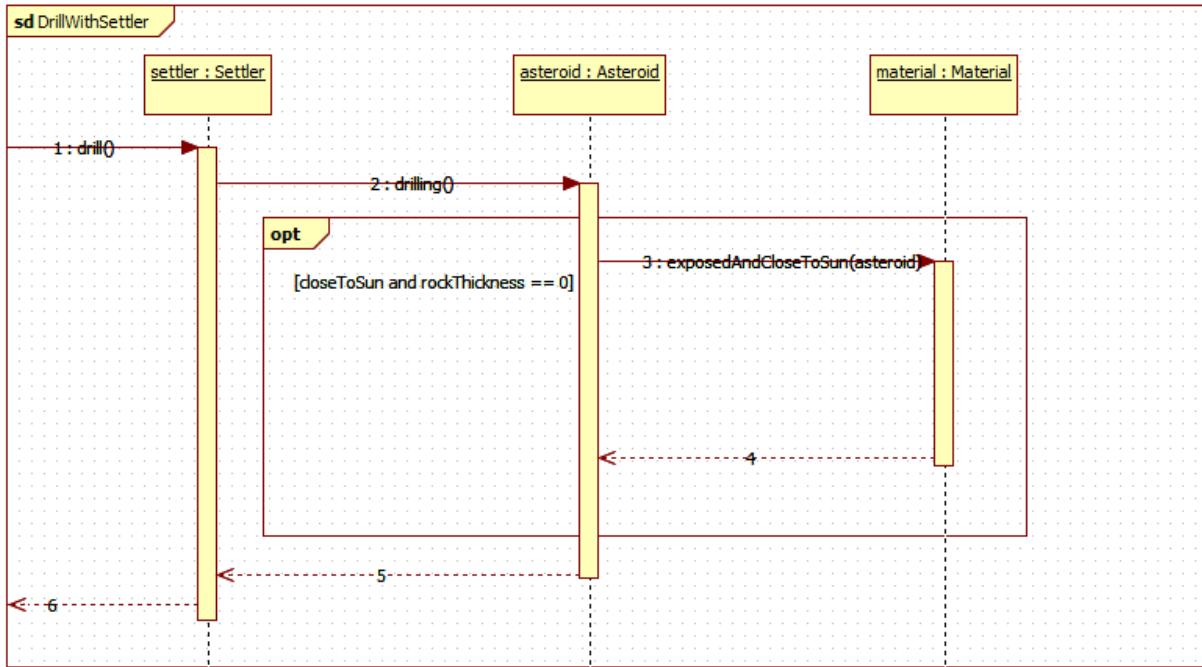
Függvények:

displayMessage(s: String): Ez a függvény kiírja az s Stringben található üzenetet a képernyőre.

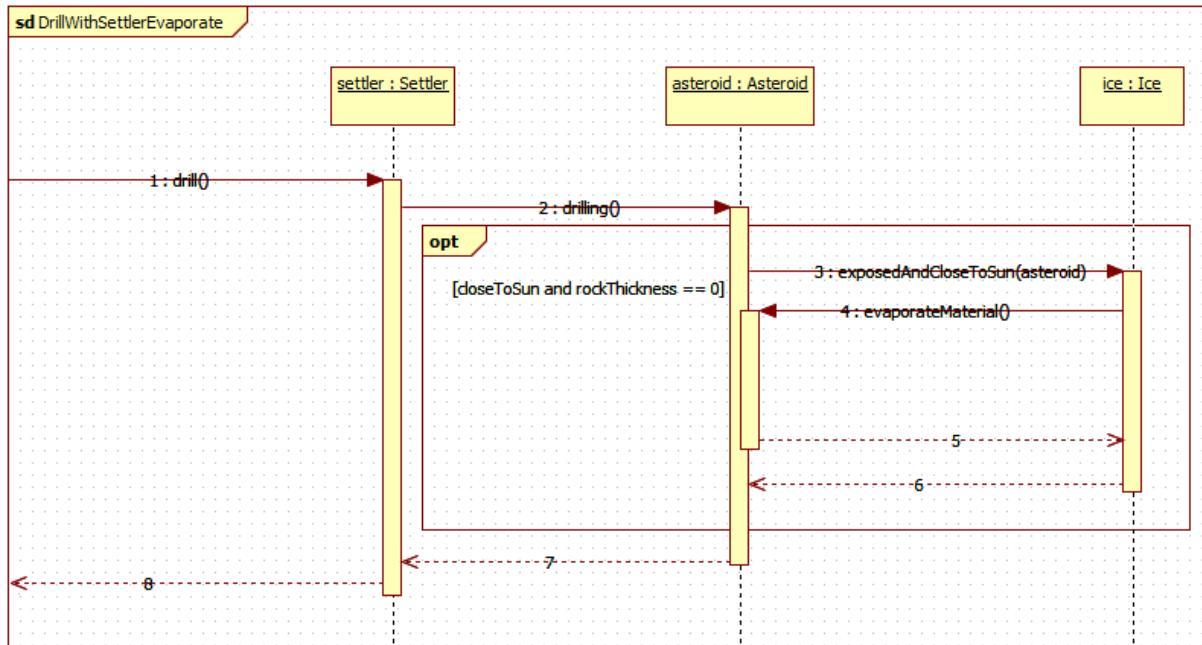
execute(s: String): Ez a függvény végrehajtja az s Stringben specifikált játékos által kiadott parancsot.

7.1.3 Szekvencia-diagramok

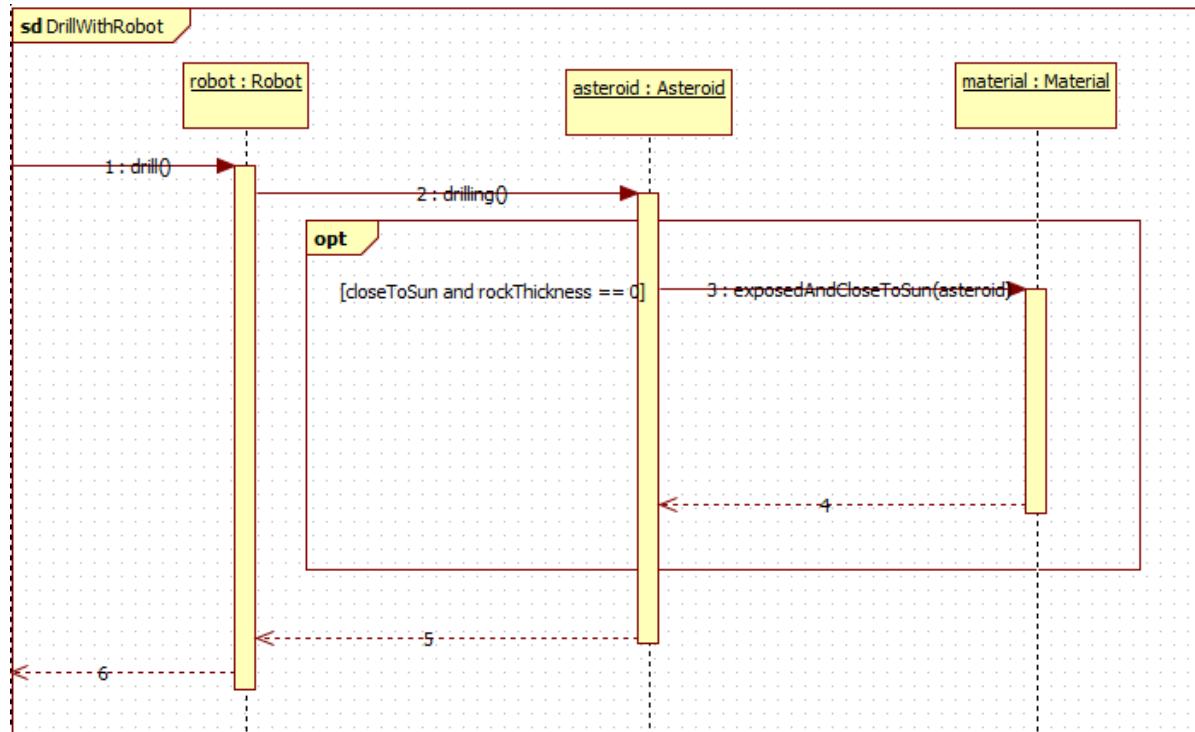
7.1.3.1 Drill With Settler



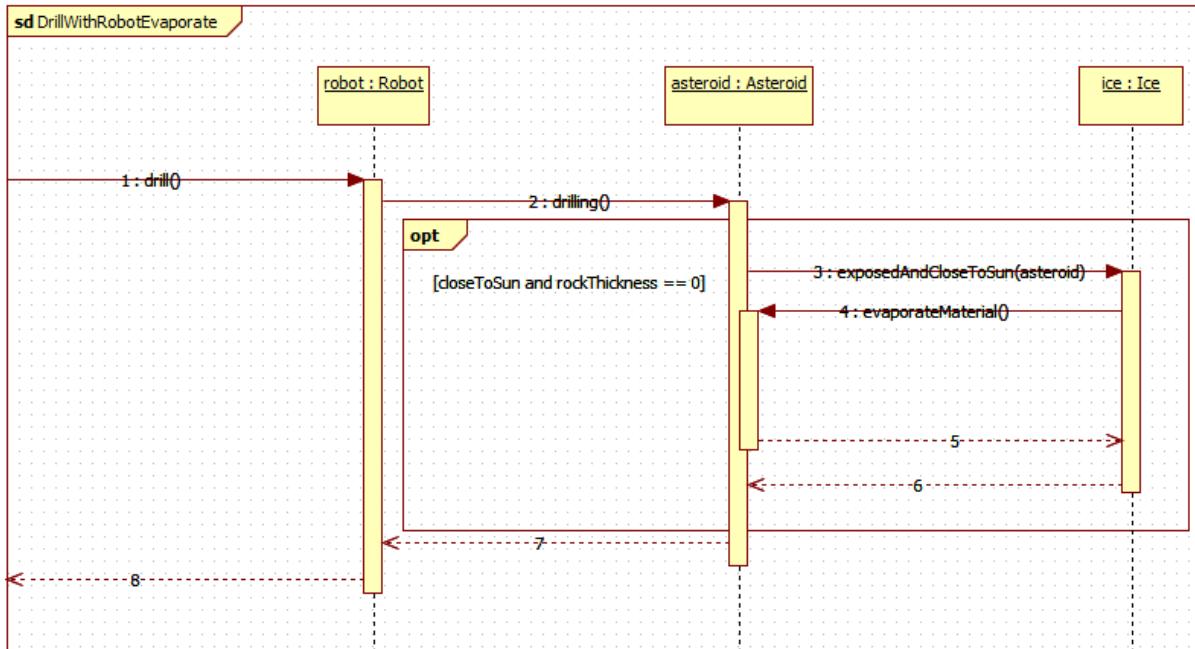
7.1.3.2 Drill With Settler Evaporate



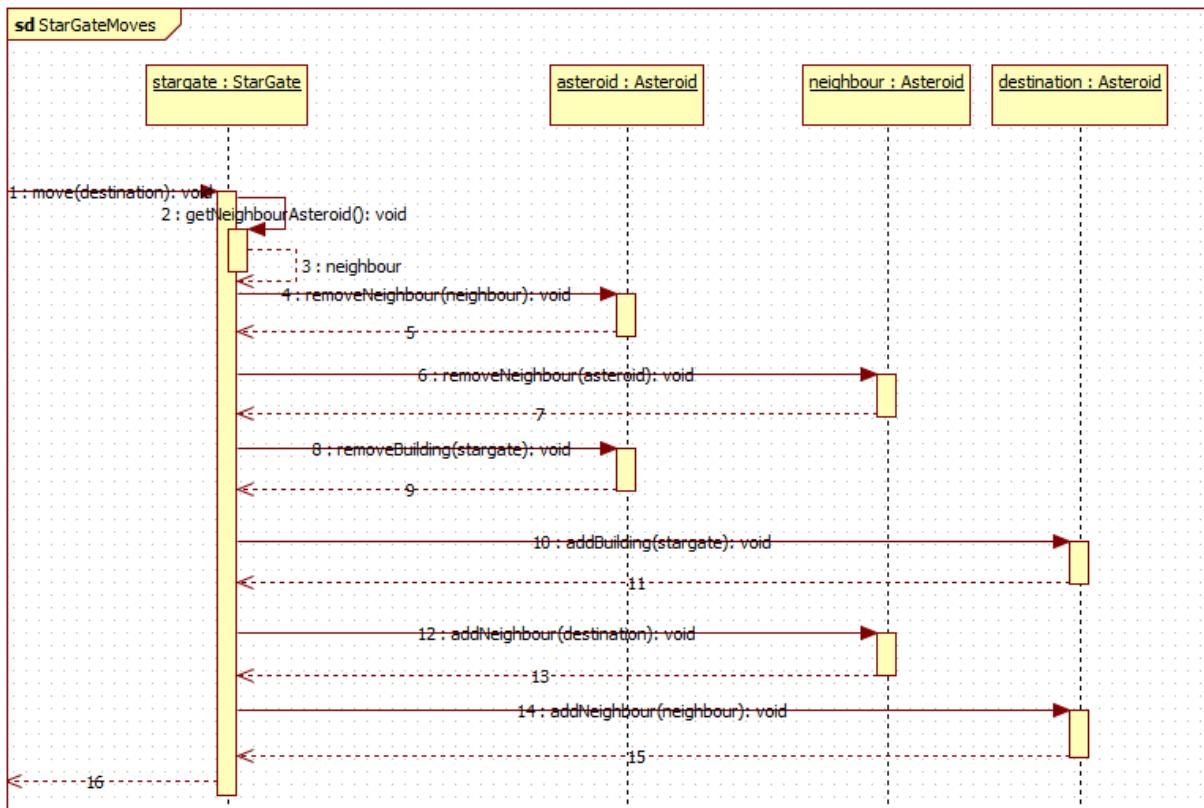
7.1.3.3 Drill With Robot



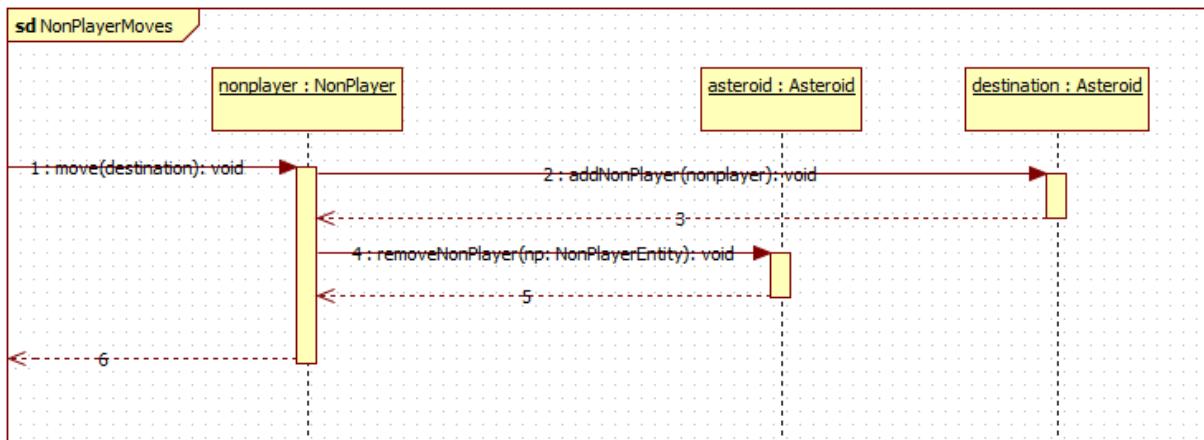
7.1.3.4 Drill With Robot Evaporate



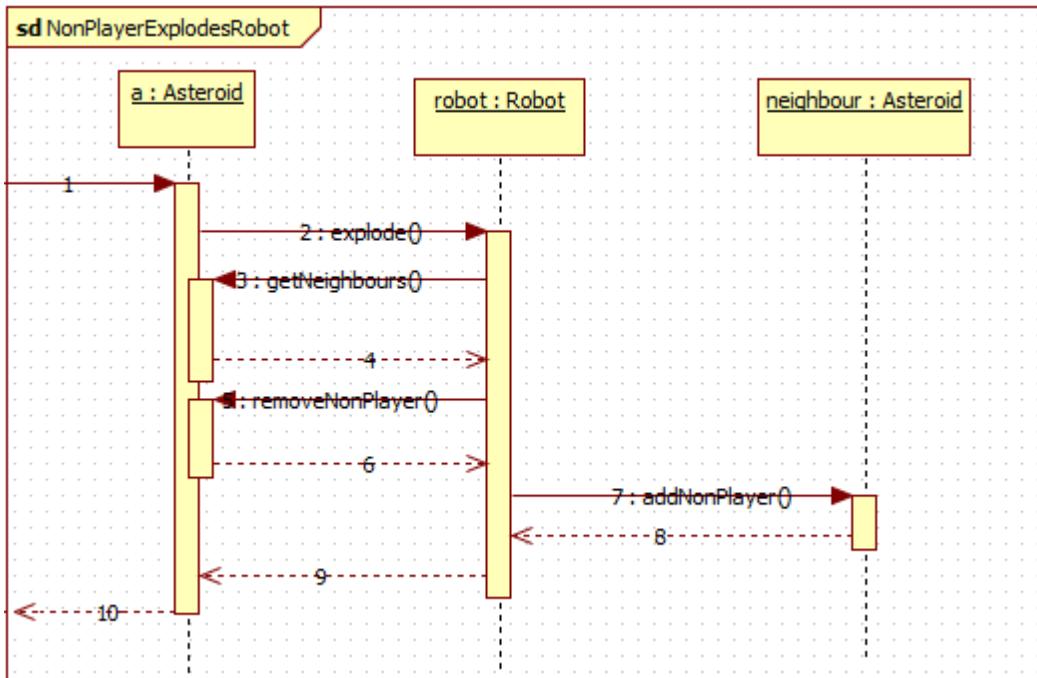
7.1.3.5 Star Gate Moves



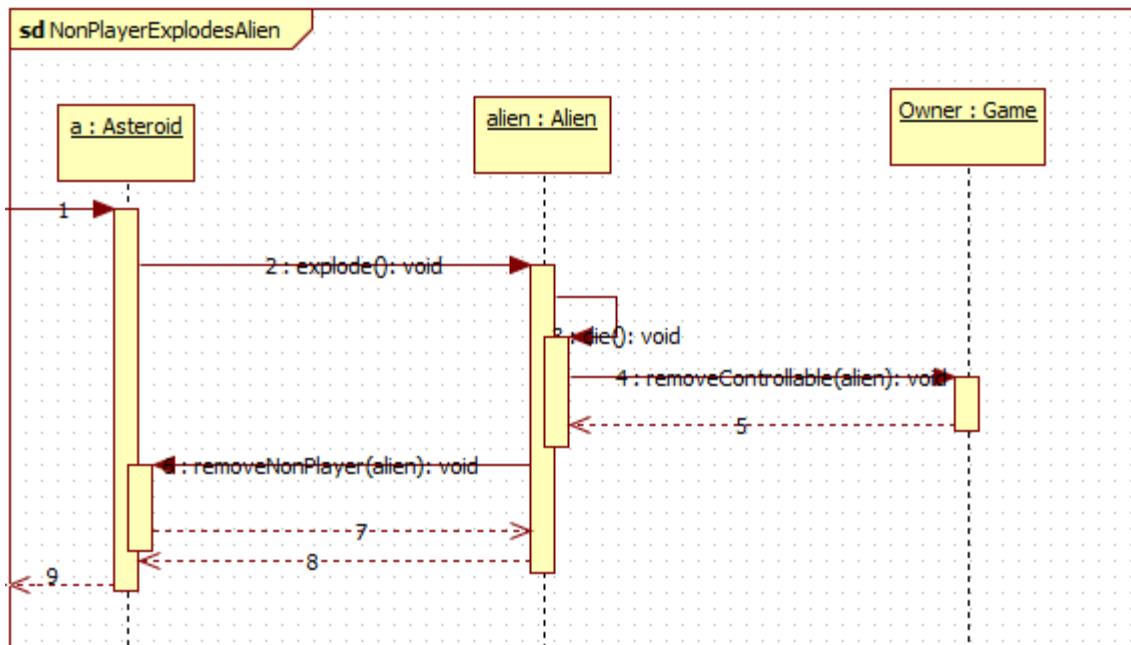
7.1.3.6 NonPlayer Moves



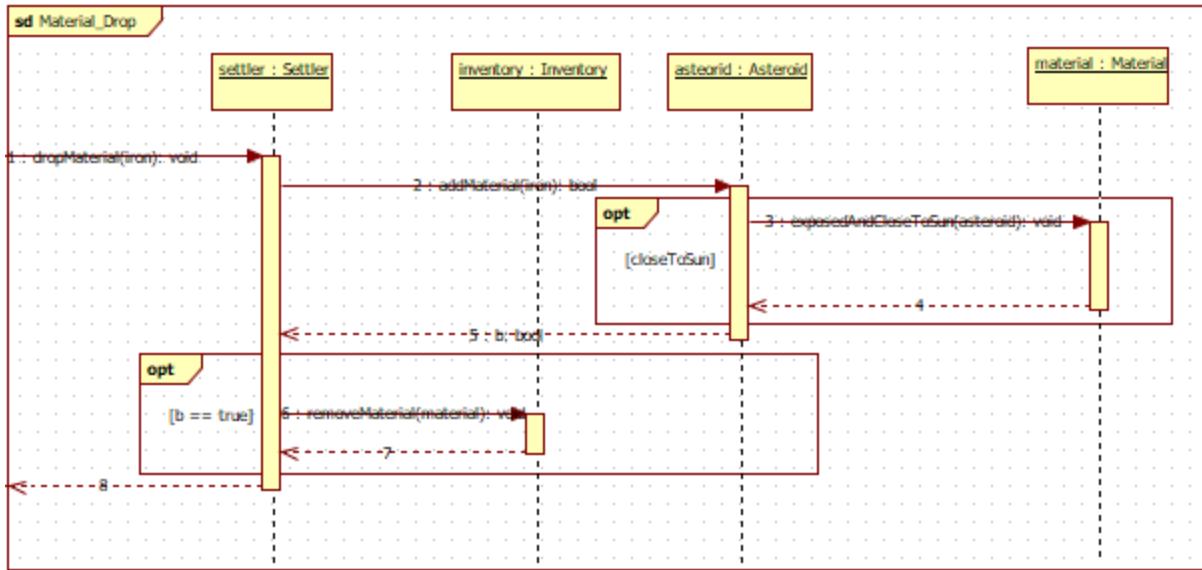
7.1.3.7 NonPlayer Explode Robot



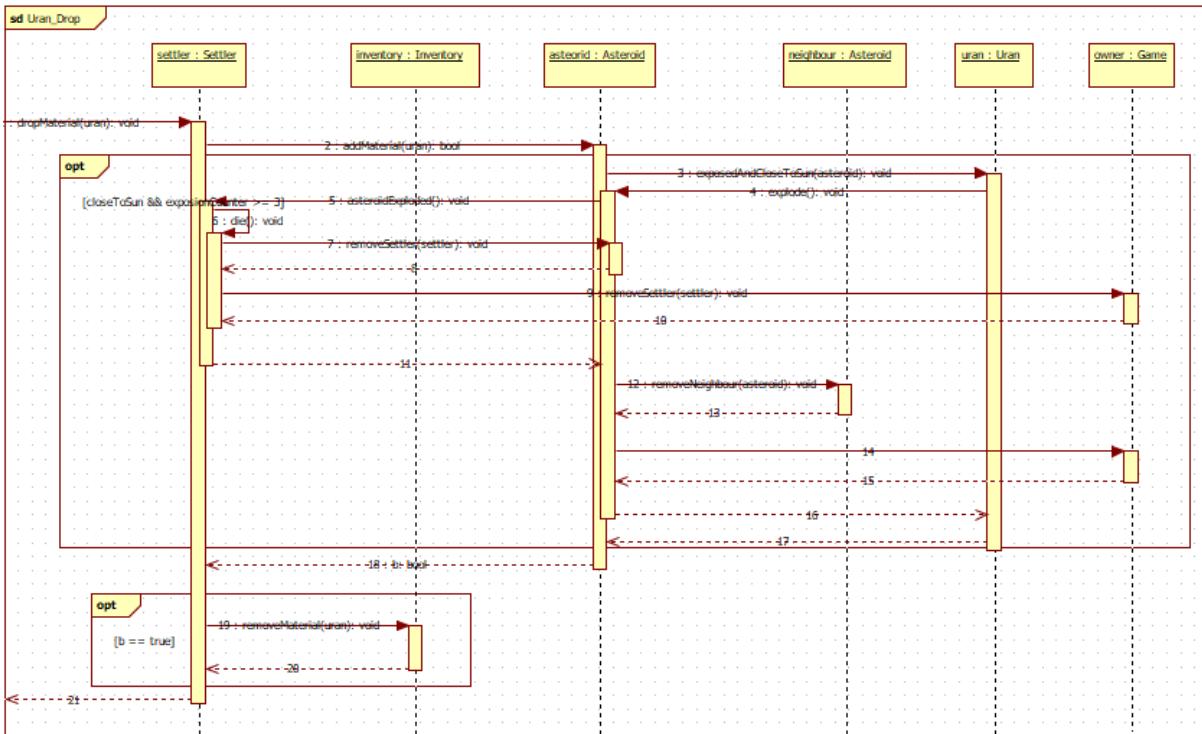
7.1.3.8 NonPlayer Explode Alien



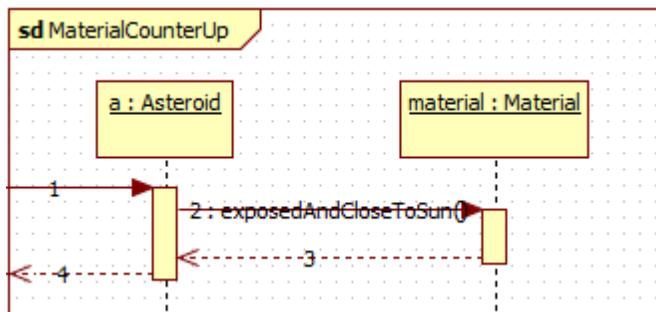
7.1.3.9 Material Drop



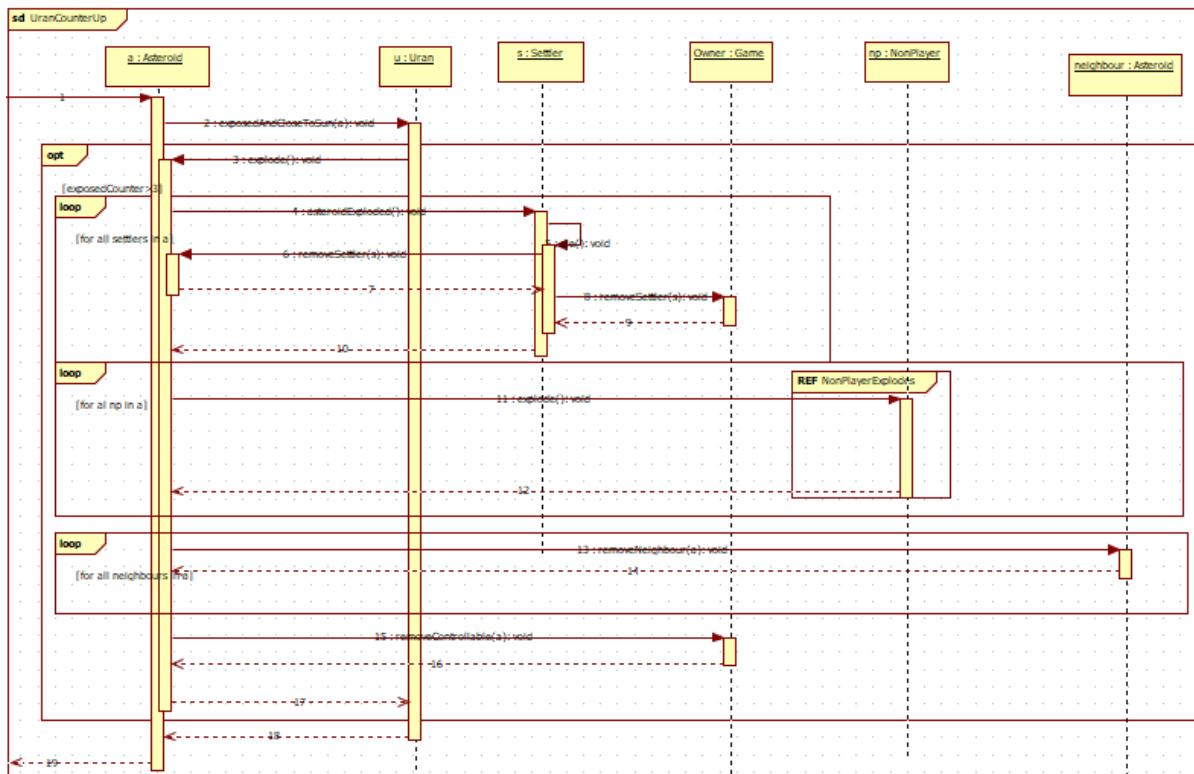
7.1.3.10 Uran Drop



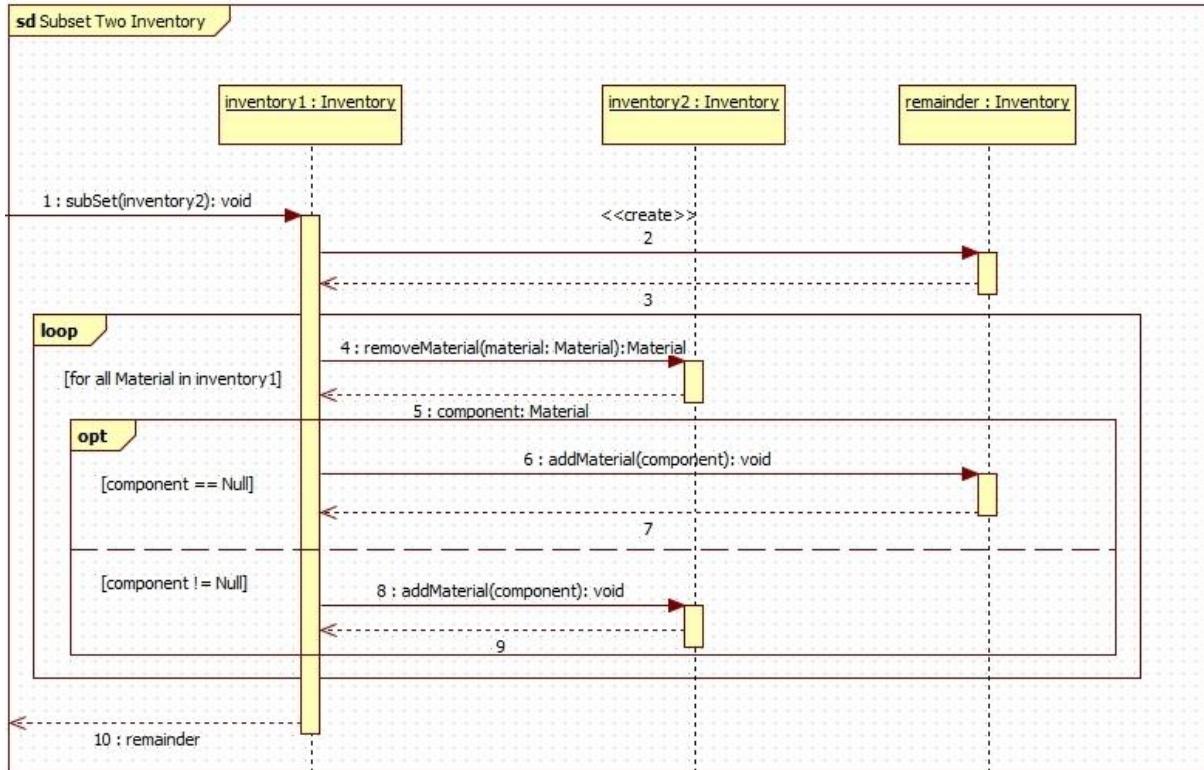
7.1.3.11 Material Counter Up



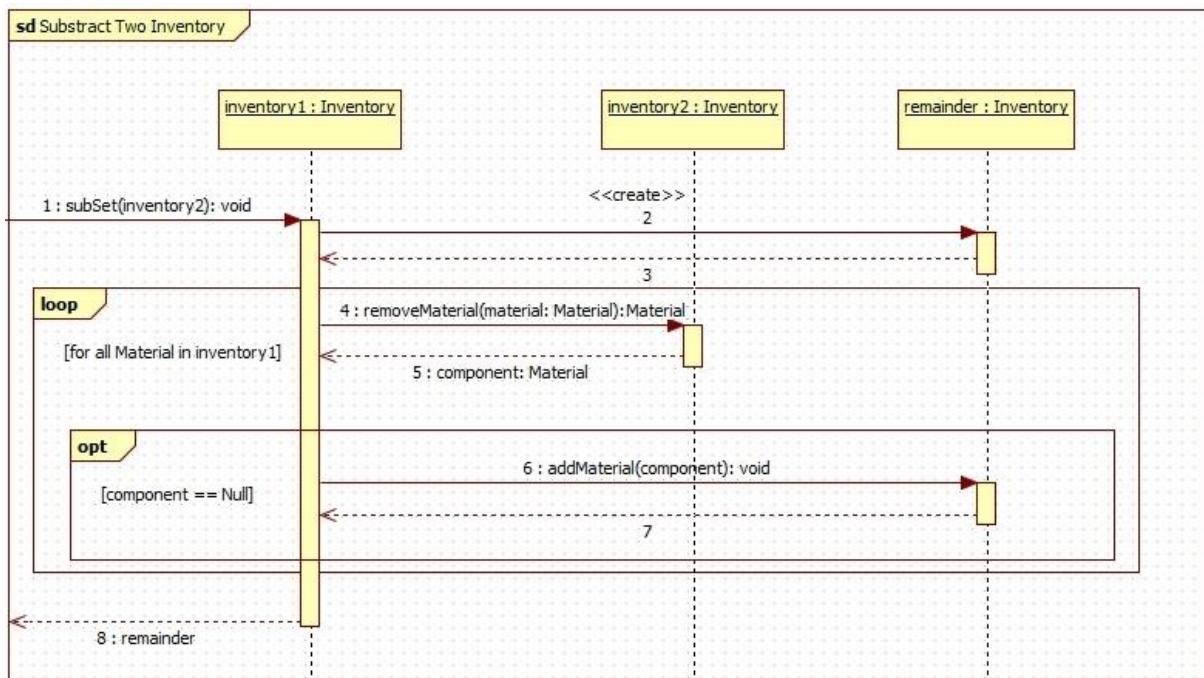
7.1.3.12 Uran Counter Up



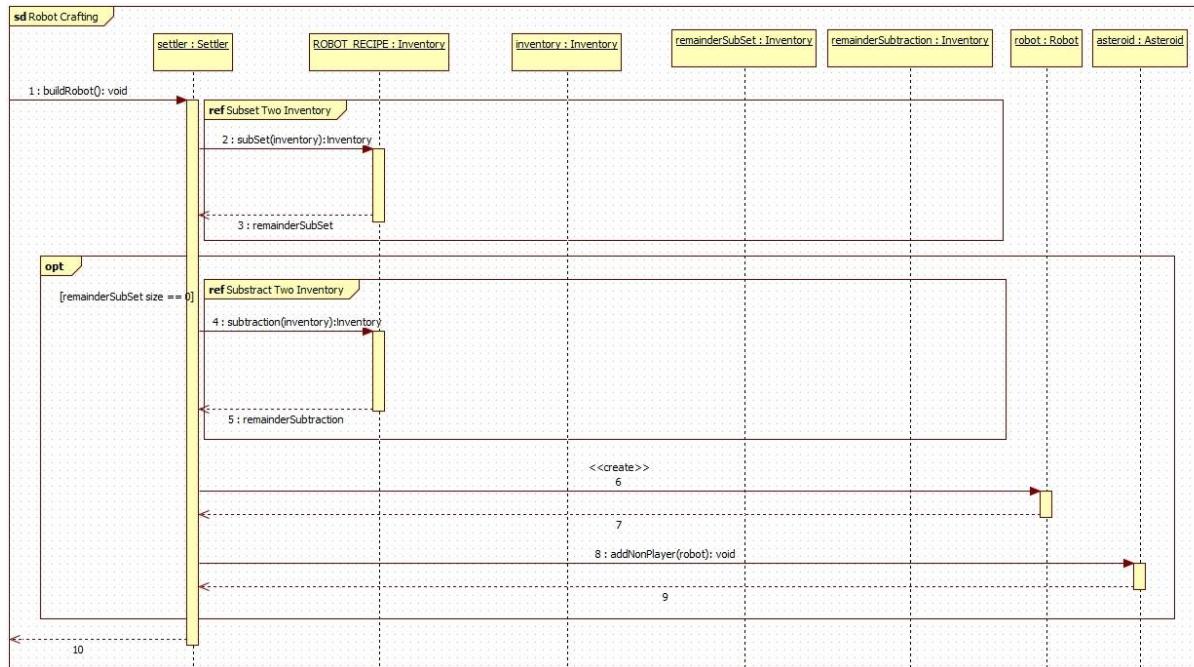
7.1.3.13 Subset Two Inventory



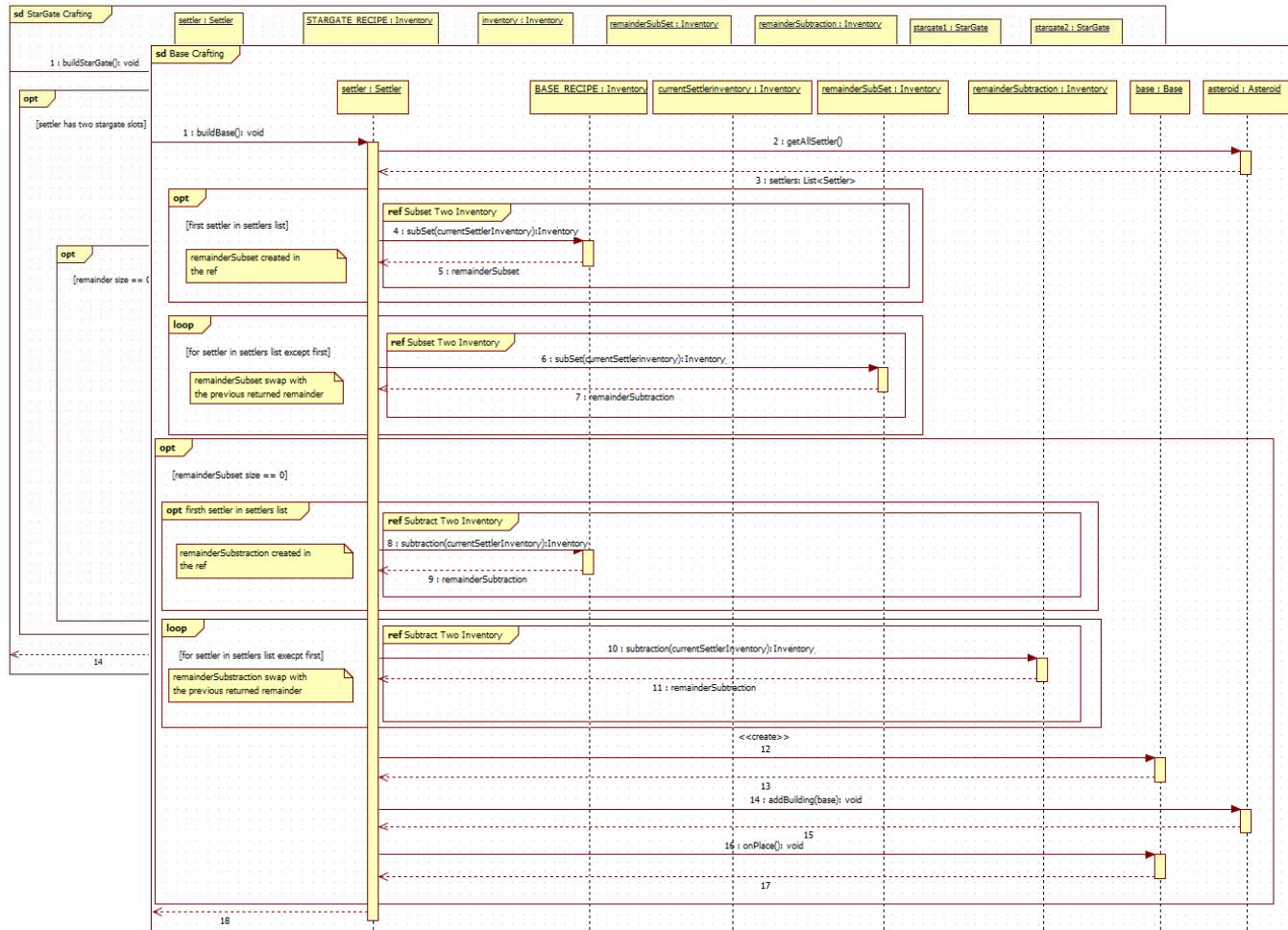
7.1.3.14 Substract Two Inventor



7.1.3.15 Robot Crafting

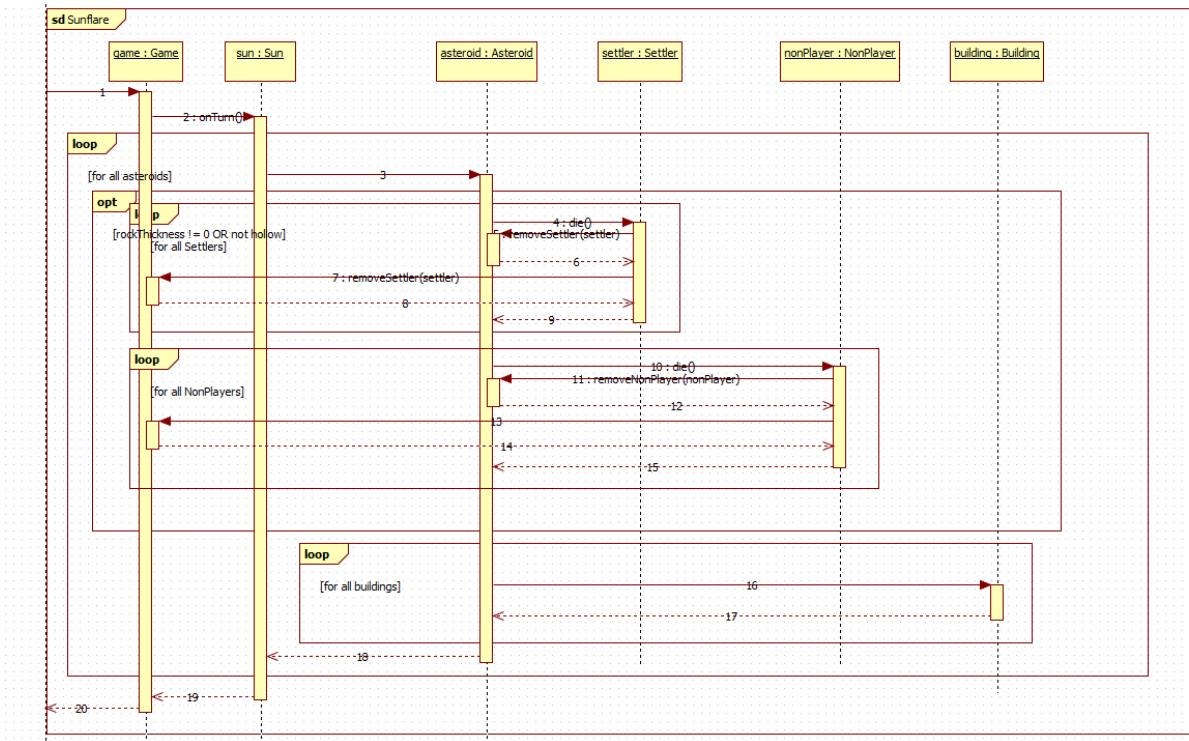


7.1.3.16 StarGate Crafting



7.1.3.17 Base Crafting

7.1.3.18 Sunflare



7.2 Prototípus interface-definíciója

7.2.1 Az interfész általános leírása

Parancssoros interface, amihez alább definiáltuk a parancsokat

7.2.2 Bemeneti nyelv

Incializálás:

init game <manually, random>

Leírás: Játék létrehozása.

manually: Alapértelmezésben minden determinisztikusra állít.

load game <file name>

Leírás: Játék betöltése.

create settler <id>

create robot <id>

create alien <id>

create asteroid <id>

create sun <id>

create inventory <id>

create coal <id>

create ice <id>

create iron <id>

create uran <id>

create stargate <id>

create base <id>

Leírás: Egyes elemek létrehozása.

id: Az objektum egyedi azonosítója.

type: A nyersanyag típusa. Értélei: ice, iron, coal, uran

set settler <property name> <settler id> <new value>

settler id: a telepes azonosítója

property name: egy telepes tulajdonsága, lehetséges értékei:

hadactionthisturn (akkor a new value t/f)

asteroid (akkor a new value az aszteroida id-ja)

inventory (akkor a new value az inventory id-ja)

set robot <property name> <robot id> <new value>

robot id: a robot azonosítója

property name: egy robot tulajdonsága, lehetséges értékei:

asteroid (akkor a new value az aszteroida id-ja)

set alien <property name> <alien id> <new value>

alien id: az alien azonosítója

property name: egy alien tulajdonsága, lehetséges értékei:

asteroid (akkor a new value az aszteroida id-ja)

set asteroid <property name> <asteroid id> <new value>

property name: egy aszteroida tulajdonsága, lehetséges értékei:

closetosun (akkor a new value t/f)

rockthickness (ekkor a new value egy szám)

set stargate <property name> <stargate id> <new value>

property name: egy kapu tulajdonsága, lehetséges értékei:

asteroid (ekkor a new value az aszteroida id-ja)

neighbour (ekkor a new value a szomszédos kapu id-ja)

set base <property name> <building id> <new value>

property name: egy kapu tulajdonsága, lehetséges értékei:

asteroid (ekkor a new value az aszteroida id-ja)

add asteroid settler <asteroid id> <settler id>

Leírás: Egy telepest ad hozzá az aszteroidához.

add asteroid nonplayer <nonplayer id> <settler id>

Leírás: Egy nem játékos által irányított entitást ad hozzá az aszteroidához.

nonplayer id: Egy nem játékos által irányított entitás azonosítója.

add asteroid neighbour <asteroid id> <neighbour id>

Leírás: Egy szomszédöt ad hozzá az aszteroidához.

neighbour id: A szomszéd aszteroida azonosítója.

add asteroid building <asteroid id> <building id>

Leírás: Egy szomszédöt ad hozzá az aszteroidához.

building id: Az épület azonosítója.

add asteroid material <asteroid id> <material id>

Leírás: Egy nyersanyagot ad az aszteroidához.

material id: A nyersanyag azonosítója.

add sun asteroid <sun id> <asteroid id>

Leírás: Egy aszteroidát ad hozzá a naphoz.

set material exposed <material id> <new value>

Leírás: Egyes elemek felfedettségének beállítása.

new value: Egy szám, ami a felfedettséget jelöli.

add inventory material <inventory id> <material id>

Leírás: Nyersanyag hozzáadása az inventoryhoz

material id: A nyersanyag azonosítója.

add inventory stargate <inventory id> <stargate id>

Leírás: Teleportkapu hozzáadása az inventoryhoz

stargate id: A teleportkapu azonosítója.

set alien deterministic behaviour <id> <t/f>

set robot deterministic behaviour <id> <t/f>

set sun deterministic behaviour <id> <t/f>

set asteroid deterministic behaviour <id> <t/f>

set stargate deterministic behaviour <id> <t/f>

Leírás: Determinisztikus viselkedés beállítása.

start game

Leírás: Játék indítása.

save game <file name>

Leírás: Játék fájlba mentése.

quit

Leírás: Program leállítása.

Játék parancsai:***start next turn***

Leírás: Lezárja az aktuális kört. Hatására megkezdődik az új kör.

list all settlers

Leírás: Kilistázza a játékban lévő összes telepes azonosítóját és aszteroidáját, ahol éppen tartózkodik.

Opciók: -

list settler <id> info

Leírás: Kilistázza az adott telepes azonosítóját, aszteroidáját, aszteroidájának belső állapotát és az inventory-jának tartalmát, belső állapotát. minden olyan információt visszaad, ami egy lépés eldöntéséhez szükséges lehet a játékos számára.

Opciók:

-id: A telepes azonsítója.

move settler <id> dest <dest. asteroid id / stargate id>

Leírás: Elmozgatja az adott telepet egy adott szomszédos aszteroidára.

Opciók:

-id: A telepes azonsítója.

-dest. asteroid id: Egy szám, ami azonosítja a szomszédos célaszteroidát.

-stargate id: Azt a kaput azonosítja, amire rálép a telepes.

drill settler <id>

Leírás: Az adott azonosítójú telepessel fúr az aszteroida felszínén.

Opciók:

-id: A telepes azonsítója.

mine settler <id>

Leírás: Az adott telepes kibányássza az adott aszteroidában lévő nyersanyagot.

Opciók:

-id: A telepes azonsítója.

drop settler <id> material <material id>

Leírás: Az adott telepes eldobja a megjelölt nyersanyagot.

Opciók:

-id: A telepes azonsítója.

-material id: Egy szám, ami azonosítja az adott nyersanyagot.

build settler <id> <obj.>

Leírás: Az adott telepes felépíti a kijelölt objektumot az aszteroidáján.

Opciók:

-id: A telepes azonsítója.

-obj: Az adott objektumot azonosító szöveg. (robot, stargate, base)

place gate settler <id> <gate id>

Leírás: Az adott telepes lehelyezi a megadott kaput a jelenlegi aszteroidájára.

Opciók:

- id: A telepes azonsítója.
- gate id: Az adott kaput azonosító szám.

Nem játékos által irányított objektumok parancsai:

Ezen parancsokat az adott objektum azonnal végrehajtja, ha az inicializálásnál determinisztikusra állítottuk a viselkedést.

move robot <id> dest <dest. asteroid id / stargate id >

Leírás: Elmozgatja az adott robotot egy adott szomszédos aszteroidára.

Opciók:

- id: A robot azonsítója.
- dest. asteroid id: Egy szám, ami azonosítja a szomszédos célaszteroidát.
- stargate id: Azt a kaput azonosítja, amire rálép a robot.

move alien <id> dest <dest. asteroid id / stargate id >

Leírás: Elmozgatja az adott alienet egy adott szomszédos aszteroidára.

Opciók:

- id: Az alien azonsítója.
- dest. asteroid id: Egy szám, ami azonosítja a szomszédos célaszteroidát.
- stargate id: Azt a kaput azonosítja, amire rálép az alien.

drill robot <id>

Leírás: Az adott azonosítójú robottal fúr az aszteroida felszínén.

Opciók:

- id: A robot azonsítója.

mine alien <id>

Leírás: Az adott alien-nel kibányássza az adott aszteroidában lévő nyersanyagot.

Opciók:

- id: Az alien azonsítója.

change close to sun <id>

Leírás: Az adott aszteroida napközelsége az ellentétre változik.

Opciók:

- id: Az aszteroida azonsítója.

sunflare <id>

Leírás: Az adott nap kitör.

Opciók:

- id: A nap azonsítója.

Játék mentése file-ba:

save

Leírás: Elmenti a játék állását a „game.txt”-be.

game.txt leírása: A játék egyes szerializált objektumai szerepelnek benne ;-vel elválasztva.

(Jelmagyarázat: <bool> - t / f)

Settler:

hadActionThisTurn: <bool>

```

id: <id>
asteroid: <asteroid id>
inventory: <inv id>
;
```

Inventory:

```

id: <id>
materials:
<material id>
<material id>
...
stargates
<stargate id>
<stargate id>
...
;
```

Material: (Közös struktúra minden nyersanyagra)

```

class: <Type>
id: <id>
exposedCounter: <int>
isRadio: <bool>
canEvaporate: <bool>
;
```

Asteroid:

```

id: <id>
rockThickness: <bool>
closeToSun: <bool>
isRandom: <bool>
capacity: <int>
neighbours:
<neighbour id>
...
settlers:
<settler id>
...
nonPlayers:
<nonPlayer id>
...
materials:
<material id>
...
buildings:
<building id>
...
;
```

Sun:

```
id: <id>
```

```
isRandom: <bool>
asteroids:
<asteroid id>
...
;
```

```
NonPlayer:
class: <Type>
id: <id>
asteroid: <asteroid id>
;
```

load

Betölti a game.txt tartalmát. Ennek megfelelően inicializálja a játékot.

7.2.3 Kimeneti nyelv

list turn events

Leírás: A parancs hatására kiíródik egy rövid összefoglalás az elmúlt kör eseményeiről. Ennek tartalma a következő bejegyezésekkel állhat:

```
material exploded <id>
material evaporated <id>
material mined <id>
material dropped <id>
asteroid exploded <id>
asteroid drilled <id>
asteroid mined <id>
asteroid sun distance changed <id>
settler died <id>
alien died <id>
robot died <id>
stargate broke <id>
settler move <id> <destination asteroid id>
robot move <id> <destination asteroid id>
alien move <id> <destination asteroid id>
stargate move <id> <destination asteroid id>
sunflare on sun <sun id>
robot crafted <robot id>
stargate crafted <stargate id>
base crafted <base id>
```

list all

Leírás: Meghívja az összes objektumot az alább leírt list info parancsokat. Először a napokon, aztán aszteroidákon iterál végig, majd következnek az épületek, a telepesek, azután minden egyéb entitás (robot, alien).

list sun <id> info

Kiírás:

Sun id: <id>
isRandom: <t / f>

asteroids:

asteroid: <asteroid id>

 ...

list asteroid <id> info

Kiírás:

Asteroid id: <id>

rockThickness: <thickness>

closeToSun: <t / f>

isRandom: <t / f>

materials:

material <material id>

 ...

neighbours:

neighbour: <asteroid id>

buildings:

building: <building id>

settlers:

settler: <settler id>

 ...

nonPlayers:

nonPlayer: <nonPlayer id>

 ...

list stargate <id> info

Kiírás:

StarGate id: <id>

isRandom: <t / f>

works: <t / f>

wasInSunFlare: <t / f>

neighbour: <neighbour gate id>

asteroid: <asteroid id>

list Base <id> info

Kiírás:

Base id: <base id>

asteroid: <asteroid id>

list settler <id> info

Kiírás:

Settler id: <id>

hadActionThisTurn: <t / f>

asteroid: <asteroid id>

inventory:

materials:

material: <material id>

 ...

starGates:

starGate: <starGate id>

list robot <id> info**Kiírás:****Robot id: <id>****isRandom: <t / f>****asteroid: <asteroid id>*****list alien <id> info*****Kiírás:****Alien id: <id>****isRandom: <t / f>****asteroid: <asteroid id>**

7.3 Összes részletes use-case

Use-case neve	Játékban lévő telepesek listázása.
Rövid leírás	A játékos kíllistázza, hogy jelenleg hány telepes található a pályán és hol.
Aktorok	Játékos
Forgatókönyv	A játékos beírja a list all settlers parancsot és visszakapja a parancs kimenetét.

Use-case neve	Egy telepes állapotának listázása.
Rövid leírás	A játékos kíllistázza, egy adott telepes belső állapotát.
Aktorok	Játékos
Forgatókönyv	A játékos beírja a list settler <id> info parancsot és visszakapja a parancs kimenetét.

Use-case neve	Lépés egy adott telepessel.
Rövid leírás	A játékos lép egy adott telepessel, egy megadott szomszédos aszteroidára.
Aktorok	Játékos
Forgatókönyv	1a. A játékos beírja a move settler <id> dest <dest asteroid id> parancsot, ahol megadja melyik telepessel szeretne lépni, melyik szomszédos aszteroidára. 1b. Ha a játékos érvénytelen paramétereket ad meg, akkor erről tájékoztatjuk őt. 1c. Ha a játékos nem tud átlépni a megadott aszteroidára, értesítük őt erről.

Use-case neve	Fúrás egy adott telepessel.
Rövid leírás	A játékos fúr egy adott telepessel, azon az aszteroidán, ahol éppen áll.
Aktorok	Játékos
Forgatókönyv	1a. A játékos beírja a drill settler <id> parancsot, ahol megadja melyik telepessel szeretné fúrni. 1b. Ha a játékos érvénytelen paramétereket ad meg, akkor erről tájékoztatjuk őt.

	1c. Ha a játékos uránt, vagy vízjeget fúrt értesítjük erről őt.
--	---

Use-case neve	Bányászás egy adott telepessel.
Rövid leírás	A játékos bányászik egy adott telepessel, azon az aszteroidán, ahol éppen áll.
Aktorok	Játékos
Forgatókönyv	1a. A játékos beírja a mine settler <id> parancsot, ahol megadja melyik telepessel szeretne bányászni. 1b. Ha a játékos érvénytelen paramétereket ad meg, akkor erről tájékoztatjuk őt. 1c. Ha a játékos nem tud bányászni az adott aszteroidán, értesítjük erről őt.

Use-case neve	Egy nyersanyag eldobása.
Rövid leírás	Játékos eldobja egy telepes egy adott nyersanyagát.
Aktorok	Játékos
Forgatókönyv	1a. A játékos beírja a drop settler <id> material <material id> parancsot, ahol megadja melyik telepesnek, melyik nyersanyagát szeretné eldobni. 1b. Ha a játékos érvénytelen paramétereket ad meg, akkor erről tájékoztatjuk őt. 1c. Ha a játékos nem tudja eldobni az adott nyersanyagot az aszteroidán, értesítjük erről őt.

Use-case neve	Egy nyersanyag eldobása.
Rövid leírás	Játékos eldobja egy telepes egy adott nyersanyagát.
Aktorok	Játékos
Forgatókönyv	1a. A játékos beírja a drop settler <id> material <material id> parancsot, ahol megadja melyik telepesnek, melyik nyersanyagát szeretné eldobni. 1b. Ha a játékos érvénytelen paramétereket ad meg, akkor erről tájékoztatjuk őt. 1c. Ha a játékos nem tudja eldobni az adott nyersanyagot az aszteroidán, értesítjük erről őt.

Use-case neve	Egy objektum építése.
Rövid leírás	Játékos felépít egy objektumot egy adott telepessel, a telepes aszteroidáján.
Aktorok	Játékos
Forgatókönyv	1a. A játékos beírja a build settler <id> <obj.> parancsot, ahol megadja melyik telepessel mit szeretne építeni. 1b. Ha a játékos érvénytelen paramétereket ad meg, akkor erről tájékoztatjuk őt. 1c. Ha a játékos nem tudja felépíteni az adott objektumot, értesítjük erről őt.

Use-case neve	Egy kapu lehelyezése.
Rövid leírás	Játékos lehelyez egy kaput
Aktorok	Játékos
Forgatókönyv	1a. A játékos beírja a place gate settler <id> <gate id> parancsot, ahol megadja melyik telepessel melyik kaput szeretné lehelyezni. 1b. Ha a játékos érvénytelen paramétereket ad meg, akkor erről tájékoztatjuk őt.

Use-case neve	A játék elmentése.
Rövid leírás	Játékos elmenti a játék egy adott állását egy szöveges file-ba.
Aktorok	Játékos
Forgatókönyv	1a. A játékos beírja a save parancsot, amivel elmenti a játék állapotát.

Use-case neve	A játék betöltése.
Rövid leírás	Játékos elmenti a játék egy adott állását egy szöveges file-ba.
Aktorok	Játékos
Forgatókönyv	1a. A játékos beírja a load parancsot, amivel elmenti a játék állapotát.

7.4 Tesztelési terv

Teszt-eset neve	Mozgás telepessel
Rövid leírás	Egy telepessel átlépünk egy másik aszteroidára.
Teszt célja	Annak vizsgálata, hogy a telepes tud-e mozogni aszteroidák között.

Teszt-eset neve	Mozgás telepessel nem szomszédos aszteroidára.
Rövid leírás	Egy telepessel megpróbálunk átlépní egy nem szomszédos aszteroidára.
Teszt célja	Annak vizsgálata, hogy a telepes valóban nem tud nem szomszédos aszteroidára lépni.

Teszt-eset neve	Fúrás telepessel
Rövid leírás	Egy telepessel fúrunk az aszteroidája felszínén.
Teszt célja	Annak vizsgálata, hogy a telepes valóban megfúrja-e aszteroidájának felszínét és a sziklaréteg csökken-e.

Teszt-eset neve	Fúrás telepessel, radioaktív aszteroidán, napközelben.
Rövid leírás	Egy telepessel fúrunk egy radioaktív aszteroida felszínén.
Teszt célja	Annak vizsgálata, hogy a telepes valóban megfúrja-e radioaktív aszteroidájának felszínét és a sziklaréteg csökken-e. Azt is itt vizsgáljuk, hogy ha az utolsó sziklaréteget átfúrjuk, az urán valóban öregszik, illetve ha már 3-szor volt napközelben felrobban.

Teszt-eset neve	Fúrás telepessel napközelben, vízjeget tartalmazó aszteroidán.
Rövid leírás	Egy telepessel fúrunk egy vízjeget tartalmazó, napközelben lévő aszteroida felszínén.
Teszt célja	Annak vizsgálata, hogy a telepes valóban megfúrja-e aszteroidájának felszínét és a sziklaréteg csökken-e. Azt is itt vizsgáljuk, hogy ha az utolsó sziklaréteget átfúrjuk, a vízjég valóban elpárolog-e, mivel az aszteroida napközelben van.

Teszt-eset neve	Bányászás telepessel.
Rövid leírás	Egy telepes bányászik az aszteroidájából.
Teszt célja	Annak vizsgálata, hogy a telepes valóban képes-e bányászni az aszteroidájából és a kibányászott nyersanyagot eltárolja-e.

Teszt-eset neve	Bányászás telepessel, teli tárolóval.
Rövid leírás	Egy telepes bányászna az aszteroidájából, de nem tud, mivel nem fér több nyersanyag a tárolójába.
Teszt célja	Annak vizsgálata, hogy a telepes valóban nem képes-e 10-nél több nyersanyagot eltárolni.

Teszt-eset neve	Telepes visszahelyez egy nyersanyagot.
Rövid leírás	Egy telepes visszahelyez egy nyersanyagot az aszteroidájába.
Teszt célja	Annak vizsgálata, hogy a telepes valóban képes-e visszahelyezni egy nyersanyagot az aszteroidájába, ami üreges.

Teszt-eset neve	Telepes nem tud visszahelyezni egy nyersanyagot.
Rövid leírás	Egy telepes nem tud visszahelyezni nyersanyagot az aszteroidájába.
Teszt célja	Annak vizsgálata, hogy a telepes valóban nem tud-e visszahelyezni nyersanyagot egy olyan aszteroidába, aminek felszíne nincs átfúrva, vagy már van benne valami.

Teszt-eset neve	Telepes visszahelyez egy uránt napközelben.
Rövid leírás	Egy telepes visszahelyez egy uránt az aszteroidájába, napközelben.
Teszt célja	Annak vizsgálata, hogy miután a telepes visszahelyezte az uránt az aszteroidájába, valóban öregedett az urán, vagy ha szükséges felrobban.

Teszt-eset neve	Telepes visszahelyez egy vízjeget napközelben.
Rövid leírás	Egy telepes visszahelyez egy vízjeget az aszteroidájába, napközelben.
Teszt célja	Annak vizsgálata, hogy miután a telepes visszahelyezte a vízjeget az aszteroidájába, az valóban elpárolgott-e.

Teszt-eset neve	Telepes épít egy robotot.
Rövid leírás	Egy telepes megépít egy robotot az aszteroidán, ahol tartózkodik.
Teszt célja	Annak vizsgálata, hogy a telepes a megfelelő nyersanyagok birtokában képes-e létrehozni egy robotot, illetve ezeket a nyersanyagokat felhasználja-e.

Teszt-eset neve	Telepes nem tud építeni egy robotot.
Rövid leírás	Egy telepes megpróbál megépíteni egy robotot, de nem sikerül neki.
Teszt célja	Annak vizsgálata, hogy a telepes a megfelelő nyersanyagok birtoklása nélkül valóban nem képes-e létrehozni egy robotot.

Teszt-eset neve	Telepes épít egy teleportkapu párát.
Rövid leírás	Egy telepes megépít egy teleportkapu párát az aszteroidán, ahol tartózkodik.
Teszt célja	Annak vizsgálata, hogy a telepes a megfelelő nyersanyagok birtokában képes-e létrehozni egy teleportkapu párát, illetve ezeket a nyersanyagokat felhasználja-e.

Teszt-eset neve	Telepes nem tud építeni egy teleportkapu párát.
Rövid leírás	Egy telepes megépítene egy teleportkapu párát az aszteroidán, ahol tartózkodik, de nem jár sikkerrel.
Teszt célja	Annak vizsgálata, hogy a telepes a teleportkapu párhoz szükséges helyek nélkül valóban nem képes-e létrehozni a teleportkapukat.

Teszt-eset neve	Telepes épít egy bázist
Rövid leírás	Egy telepes épít egy bázist az aszteroidáján.
Teszt célja	Annak vizsgálata, hogy a telepes képes-e megépíteni egy bázist, illetve ezután megnyeri-e a játékot.

Teszt-eset neve	Telepes lehelyezi az első teleportkaput.
Rövid leírás	Egy telepes lehelyezi az első teleportkaput egy párból.
Teszt célja	Annak vizsgálata, hogy a telepes képes-e lehelyezni egy teleportkapu pár első tagját.

Teszt-eset neve	Telepes lehelyezi második teleportkaput.
Rövid leírás	Egy telepes lehelyezi a második teleportkaput egy párból.
Teszt célja	Annak vizsgálata, hogy a telepes képes-e lehelyezni a második teleportkaput egy párból és ekkor a két kapu összekötődik-e.

Teszt-eset neve	Telepes lehelyezné a második teleportkaput.
Rövid leírás	Telepes lehelyezné a második teleportkaput, de az első megsemmisült
Teszt célja	Annak vizsgálata, hogy amikor a telepes lehelyezi egy megsemmisült teleportkapu második tagját, az a megsemmisült példánnyal valóban nem lép-e kapcsolatba, nem lép működésbe.

Teszt-eset neve	Aszteroida felrobban
Rövid leírás	Egy aszteroida felrobban a rajta álló telepesekkel, épületekkel és alienekkel együtt.
Teszt célja	Annak vizsgálata, hogy amikor egy aszteroida felrobban a rajta álló telepesek, alienek és épületek valóban megsemmisülnek.

Teszt-eset neve	Aszteroida felrobban, szomszéddal.
Rövid leírás	Egy aszteroida felrobban a és a rajta álló robot átlép a szomszédos aszteroidára.
Teszt célja	Annak vizsgálata, hogy amikor egy aszteroida felrobban a rajta álló robot valóban átkerül a szomszédos aszteroidára.

Teszt-eset neve	Aszteroida felrobban, szomszéd nélkül.
Rövid leírás	Egy szomszédok nélküli aszteroida felrobban és a rajta álló robot megsemmisül.
Teszt célja	Annak vizsgálata, hogy amikor egy szomszédok nélküli aszteroida felrobban a rajta álló robot valóban megsemmisül.

Teszt-eset neve	Napvihar történik.
Rövid leírás	Egy nem üreges aszteroidát napvihar ér.
Teszt célja	Annak vizsgálata, hogy amikor napvihar ér egy nem üreges aszteroidát a rajta lévő telepesek, robotok, alienek valóban megsemmisülnek-e és a rajta lévő teleportkapu elromlik-e.

Teszt-eset neve	Üreges napvihar történik.
Rövid leírás	Egy üreges aszteroidát napvihar ér.
Teszt célja	Annak vizsgálata, hogy amikor napvihar ér egy üreges aszteroidát a rajta lévő telepesek, robotok, alienek valóban eltudnak-e bújni.

Teszt-eset neve	Robot lép.
Rövid leírás	Egy robot átlép egy szomszédos aszteroidára.
Teszt célja	Annak vizsgálata, hogy egy robot valóban átléphet-e egy szomszédos aszteroidára.

Teszt-eset neve	Robot lépne.
Rövid leírás	Egy robot szeretne lépni egy aszteroidára, ami nem szomszédos.
Teszt célja	Annak vizsgálata, hogy egy robot valóban nem tud átlépní egy nem szomszédos aszteroidára.

Teszt-eset neve	Robot fúr.
Rövid leírás	Egy robot az aszteroidája felszínén.
Teszt célja	Annak vizsgálata, hogy egy robot valóban tud-e fúrni az aszteroidája felszínén.

Teszt-eset neve	Alien lép.
Rövid leírás	Egy alien átlép egy szomszédos aszteroidára.
Teszt célja	Annak vizsgálata, hogy egy alien valóban átlép-e egy szomszédos aszteroidára.

Teszt-eset neve	Alien bányászik.
Rövid leírás	Egy alien kibányászik egy nyersanyagot az aszteroidájából.
Teszt célja	Annak vizsgálata, hogy egy alien valóban tud-e bányászni az aszteroidájából

Teszt-eset neve	Alien bányászna.
Rövid leírás	Egy alien szeretne kibányászni egy nyersanyagot az aszteroidájából.
Teszt célja	Annak vizsgálata, hogy egy alien valóban nem tud-e bányászni az aszteroidájából abban az esetben, ha nincs átfűrva az aszteroida.

7.5 Tesztelést támogató segéd- és fordítóprogramok specifikálása

Java Development Kit

7.6 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2021. 03. 26. 16:00	3,5 óra	Bárkányi Kárpáti Mohácsi Simon Tóth	Értekezlet. Megvitattuk milyen osztályokat és metódusokat érintenek a változtatások és felosztottuk egymás között a szekvenicákat.
2021. 03. 27. 16:00	4 óra	Bárkányi Kárpáti Mohácsi Simon Tóth	Mindenki elmondta hogyan módosította a szekvenciákat, megvitattuk a módosult osztálydiagramot. Belekezdtünk a beviteli nyelv megvitatásába is.
2021. 03. 28. 16:00	4 óra	Bárkányi Kárpáti Mohácsi Simon Tóth	Megvitattuk a nyelvet, annak parancsait, amivel majd a felhasználó kommunikál a játékkal. Valamint a visszajelzések, file-ok formátumát.
2021. 03. 29. 10:00	2 óra	Bárkányi Kárpáti Mohácsi Simon Tóth	Dokumentum összevágása, elkészítése.

8. Részletes tervezet

67 – brainstormers

Konzulens:
Simon Balázs

Csapattagok

Mohácsi Márton

Simon Zoltán

Kárpáti Márk András

Bárkányi Csaba

Tóth Ádám

T8ZT88

HRSNUG

O1BG0Z

BQI7QQ

QEYOW2

moha.mohacsi@gmail.com

simonzolika1999@gmail.com

mark.karpati1999@gmail.com

csaba.barkanyi.csaba@gmail.com

toth99adam@gmail.com

2021-04-12

Tartalomjegyzék

8.	Részletes tervezet	125
8.1	Osztályok és metódusok tervei.....	125
8.1.1	Alien	125
8.1.2	Asteroid.....	126
8.1.3	Base	128
8.1.4	Building	129
8.1.5	Coal.....	130
8.1.6	Drilling.....	131
8.1.7	Entity	131
8.1.8	Game.....	132
8.1.9	Ice.....	133
8.1.10	Inventory	134
8.1.11	Iron	135
8.1.12	Material.....	136
8.1.13	Mining.....	136
8.1.14	Moving.....	137
8.1.15	NonPlayer.....	137
8.1.16	Planet	138
8.1.17	Printable.....	140
8.1.18	ReactsToSunflare.....	140
8.1.19	Robot.....	141
8.1.20	Settler.....	142
8.1.21	StarGate.....	143
8.1.22	Sun.....	145
8.1.23	UI.....	146
8.1.24	Uran.....	146
8.2	A tesztek részletes tervezet, leírásuk a teszt nyelvén	147
8.2.1	Bányászás telepessel	147
8.2.2	Bányászás telepessel, teli tárolóval	148
8.2.3	Telepes visszahelyez egy nyersanyagot.....	150
8.2.4	Telepes nem tud visszahelyezni egy nyersanyagot.....	151
8.2.5	Telepes visszahelyez egy uránt napközelben	152
8.2.6	Telepes visszahelyez egy vízjeget napközelben	153
8.2.7	Alien bányászik	154
8.2.8	Alien bányászna.....	155
8.1.1	Nem elvégezhető Robot Crafting.....	156
8.1.2	Elvégezhető Robot Crafting.....	157
8.1.3	Nem elvégezhető Stargate Crafting.....	158
8.1.4	Elvégezhető Stargate Crafting	160
8.1.5	Nem elvégezhető Base Crafting egy telepessel.....	161
8.1.6	Nem elvégezhető Base Crafting telepesek külön aszteroidán	163
8.1.7	Elvégezhető Base Crafting	165
8.1.8	Első Stargate lehelyezése a párból	167
8.1.9	Második Stargate lehelyezése a párból.....	168
8.2	Napvihar kitörése nem átfúrt aszteroidán.....	170
8.3	Napvihar kitörése átfúrt aszteroidán	173
8.4	Fűrás robottal.....	175
8.5	Fűrás robottal, elpárolog	176

8.6	Fúrás robottal, felrobban.....	178
8.7	Fúrás robottal, felrobban, nincs szomszéd.....	179
8.8	Fúrás telepessel.....	181
8.9	Fúrás telepessel, elpárolog	182
8.10	Fúrás telepessel, felrobban.....	183
8.11	Mozgás robottal, kudarc	185
8.12	Mozgás robottal, siker.....	186
8.13	Mozgás telepessel, kudarc	187
8.14	Mozgás telepessel, siker.....	189
8.15	Telepes átlép egy kapun.	190
8.16	Teleportkapu elromlik.	192
8.3	A tesztelést támogató programok tervei	194
8.4	Napló	195

8. Részletes tervezettségek

8.1 Osztályok és metódusok tervezettségei.

8.1.1 Alien

8.1.1.1 Felelősségek

Ez az osztály felel az űrlényért és függvényeiért. Pl.: hogy az űrlény vegyen fel nyersanyagot.

8.1.1.2 Ősosztályok

Printable->Entity->NonPlayer->Alien

8.1.1.3 Interfészek

ReactsToSunflare, Moving, Mining

- Attribútumok

2. **-id: int:** Ez az adott Alien id-je, ezt arra használjuk fel, hogy az Alien-t azonosítani tudjuk.
3. **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
4. **-asteroid:Asteroid:** Ezen az aszteroidán van az Alien.
5. **-owner:Game:** Ez az osztály tárolja le az Alien-t
6. **-inventory:Inventory:** Ebben tárolja az Alien általa össze gyűjtött nyersanyagokat.

8.1.1.4 Metódusok

- **+getUIString(): String:** Ez a függvény generálja az adott Alien fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén.
- **+getNotifiedAboutSunflare():** Ez a függvény az osztály die függvényét hívja, amennyiben az asteroidea-ja nem igaz értékkel tér vissza az isHollow függvényre
- **+move(destination: Asteroid):** egy függvény, aszteroidáról aszteroidára való mozgást végezti.

```
If(destination is reachable from NonPlayer.asteroid) {
    destination.add(NonPlayer)
    NonPlayer.asteroid.remove(NonPlayer)
    NonPlayer.asteroid=destination
}
```

- **+move(destinationGate: StarGate):** ez a függvény, amely elvégzi azokat a mozgással kapcsolatos dolgokat ahol egy teleport kaput használunk, amiket a mozgásra képes osztályoknak kell.

```
If(destinationGate is active) {
    destinationGate.neighbour.asteroid.add(NonPlayer)
    NonPlayer.asteroid.remove(NonPlayer)
    NonPlayer.asteroid= destinationGate.neighbour.asteroid
}
```

- `+die()`: ez a függvény kiveszi a Alien-t az aszteroidáról, és jelzi a játéknak, hogy kivehető a létező elemekből.
- `+asteroidExploded()`: ez a függvény meghívja az Alien die függvényét.
- `+onTurn()`: Ez a függvény hajtja végre az Alien veselkedését a körön belül. Egyik körben mine függvényt hívja, a másikban az asteroida-jának véletlenszerűen kiválasztott szomszédjával a move függvényét. Majd ezt ismételgeti.
- `+mine()`: egy függvény, amely kivesz egy Material-t az aszteroidából, ha az aszteroida kérge 0.Osztály2

8.1.2 Asteroid

8.1.2.1 Felelősség

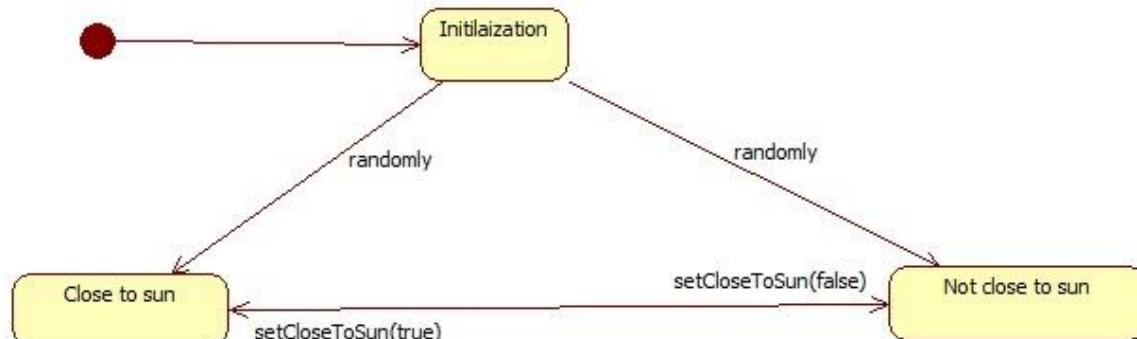
Ez az osztály felel a játékban található aszteroidák feladataiért, pl.: amikor napkitörés van ez az osztály hívja meg a rajta található entitásokon s épületeken a `getNotifiedAboutSunflare()` függvényt, ezzel jelezve nekik hogy épp napkitörés van.

8.1.2.2 Állapotdiagrammok

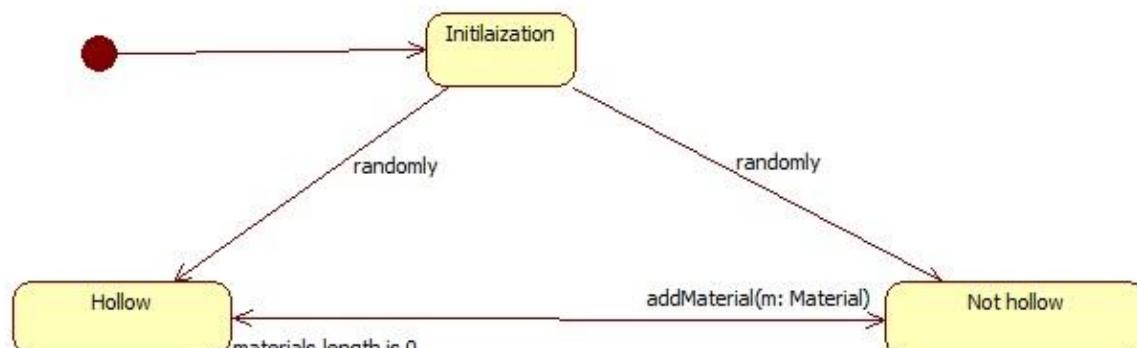
- Viselkedés



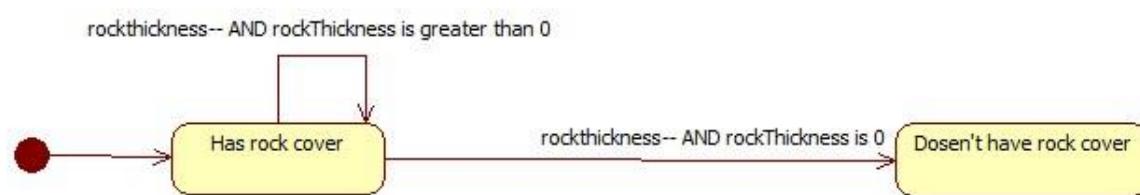
- Napközelség



- Üreges



- Kifúrt



(A felső feltétel a balról jobbra menő nyilakhoz tartozik, az alsó pedig a jobbról balra menő nyilakhoz.)

8.1.2.3 Ősosztályok

Printable -> Planet -> Asteroid.

8.1.2.4 Interfészek

Controllable, ReactsToSunflare

8.1.2.5 Attribútumok

- **-id: String:** Ez az adott aszteroida id-je, ezt arra használjuk fel, hogy a játékos ki tudja választani a megfelelő aszteroidát.
- **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
- **-owner: Game:** Egy referencia a Game osztályhoz.
- **-neighbours: Asteroid[0...*]:** Ez a lista tárolja az aszteroidával „fizikailag” (nem StarGate-n keresztül) szomszédos aszteroidákat.
- **-materials: Material:** Ez a lista tárolja az aszteroidában található materialokat, ha ez a lista null akkor az aszteroida üreges.
- **-settlers: Settler:** Ez a lista tárolja az aszteroidán található settlereket.
- **-nonPlayers: NonPlayer[0...*]:** Ez a lista tárolja az aszteroidán található nonPlayereket.
- **-buildings: Building[0...*]:** Ez a lista tárolja az aszteroidán található épületeket.
- **-rockThickness: int:** Ez az int tárolja az aszteroida kérgenek vastagságát.
- **-closeToSun: Boolean:** Ez a bool azt jelzi, hogy az aszteroida éppen közel van-e a napjához.
- **-isRandom: Boolean:** Ez a bool azt jelzi, hogy az aszteroida random módon, vagy determinisztikusan működik-e (napközelbe kerülés esély számlálás közben), ez csak a teszteléshez kell.
- **-capacity: int:** Ez az int azt jelzi, hogy az aszteroidába mennyi darab nyersanyag fér bele.
- **-sun: Sun:** Ez tárolja a napot amihez az aszteroida tartozik

8.1.2.6 Metódusok

- **+genUIString(): String:** Ez a függvény generálja az adott Asteroid fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az Asteroid id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén. A generált string az asteroid id-jét, rockThickness értékét, closeToSun értékét, isRandom értékét, a materials-ban található materialok genUIString által generált stringjeit, a neighbours-ban található aszteroidák id-jét, a settlers-ben található settlerek genUIString által generált stringjeit, illetve a nonPlayers-ben található nonPlayerek genUIString által generált stringjeit.

- **+onTurn():** Egy absztrakt függvény mely elvégzi az adott aszteroida egy körben elvégzendő feladatait: kisorsolja hogy ebben a körben ez az aszteroida változtatja-e a napközelséget (ha napközelben van akkor kikerül-e onnan s ha nincs napközelbe akkor bekerül-e oda) s ha igen akkor beállítja a closeToSun változót ennek megfelelően. Ezen kívül, ha egy aszteroida kérge 0 vastag akkor a benne található matrialon (ha van) meghívja az exposedAndCloseToSun() függvényt.
- **+getNotifiedAboutSunflare():** Ez a függvény akkor hívódik meg az aszteroidán amikor azt napvihar éri, ilyenkor az aszterida meghívja a getNotifiedAboutSunflare() függvényt az összes épületen, telepesen s nonPlayeren ami rajta van.
- **+addSettler(s: Settler):** Hozzáadja a settlers listához az s Settler-t.
- **+addNonPlayer(np: NonPlayer):** Hozzáadja a nonPlayers listához az np NonPlayert.
- **+removeso(s: Settler):** Kiveszük a settlers listából az s Settler-t.
- **+removeNonPlayer(np: NonPlayer):** Kiveszük a nonPlayers listából az np NonPlayert.
- **+explode():** Ez a függvény akkor hívódik meg amikor az aszteroida felrobban, ilyenkor meghívja a rajta található entityken az asteroidExploded() függvényt, illetve meghívja a rajta található épületeken a destroy() függvényt.
- **+drilling():** Ez a függvény akkor hívódik meg amikor valami fűr az aszteroidán, az aszteroida kérge csökken egyel (ha tud), illetve ha épp 0-ra csökken akkor meghívja az aszteroidában találhat nyersanyagok egyikén az exposedAndCloseToSun() függvényt.
- **+removeMaterial(): Material:** Ez a függvény eltávolít egy nyersanyagot az aszteroida belsejiből (ha tud) s azt visszaadja.
- **+addMaterial(m: Material): Boolean:** Ez a függvény megpróbálja belehelyezni az aszteroida belsejébe az m nyersanyagot, a sikerességgel azonos boollal tér vissza. A nyersanyag visszahelyezése csak akkor sikertelen, ha az aszteroida kérge nagyobb mint nulla, vagy hogyha az aszteroida elérte a kapacitását (materials.length() == capacity)
- **+removeNeighbour(a: Asteroid):** Kiveszük a neighbours listából az a Asteroidot.
- **+addNeighbour(a: Asteroid):** Hozzáadja a neighbours listához az a Asteroidot.
- **+removeBuilding(b: Building):** Kiveszük a buildings listából a b Buildinget.
- **+addBuilding(b: Building):** Hozzáadjuk a buildings listához a b Buildinget.
- **+isHollow(): Boolean:** Ez a függvény le ellenőrzi, hogy az aszteroida belseje üreges-e (üres-e a Materials lista) s ennek megfelelő boollal tér vissza.
- **+evaporateMaterial():** Ez a függvény elpárollogtatja a belsejében található nyersanyagokat. Tehát kiveszi őket a materials listából.
- **+initialize():** Ez a függvény inicializálja az aszteroidát. Kisorsolja, hogy az aszteroida belsejében milyen nyersanyag legyen (esetleg üres legyen-e), illetve, hogy ebből a nyersanyagból mennyi legyen benne. Ezen kívül azt is kisorsolja, hogy az aszteroida épp napközelben van-e.

8.1.3 Base

8.1.3.1 Felelősség

Ez az osztály felel a bázis épületekért s azoknak feladataiért pl.: amikor ráhelyezzük őket egy aszteroidára.

8.1.3.2 Ősosztályok

Printable -> Building -> Base

8.1.3.3 Interfészek

ReactsToSunflare

8.1.3.4 Attribútumok

- **-id: String:** Ez az adott bázis id-je, ezt arra használjuk fel, hogy a játékos ki tudja választani a megfelelő bázist. Ez most a bázisban nincsen használva, de ha a továbbiakban bővül a specifikáció akkor használhatjuk.
- **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
- **-asteroid: Asteroid:** Az aszteroida, amin a bázis található.
- **-owner: Game:** Egy referencia a Game osztályhoz, ezen keresztül fog a bázis jelezni a játéknak, ha le lett helyezve s ezzel megnyerte a játékos a játékot.

8.1.3.5 Metódusok

- **+genUIString(): String:** Ez a függvény generálja az adott bázis fontos tulajdonságait tartalmazó stringet (ennek legelső eleme a bázis id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén. A generált string az aszteroida id-jét tartalmazza.
- **+getNotifiedAboutSunflare():** Ez a függvény akkor hívódik meg a bázison amikor azt napvíhar éri. Ez most a bázisban nem lesz használva, de ha a továbbiakban bővül a specifikáció akkor használhatjuk.
- **+onPlace(a: Asteroid):** Ez a függvény végrehajtja a bázis feladatait abban a pillanatban amikor ráhelyezzük az aszteroidára, ez az hogy szól a game osztálynak, hogy meg lett nyerve a játék. Mindezt úgy érjük el, hogy a game osztályban meghívjuk a gameWon() függvényt.
- **+destroy():** Ez a függvény végrehajtja a bázis feladatait amikor az elpusztul. Ez most a bázisban nem lesz használva, de ha a továbbiakban bővül a specifikáció akkor használhatjuk.

8.1.4 Building

8.1.4.1 Felelősség

Ez egy absztrakt osztály mely a játékban található épületekért felel. Elpusztítja őket, illetve elvégzi a dolgukat amikor rákerülnek egy aszteroidára.

8.1.4.2 Ősosztályok

Printable -> Building

8.1.4.3 Interfészek

ReactsToSunflare, Controlable

8.1.4.4 Attribútumok

- **-id: String:** Ez az adott épület id-je, ezt arra használjuk fel, hogy a játékos ki tudja választani a megfelelő objektumot, pl: amikor egy settlerrel StarGate-re lép, akkor a StarGate id-jét megadva választhatja ki melyikre kíván rálépni.
- **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.

- **#asteroid: Asteroid:** Az aszteroida, amin az épület található.
 - **#boolean:hadActionThisTrun:** ez true, ha már léptetve volt az épület.
- owner: Game:** Egy referencia a Game osztályhoz.

8.1.4.5 Metódusok

- **+genUIString(): String:** Ez a függvény generálja az adott épület fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén. A Building osztályban ez egy absztrakt függvény.
- **+getNotifiedAboutSunflare():** Ez a függvény akkor hívódik meg az épületen amikor azt napvihar éri, ilyenkor az épület reagál arra ha kell neki.
- **+onTurn():** Ez ez absztrakt fv, ezt valósítják meg a leszármazotak, és hogy mit csináljanak egy adott körben.
- **+onPlace(a: Asteroid):** Ez a függvény végrehajtja az épület feladatait abban a pillanatban amikor rátávolítják az aszteroidára.
- **+destroy():** Ez a függvény végrehajtja az épület feladatait amikor az elpusztul.

8.1.5 Coal

8.1.5.1 Felelősség

Ez az osztály felel a coal viselkedésért.

8.1.5.2 Ősosztályok

Printable->Material

8.1.5.3 Interfészek

-

8.1.5.4 Attribútumok

7. **-id: int:** Ez az adott Material id-je, ezt arra használjuk fel, hogy az Material -t azonosítani tudjuk.
8. **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
9. **#isRadio: boolean:** Ez a változó mondja meg, hogy radioaktív-e az adott nyersanyag.
10. **#Name:String:** Ez tárolja az adott nyersanyag nevét.
11. **#canEvaporate:boolean:** Ez a változó mondja meg, hogy elpárolog-e a nyersanyag ha napközelbe kerül
12. **-exposedCounter:int:** Ez tárolja számon, hogy milyen hosszan volt kitéve a napnak az adott nyersanyag.

8.1.5.5 Metódusok

13. **+getUIString(): String:** Ez a függvény generálja az adott Material fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén.
14. **+exposedAndCloseToSun(a:Asteroid):** Ez a függvény hívódik, amikor napközelben felszínen van a nyersanyag. A paraméterül kapott Asteroid aminek a felszínén van. Növeli exposedCounter-t

8.1.5.6 Controllable

8.1.5.7 Felelősség

Ez egy olyan interface, amit olyan osztályok valósítanak meg miknek van autonóm működése pl.: a robotok melyek minden körben mozognak/fúrnak/nem csinálnak semmit.

8.1.5.8 Ősosztályok

-

8.1.5.9 Interfészek

-

8.1.5.10 Attribútumok

-

8.1.5.11 Metódusok

- `+onTurn()`: Egy absztrakt függvény mely elvégzi az adott objektum egy körben elvégzendő feladatait.

8.1.6 Drilling

8.1.6.1 Felelősség

Ez az interfész azok az osztályok valósítják meg, amik tudnak fúrni.

8.1.6.2 Ősosztályok

-

8.1.6.3 Interfészek

-

8.1.6.4 Attribútumok

-

8.1.6.5 Metódusok

- `int drill()`: Egy absztrakt függvény amely elvégzi azokat a fúrással kapcsolatos dolgokat, amiket a fúrásra képes osztályoknak kell

8.1.7 Entity

8.1.7.1 Felelősség

Ez az absztrakt Ősosztály felel azokért, akik meg tudnak halni, és érinti őket valahogy az aszteroida felrobbanása.

8.1.7.2 Ősosztályok

Printable->Entity

8.1.7.3 Interfészek

ReactstoSunflare, Moving

8.1.7.4 Attribútumok

- **-id: int:** Ez az adott Entity id-je, ezt arra használjuk fel, hogy az entity-t azonosítani tudjuk.
- **int nextId():** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.

8.1.7.5 Metódusok

- **String getUIString():** Ez a függvény generálja az adott entity fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap a játékosnak kiír megfelelő parancs esetén. Az Entity osztályban ez egy absztrakt függvény.
- **getNotifiedAboutSunflare():** Ez a függvény akkor hívódik meg az entity-n amikor azt napvihar éri, ilyenkor az reagál arra ha kell neki.
- **void move(destination: Asteroid):** egy absztrakt függvény, amely elvégzi azokat a mozgással kapcsolatos dolgokat ahol a cél egy másik aszteroida, amiket a mozgásra képes osztályoknak kell.
- **void move(destinationGate: StarGate):** egy absztrakt függvény, amely elvégzi azokat a mozgással kapcsolatos dolgokat ahol egy teleport kaput használunk, amiket a mozgásra képes osztályoknak kell megvalósítani.
- **void die():** ez az absztrakt függvény végzi el a meghalással kapcsolatos dolgokat. Ez az Entity-ben absztrakt.
- **void asteroidExploded():** ez az absztrakt függvény biztosítja, hogy hogyan viselkedjen az osztály, amikor alatta egy aszteroida felrobban.

8.1.8 Game

8.1.8.1 Felelősség

Ez a függvény összefogja a játékot, felel a játék megnyeréséért s elvesztéséért, kontrollálja a játék által irányított objektumokat, illetve végrehajtja az UI-tól kapott parancsokat.

8.1.8.2 Ősosztályok

-

8.1.8.3 Interfészek

-

8.1.8.4 Attribútumok

- **ui: UI:** Az egyetlen UI amely a játékossal kommunikál, a game ennek visszaadhatja az egyes objektumok státuszát, amit aztán az UI kiírhat a játékosnak.
- **settlers: Settler[0...*]:** A játékban létező összes Settler tároló tömb.
- **controllables: Controllable[0...*]:** A játékban létező összes Controllable tároló tömb.

8.1.8.5 Metódusok

- **void removeSettler(s: Settler):** Kiveszi a settlers listából az s Settlert.
- **void removeControllable(c: Controllable):** Kiveszi a controllables listából az s Controllablet.
- **void addSettler(s: Settler):** Hozzáadja a settlers listához az s Settlert.
- **void addControllable(c: Controllable):** Hozzáadja a controllables listához a c Controllablet.
- **void gameWon():** Elvégzi a játék megnyerésekor elvégzendő feladatokat: üzenet a játékosnak az UI-n keresztül, a game-ben tárolt listák kiürítése.
- **boolean isGameLost():** Leellenőrzi, hogy el van-e veszítve a játék s ennek megfelelő boollal tér vissza. A játék akkor van elveszítve amikor az összes settler meghalt, tahát ezt az ellenőrzést elvégezhetjük a settlers lista hosszának megnézésével.
- **void gameLost():** Elvégzi a játék elvesztésekor elvégzendő feladatokat: üzenet a játékosnak az UI-n keresztül, a game-ben tárolt listák kiürítése.
- **void startTurn():** Ez a függvény akkor hívódik meg amikor a játékos kiadja a lépés a következő körre parancsot, ezután minden controllable autonóm módon (az onTurn() függvényének segítségével) elvégzi az adott körben elvégzendő dolgát. minden Controllable-re a game osztályban meghívjuk az onTurn függvényét.
- **void listAllSettlers(): String:** Ez a függvény visszaad egy stringet amiben az összes settler minden fontos tulajdonsága található, ezt a stringet a UI írja ki a list all settlers parancs hatására. A függvény végigloopol a settlers listán s meghívja mindegyik tagjára a genUIString függvényt, majd ezeket összefűzve visszaadja.
- **void initGame(isManual: Boolean):** A játék inicializálását végző függvény. Ha az isManual igaz akkor a felhasználó fogja a create s set függvényekkel a játékban található összes objektumot inicializálni. Ha az isManual false akkor ez a függvény automatikusan létrehozza a játék objektumait, létrehozza a játékban található napokat, ezekhez tartozó aszteroidákat, majd létrehozza a telepeseket s ürlényeket s ezeket röhelyezi random módon az aszteroidákra. Az aszteroidát ezek után még inicializálja is az initialize() függvényével, ez azért kell mert az aszteroidáknak pl. többféle nyersanyaq lehet a belsejében, s ezek előtöntéséért felel az initialize() függvény. Mindezek után pedig random mennyiségű szomszédot hozzáadunk minden aszteroidához az összes aszteroida közül.
- **void loadGame(fileName: String):** Ez a függvény a fileName Stringgel meghatározott file-ból betölti a játékot. A file-ba a genUIString segítségével írtuk ki az objektuainkat, szóval a betöltés a genUIString viszafejtése alapján fog működni.
- **void startGame():** Ez a függvény elindítja a játékot. Ez jelzi az inicializálás befejeztét. Mivel a settlerek mindenkor legutoljára kerülnek sorra ezért a játékot egy startTurn() hívással kezdjük.
- **void saveGame(fileName: String):** Ez a függvény lementi a játék állását a fileName nevű fileba.
- **void list(arguments: String):** Ez a függvény kilistázza az arguments-ben található id-vel specifikált settler információt a genUIString függvény segítségével.

8.1.9 Ice

8.1.9.1 Felelősség

Ez az osztály az Ice viselkedésér-t felel

8.1.9.2 Ősosztályok

Printable->Material

8.1.9.3 Interfészek

-

8.1.9.4 Attribútumok

- **-id: int:** Ez az adott Material id-je, ezt arra használjuk fel, hogy az Material -t azonosítani tudjuk.
- **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
- **#isRadio: boolean:** Ez a változó mondja meg, hogy radioaktív-e az adott nyersanyag.
- **#Name: String:** Ez tárolja az adott nyersanyag nevét.
- **#canEvaporate: boolean:** Ez a változó mondja meg, hogy elpárolog-e a nyersanyag ha napközelbe kerül
- **-exposedCounter:int:** Ez tárolja számon, hogy milyen hosszan volt kitéve a napnak az adott nyersanyag.

8.1.9.5 Metódusok

- **String getUIString():** Ez a függvény generálja az adott Material fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén.
- **void exposedAndCloseToSun(a:Asteroid):** Ez a függvény hívodik, amikor napközelben felszínen van a nyersanyag. A paraméterül kapott Asteroid aminek a felszínén van. Meghívja a paraméterül kapott aszteroida evaporateMaterial függvényét.

8.1.10 Inventory

8.1.10.1 Felelősség

Ez az osztály felel a nyersanyagok, kapuk tárolásáért.

8.1.10.2 Ősosztályok

Printable

8.1.10.3 Interfészek

-

8.1.10.4 Attribútumok

- **-id: int:** Ez az adott Inventory id-je, ezt arra használjuk fel, hogy az Inventory -t azonosítani tudjuk.
- **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
- **-material:Material[0...10]** egy tömb ami Material típusú objektumokat táról. Ebben van számoltartva, hogy milyen nyersanyagok vannak a jelenlegi Inventory-ban.

- **-starGate:StarGate[0...3]:** egy tömb ami az elkészített StarGate-t tárolja

8.1.10.5 Metódusok

- **String getUIString():** Ez a függvény generálja az adott Inventory fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén.
- **Material removeMaterial(m:Material):** a megadott típusú Material-t kiveszi az Inventory-ból és vissza adja vissza térsí értékként.
- **StartGate removeStarGate():** kivesz egy StarGate-t az inventory-ból és azt adja vissza mint visszatérési érték.
- **Inventory subSet(i: Inventory):** ezt a függvényt hívjuk a receptnek, és azt kapjuk meg, hogy mi hiányzik még a recept elkészítéséhez.
- **Inventory subtraction(i: Inventory):** ezt a függvényt hívjuk a recepten amikor már ki is akarjuk szedni az adot inventory-ból a releváns nyersanyagokat.
- **void addStarGAtes(sg:StarGate):** hozzá ad egy StarGate-t az inventory-hoz.
- **void addMaterial(m:Material):** hozzá adja a megadott Material-t az Inventory-hoz.
- **StarGate[0..3] getStarGates():** a teleport kapukat adja vissza.

8.1.11 Iron

8.1.11.1 Felelősség

Ez az osztály felel a vas viselkedésért.

8.1.11.2 Ósosztályok

Printable->Material

8.1.11.3 Interfészek

-

8.1.11.4 Attribútumok

- **-id: int:** Ez az adott Material id-je, ezt arra használjuk fel, hogy az Material -t azonosítani tudjuk.
- **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
- **#isRadio: boolean:** Ez a változó mondja meg, hogy radioaktív-e az adott nyersanyag.
- **#Name: String:** Ez tárolja az adott nyersanyag nevét.
- **#canEvaporate: boolean:** Ez a változó mondja meg, hogy elpárolog-e a nyersanyag ha napközelbe kerül
- **-exposedCounter:int:** Ez tárolja számon, hogy milyen hosszan volt kitéve a napnak az adott nyersanyag.

8.1.11.5 Metódusok

- **String getUIString():** Ez a függvény generálja az adott Material fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén.

- **void exposedAndCloseToSun(a:Asteroid):** Ez a függvény hívódik, amikor napközelben felszínen van a nyersanyag. A paraméterül kapott Asteroid aminek a felszínén van. Meghívja a paraméterül kapott aszteroida evaporateMaterial függvényét.

8.1.12 Material

8.1.12.1 Felelősség

Ez az absztrakt osztály felel a nyersanyagokért és viselkedésükért.

8.1.12.2 Ősosztályok

Printable->Material

8.1.12.3 Interfészek

-

8.1.12.4 Attribútumok

15. **-id: int:** Ez az adott Material id-je, ezt arra használjuk fel, hogy az Material -t azonosítani tudjuk.
16. **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
17. **#isRadio: boolean:** Ez a változó mondja meg, hogy radioaktív-e az adott nyersanyag.
18. **#Name:String:** Ez tárolja az adott nyersanyag nevét.
19. **#canEvaporate:boolean:** Ez a változó mondja meg, hogy elpárolog-e a nyersanyag ha napközelbe kerül
20. **-exposedCounter:int:** Ez tárolja számon, hogy milyen hosszan volt kitéve a napnak az adott nyersanyag.

8.1.12.5 Metódusok

21. **+getUIString(): String:** Ez a függvény generálja az adott Material fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén.
22. **+exposedAndCloseToSun(a:Asteroid):** Ez a függvény hívóik, amikor napközelben felszínen van a nyersanyag. A paraméterül kapott Asteroid aminek a felszínén van. Növeli exposedCounter-t

8.1.13 Mining

8.1.13.1 Felelősség

Ez az interfész azok az osztályok valósítják meg, amik képesek nyersanyagot kibányászni.

8.1.13.2 Állapotdiagram

-

8.1.13.3 Ősosztályok

8.1.13.4 Interfészek

8.1.13.5 Attribútumok

8.1.13.6 Metódusok

- **+mine()**: egy absztrakt függvény, amely elvégzi a azokat a bányászással kapcsolatos dolgokat, amiket a bányászára képes osztályoknak kell.

8.1.14 Moving

8.1.14.1 Felelősség

Ez az interfész azok az osztályok valósítják meg, amik képesek mozogni.

8.1.14.2 Állapotdiagram

8.1.14.3 Ősosztályok

8.1.14.4 Interfészek

8.1.14.5 Attribútumok

8.1.14.6 Metódusok

- **+move(destination: Asteroid)**: egy absztrakt függvény, amely elvégzi azokat a mozgással kapcsolatos dolgokat ahol a cél egy másik aszteroida, amiket a mozgásra képes osztályoknak kell.
- **+move(destinationGate: StarGate)**: egy absztrakt függvény, amely elvégzi azokat a mozgással kapcsolatos dolgokat ahol egy teleport kaput használunk, amiket a mozgásra képes osztályoknak kell.

8.1.15 NonPlayer

8.1.15.1 Felelősség

Ez az absztrakt ősosztály felel a nem játékos álltal irányítót karakterekért

8.1.15.2 Ősosztályok

Printable->Entity->NonPlayer

8.1.15.3 Interfészek

ReactstoSunflare, Moving, Controllable

8.1.15.4 Attribútumok

23. **-id: int:** Ez az adott NonPlayer id-je, ezt arra használjuk fel, hogy az NonPlayer-t azonosítani tudjuk.
24. **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
25. **-asteroid:Asteroid:** Ezen az aszteroidán van a NonPlayer.
26. **-owner:Game:** Ez az osztály tárolja le a NonPlayer-t

8.1.15.5 Metódusok

- **+getUIString(): String:** Ez a függvény generálja az adott NonPlayer fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén. Az NonPlayer osztályban ez egy absztrakt függvény
- **+getNotifiedAboutSunflare():** Ez a függvény az osztály die függvényét hívja, amennyiben az asteroida-ja nem igaz értékkel tér vissza az isHollow függvényre
- **+move(destination: Asteroid):** egy függvény, aszteroidáról aszteroidára való mozgást végez.

```
If(destination is reachable from NonPlayer.asteroid) {
    destination.add(NonPlayer)
    NonPlayer.asteroid.remove(NonPlayer)
    NonPlayer.asteroid=destination
}
```

- **+move(destinationGate: StarGate):** ez a függvény, amely elvégzi azokat a mozgással kapcsolatos dolgokat ahol egy teleport kaput használunk, amiket a mozgásra képes osztályoknak kell.

```
If(destinationGate is active) {
    destinationGate.neighbour.asteroid.add(NonPlayer)
    NonPlayer.asteroid.remove(NonPlayer)
    NonPlayer.asteroid= destinationGate.neighbour.asteroid
}
```

- **+die():** ez a függvény kiveszi a NonPlayer-t az aszteroidáról, és jelzi a játéknak, hogy kivehető a létező elemekből.
- **+asteroidExploded():** ez a függvény biztosítja, hogy hogyan viselkedjen az osztály, amikor alatta egy aszteroida felrobban.
- **+onTurn():** Egy absztrakt függvény mely elvégzi az adott objektum egy körben elvégzendő feladatait.

8.1.16 Planet

8.1.16.1 Felelősség

Ez az absztrakt ősosztály felel a „bolygszerű” objektumokért a játékban, ebben a beadásban erre nem lenne szükség, de később grafikus megjelenítés esetén ezeket az objektumokat hasonlóképp kell majd megjeleníteni.

8.1.16.2 Állapotdiagramok

8.1.16.3 Ősosztályok

Printable -> Planet

8.1.16.4 Interfészek

Controllable

8.1.16.5 Attribútumok

- **-id: String:** Ez az adott planet id-je, ezt arra használjuk fel, hogy a játékos ki tudja választani a megfelelő planetet, pl: amikor egy settlerrel mozog, akkor az aszteroida id-jét megadva választhatja ki melyikre kíván rálépni.
- **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
- **-owner: Game:** Egy referencia a Game osztályhoz.

8.1.16.6 Metódusok

- **+getUIString(): String:** Ez a függvény generálja az adott planet fontos tulajdonságait tartalmazó stringet (ennek legelső eleme a planet id-je) amit a UI megkap a játékosnak kiír megfelelő parancs esetén. A Planet osztályban ez egy absztrakt függvény.
- **+onTurn():** Egy absztrakt függvény mely elvégzi az adott planet egy körben elvégzendő feladatait.

8.1.17 Printable

8.1.17.1 Felelősség

Ez az absztrakt ōsosztály felel az UI-ra kiírandó objektumok releváns tulajdonságait tartalmazó stringek előállításáért, illetve azért, hogy minden objektum (amelyiknek kell) legyen egyedi id-je.

8.1.17.2 Állapotdiagramok

-

8.1.17.3 Ősosztályok

-

8.1.17.4 Interfészek

-

8.1.17.5 Attribútumok

- **-id: String:** Ez az adott objektum id-je, ezt arra használjuk fel, hogy a játékos ki tudja választani a megfelelő objektumot, pl: amikor egy settlerrel fúr, akkor a settler id-jét megadva választhatja ki melyikkel kíván fúrni.
- **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.

8.1.17.6 Metódusok

- **+getUIString(): String:** Ez a függvény generálja az adott objektum fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az objektum id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén. A Printable osztályban ez egy absztrakt függvény.

8.1.18 ReactsToSunflare

8.1.18.1 Felelősség

Ez az interfész azok az osztályok valósítják meg, amik reagálnak a napvíharra

8.1.18.2 Ősosztályok

-

8.1.18.3 Interfészek

-

8.1.18.4 Attribútumok

8.1.18.5 Metódusok

27. **+getNotifiedAboutSunflare():** Egy absztrakt függvény amely elvégzi azokat a függvény hívásokat, amiket a napihar esetén kell az osztálynak hívnia.

8.1.19 Robot

8.1.19.1 Felelősség

Ez az osztály felel az robotértés függvényeiért.

8.1.19.2 Ősosztályok

Printable->Entity->NonPlayer->Robot

8.1.19.3 Interfészek

ReactsToSunflare, Moving, Mining

8.1.19.4 Attribútumok

28. **-id: int:** Ez az adott Robot id-je, ezt arra használjuk fel, hogy az Robot-t azonosítani tudjuk.
29. **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
30. **-asteroid:Asteroid:** Ezen az aszteroidán van a Robot.
31. **-owner:Game:** Ez az osztály tárolja le a Robot-t
32. **-isRandom: Boolean:** Egy boolean ami azt jelzi, hogy a Sun random módon, vagy determinisztikusan működik-e (kitörés esély számlálás közben), ez csak a teszteléshez kell.

8.1.19.5 Metódusok

- **+getUIString(): String:** Ez a függvény generálja az adott Robot fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén.
- **+getNotifiedAboutSunflare():** Ez a függvény az osztály die függvényét hívja, amennyiben az asteroidea-ja nem igaz értékkel tér vissza az isHollow függvényre
- **+move(destination: Asteroid):** egy függvény, aszteroidáról aszteroidára való mozgást végez.

```
If(destination is reachable from NonPlayer.asteroid) {
    destination.add(NonPlayer)
    NonPlayer.asteroid.remove(NonPlayer)
    NonPlayer.asteroid=destination
}
```

- **+move(destinationGate: StarGate):** ez a függvény, amely elvégzi azokat a mozgással kapcsolatos dolgokat ahol egy teleport kaput használunk, amiket a mozgásra képes osztályoknak kell.

```
If(destinationGate is active) {
    destinationGate.neighbour.asteroid.add(NonPlayer)
    NonPlayer.asteroid.remove(NonPlayer)
    NonPlayer.asteroid= destinationGate.neighbour.asteroid
}
```

- **+die()**: ez a függvény kiveszi a Robo-t az aszteroidáról, és jelzi a játéknak, hogy kivehető a létező elemekből.
- **+asteroidExploded()**: ez a függvény biztosítja, hogy hogyan viselkedjen az osztály, amikor alatta egy aszteroida felrobban. A robot esetében ez move-t hív a robot aszteroidjának egyi szomszédjával mint paraméter. Amennyiben az nem lehetséges Robot die függvényét hívja.
- **+onTurn()**: Egy függvény hajtja végre a Robot körön belüli viselkedését. 3 körig drill-t hívja, majd 4. körben az asteroida-jának véletlenszerűen kiválasztót szomszédjával a move függvényét meghívja. Majd ezt ismétli.
- **+drill()**: Ez a függvény meghívja a Robot asteroid-n a drilling függvényt.

8.1.20 Settler

8.1.20.1 Felelősség

Ez az osztály felel a játékos által irányítható telepesért.

8.1.20.2 Ősosztályok

Printable->Entity->Settler

8.1.20.3 Interfészek

ReactsToSunflare, Moving, Drilling, Mining

8.1.20.4 Attribútumok

33. **-id: int**: Ez az adott Settler id-je, ezt arra használjuk fel, hogy az Settler-t azonosítani tudjuk.
34. **-nextId: int**: Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
35. **-asteroid:Asteroid**: Ezen az aszteroidán van a Settler.
36. **-owner:Game**: Ez az osztály tárolja le a Robot-t
37. **-inventory:Inventory**: Ebben tárolja a Settler általa össze gyűjtött nyersanyagokat.
38. **-hadActionThisTurn:Boolean**: Ez tárolja, hogy lehet-e ezzel a telepessel lépni az adott körben.
39. **-baseRecipe**: ebben van benne, hogy mi kell egy bázis megépítéséhez.
40. **-robotRecipe**: ebben van benne, hogy mi kell egy robot megépítéséhez.
41. **-stargateRecipe**: ebben van benne, hogy mi kell egy kapú megépítéséhez.

8.1.20.5 Metódusok

- **+getUIString(): String**: Ez a függvény generálja az adott Settler fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap a játékosnak kiír megfelelő parancs esetén.

- **+getNotifiedAboutSunflare():** Ez a függvény az osztály die függvényét hívja, amennyiben az asteroida-ja nem igaz értékkel tér vissza az isHollow függvényre
- **+move(destination: Asteroid):** egy függvény, aszteroidáról aszteroidára való mozgást végezi.

```
If(destination is reachable from NonPlayer.asteroid) {
    destination.add(NonPlayer)
    NonPlayer.asteroid.remove(NonPlayer)
    NonPlayer.asteroid=destination
}
```

- **+move(destinationGate: StarGate):** ez a függvény, amely elvégzi azokat a mozgással kapcsolatos dolgokat ahol egy teleport kaput használunk, amiket a mozgásra képes osztályoknak kell.

```
If(destinationGate is active) {
    destinationGate.neighbour.asteroid.add(NonPlayer)
    NonPlayer.asteroid.remove(NonPlayer)
    NonPlayer.asteroid= destinationGate.neighbour.asteroid
}
```

- **+die():** ez a függvény kiveszi a Settler-t az aszteroidáról, és jelzi a játéknak, hogy kivehető a létező elemekből.
- **+asteroidExploded():** ez a függvény meghívja a Settler die függvényét.
- **+drill():** Ez a függvény meghívja a Settler asteroid-n a drilling függvényt.
- **+mine():** egy függvény, amely kivesz egy Material-t az aszteroidából, ha az aszteroida kérge 0.
- **+buildStarGate():** Amennyiben van kellő nyers anyag azokat kiveszi az Inventory-ból és egy kapu párthelyez az Inventory-ba
- **+buildRobot():** Amennyiben van kellő nyersanyag létrehoz egy robotot az aszteroidán, amin tartozkódik, és kiveszi a felhasznált nyersanyagokat az Inventory-ból
- **+buildBase():** Amennyiben van elegendő nyersanyag, létrehoz egy bázist az aszteroidán és a felhasznált nyersanyagokat kiveszi az Inventory-kból.
- **+dropMaterial(m:Material):** Amennyiben lehetséges (üres aszteroida átfúrt felszínnel), nyersanyagot helyez az aszteroidába az inventory-ból.
- **+placeStarGate(sg: StarGate):** Amennyiben van az Inventory-ben StarGate, lehelyezi azt az adott aszteroidára.

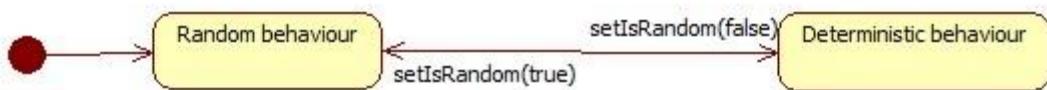
8.1.21 StarGate

8.1.21.1 Felelősség

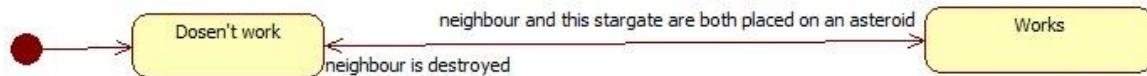
Ez az osztály felel a teleportkapukért s azok feladataiért, pl: hogy azok mindenkorral össze legyenek párosítva, illetve ha a párkból az egyik elpusztult akkor erről tudjon a másik (ha van rá szüksége).

8.1.21.2 Állapotdiagramok

- Viselkedés



- Működés



(A felső feltétel a balról jobbra menő nyilakhoz tartozik, az alsó pedig a jobbról balra menő nyilakhoz.)

8.1.21.3 Ősosztályok

Printable -> Building -> StarGate

8.1.21.4 Interfészek

reactsToSunFlare. Moving

8.1.21.5 Attribútumok

- **-id: String:** Ez az adott StarGate id-je, ezt arra használjuk fel, hogy a játékos ki tudja választani a megfelelő objektumot, pl: amikor egy settlerrel StarGate-re lép, akkor a StarGate id-jét megadva választhatja ki melyikre kíván rálépni.
- **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
- **-asteroid: Asteroid:** Az aszteroida, amin a StarGate található.
- **-neighbour: StarGate:** A StarGate-tel összepárosított StarGate.
- **-works: Boolean:** Egy bool ami azt jelzi, hogy az adott StarGate működik-e.
- **-wasInSunFlare: Boolean:** Egy bool ami azt jelzi, hogy az adott StarGate-t érte-e már napvilár.
- **-isRandom: Boolean:** Egy boolean ami azt jelzi, hogy a StarGate random módon, vagy determinisztikusan működik-e (mozgás közben), ez csak a teszteléshez kell.
- **-owner: Game:** Egy referencia a Game osztályhoz.

8.1.21.6 Metódusok

- **+genUIString(): String:** Ez a függvény generálja az adott StarGate fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén. A generált string a StarGate id-jét, isRandom értékét, works értékét, wasInSunFlare értékét, neighbour StarGate id-jét s az aszteroida id-jét tartalmazza.
- **+getNotifiedAboutSunflare():** Ez a függvény akkor hívódik meg a StarGaten amikor azt napvilár éri, ilyenkor a StarGateben true-ba billen a wasInSunFlare változó, innentől a StarGate elkezd mozogni.
- **+move(destination: Asteroid):** Ebben a függvényben a StarGate átmozog a destination aszteroidára. Ez az átlépés nem történik meg ha a destination aszteroida nem szomszédos a stargate mostani aszteroidájával. Átlépés során lekerül az éppeni aszteroidájáról, s felkerül a cél aszteroidára.

- **+move(destination: StarGate):** Ebben afüggvényben a StarGate rálép a destination StarGate-re s azon keresztül annak pájrának aszteroidájára. Ez az átlépés nem történik meg ha ez a StarGate s a destination StarGate nem azonos, illetve ha a destination StarGate működik. Átlépés során lekerül az éppeni aszteroidájáról, s felkerül a cél aszteroidára.

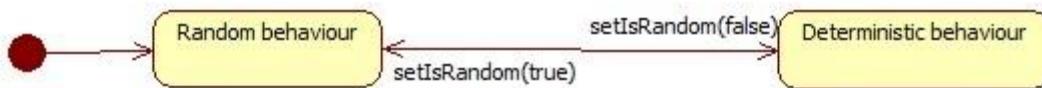
8.1.22 Sun

8.1.22.1 Felelősség

Ez az osztály felel a játékban található napokért s azoknak feladataiért, pl: azért hogy azokhoz mely aszteroidák tartoznak, mikor törnek ki az egyes napok stb.

8.1.22.2 Állapotdiagramok

- Viselkedés



(a felső feltétel a balról jobbra menő nyilakhoz tartozik, az alsó pedig a jobbról balra menő nyilakhoz)

8.1.22.3 Ősosztályok

Printable -> Planet -> Sun

8.1.22.4 Interfészek

Controllable

8.1.22.5 Attribútumok

- **-id: String:** Ez az adott sun id-je, ezt arra használjuk fel, hogy a játékos ki tudja választani a megfelelő sunt. Ez most a sunban nincsen használva, de ha a továbbiakban bővül a specifikáció akkor használhatjuk.
- **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
- **-owner: Game:** Egy referencia a Game osztályhoz.
- **-asteroids: Asteroid[0...*]:** A naphoz tartozó aszteroidák listája, ez a játék elején állítódik be. Fontos hogy ez nem azokat az aszteroidákat tartalmazza melyek közel vannak ehhez a naphoz, hanem az összes naphoz tartozó aszteroidát, legyen az most közel vagy távol a naptól.
- **-isRandom: Boolean:** Egy boolean ami azt jelzi, hogy a Sun random módon, vagy determinisztikusan működik-e (kitörés esély számlálás közben), ez csak a teszteléshez kell.

8.1.22.6 Metódusok

- **+genUIString(): String:** Ez a függvény generálja az adott Sun fontos tulajdonságait tartalmazó stringet (ennek legelső eleme a Sun id-je) amit a UI megkap s a játékosnak

kiír megfelelő parancs esetén. A string a sun id-jét, isRandom értékét, illetve az összes aszteroidának a genUIString()-je által generált stringeket tartalmazza.

- **+onTurn():** Egy absztrakt függvény mely elvégzi az adott sun egy körben elvégzendő feladatait: kisorsolja hogy ebben a körben ez a nap kitör-e s ha igen akkor a sunFlare() függvényt meghívva elvégzi a napkitörést. Az onTurn működése az isRandom változótól függ. Random esetbel a sun minden körben x esélyel török ki, nem random esetben pedig a felhasználó hívja meg a sunFlare()-t amikor szertné, így az onTurn nem csinál semmit.
- **+sunFlare():** Elvégzi a napkitörés feladatait: meghívja az összes napközelben lévő aszteroidáján a getNotifiedAboutSunflare() függvényt.

8.1.23 UI

8.1.23.1 Felelősség

Ez az osztály felel a játékossal való kommunikációval, ebbe beletartozik a játék megjelenítése is.

8.1.23.2 Ősosztályok

-

8.1.23.3 Interfészek

-

8.1.23.4 Attribútumok

- **-game: Game:** Az egyetlen game osztályra egy referencia, ez az osztály fogja végrehajtani az adott utasításokat, amiket a játékos bead a UI-nak.

8.1.23.5 Metódusok

- **+displayMessage(s: String):** Ez a függvény kiír egy stringet a játékos számára a konzolra.
- **+execute(s: String):** Ez a függvény egy beolvasott utasítást végrehajttat a game osztállyal. A függvény sok egymásba ágyazott switch case blokkból áll. Az első switch case detektálja az utasítás típusát (a nyelv alapján), az ezután következő switch casek pedig az adott utasítás további argumentumait dolgozzák fel pl: melyik id-jű objektumot kell módosítani/létrehozni stb. Miután a parancsot meghatározzuk a game beli megfelelő függvényt kell meghívunk.

8.1.24 Uran

8.1.24.1 Felelősség

Ez az osztály az Uran viselkedésér-t felel

8.1.24.2 Ősosztályok

Printable->Material

8.1.24.3 Interfészek

8.1.24.4 Attribútumok

42. **-id: int:** Ez az adott Material id-je, ezt arra használjuk fel, hogy az Material -t azonosítani tudjuk.
43. **-nextId: int:** Ez egy statikus tagváltozó mely azt tartja számon, hogy mi legyen a következő létrehozott id-t igénylő objektum id-je. Azért statikus, hogy az id-k egyediségét garantáljuk.
44. **#isRadio: boolean:** Ez a változó mondja meg, hogy radioaktív-e az adott nyersanyag.
45. **#Name:String:** Ez tárolja az adott nyersanyag nevét.
46. **#canEvaporate:boolean:** Ez a változó mondja meg, hogy elpárolog-e a nyersanyag ha napközelbe kerül.
47. **-exposedCounter:int:** Ez tárolja számon, hogy milyen hosszan volt kitéve a napnak az adott nyersanyag.

8.1.24.5 Metódusok

48. **+getUIString(): String:** Ez a függvény generálja az adott Material fontos tulajdonságait tartalmazó stringet (ennek legelső eleme az épület id-je) amit a UI megkap s a játékosnak kiír megfelelő parancs esetén.
49. **+exposedAndCloseToSun(a:Asteroid):** Ez a függvény hívodik, amikor napközelben felszínen van a nyersanyag. A paraméterül kapott Asteroid aminek a felszínén van. Növeli exposedCounter-t, ha 3 fölé emelkedik, akkor az asteroid amit paraméterül kapott explode függvényét meghívja.

8.2 A tesztek részletes tervezeti leírásuk a teszt nyelvén

A tesztek során egyes helyek #-után kommentek lettek megadva a bemeneti nyelvben valamint a kimenetben. Ezek nem értelmezi a program.

A tesztek során „init game manually” parancs a nyelvnek megfelelően minden determinisztikus állapotra állít alapértelmezésben, ennek ellenére szemléltetésképpen egyes helyeken külön beállításra kerül.

8.2.1 Bányászás telepessel

- **Leírás**

Egy telepes bányászik az aszteroidájából.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Annak ellenőrzése, hogy a telepes valóban kibányássza-e a nyersanyagot egy átfúrt aszteroidából és azt valóban eltárolja-e.

Lehetséges hiba lehet, hogy a kibányászott nyersanyag nem kerül be a telepes inventory -jába, vagy nem tűnik el az aszteroidából.

- **Bemenet**

```
init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
set asteroid capacity ast1 10
```

```

create coal coal asteroid ast1
set asteroid rockThickness ast1 0
set asteroid closeToSun ast1 f
start game
mine settler set1
list turn events
list all
    • Elvárt kimenet
# turn events:
material mined coal

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast1
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv1
materials:
MaterialId: coalstarGate: -
-----
```

8.2.2 Bányászás telepessel, teli tárolóval

- **Leírás**

Egy telepes bányászna az aszteroidájából, de nem tud, mivel nem fér több nyersanyag a tárolójába.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A telepes annak ellenére kibányássza a nyersanyagot, hogy teli van a tárolója. Az aszteroida inkonzisztens állapotban kerül, a bányászás félig megtörténik, de nem tudjuk eltárolni és nem kerül vissza a nyersanyag az aszteroidába.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
set asteroid capacity ast1 10
create coal coal asteroid ast1
create iron iro1 settler set1
create iron iro2 settler set1
create iron iro3 settler set1
create iron iro4 settler set1
create iron iro5 settler set1
create iron iro6 settler set1
create iron iro7 settler set1
create iron iro8 settler set1
create iron iro9 settler set1
create iron iro10 settler set1
set asteroid rockThickness ast1 0
start game
mine settler set1
list turn events
list all

```

Elvárt kimenet

turn events:

-

```

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    material: coal
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv1

```

```

materials:
  material: iro1
  material: iro2
  material: iro3
  material: iro4
  material: iro5
  material: iro6
  material: iro7
  material: iro8
  material: iro9
  material: iro10
starGates:
  starGate: -
-----
```

8.2.3 Telepes visszahelyez egy nyersanyagot

- **Leírás**

Egy telepes visszahelyez egy nyersanyagot az aszteroidájába.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Annak vizsgálata, hogy a telepes valóban képes-e visszahelyezni egy nyersanyagot az aszteroidájába, ami üreges. Várható hiba, hogy a visszahelyezett nyersanyag nem tűnik el a telepes tárolójából, vagy nem jelenik meg az aszteroidában.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create coal coal settler set1
set asteroid capacity ast1 10
set asteroid rockThickness ast1 0
start game
drop settler set1 material coal
list turn events
list all
```

- **Elvárt kimenet**

```
# turn events:
material droppedcoal
```

```

# list all:
SunId: sun1
isRandom: f
asteroids:
  asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
```

```

materials:
    material: coal
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: -
starGates:
    starGate: -
-----
```

8.2.4 Telepes nem tud visszahelyezni egy nyersanyagot

- **Leírás**

Egy telepes nem tud visszahelyezni nyersanyagot az aszteroidájába.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Annak vizsgálata, hogy a telepes valóban nem tud-e visszahelyezni nyersanyagot egy olyan aszteroidába, aminek felszíne nincs átfúrva, vagy már van benne valami. Várható hiba, hogy a telepes tárolójából mégis eltűnne a nyersanyag, vagy megjelenne az aszteroida belsejében.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create coal coal settler set1
set asteroid capacity ast1 10
set asteroid rockThickness ast1 1
start game
drop settler set1 material coal
list turn events
list all
```

- **Elvárt kimenet**

```
# turn events:
-
# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
```

```
-----
Asteroid id: ast1
rockThickness: 1
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: coal
starGates:
    starGate: -
-----
```

8.2.5 Telepes visszahelyez egy uránt napközelben

- **Leírás**

Egy telepes visszahelyez egy uránt az aszteroidájába, napközelben.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Annak vizsgálata, hogy miután a telepes visszahelyezte az uránt az aszteroidájába, valóban öregedett az urán, vagy ha szükséges felrobban. Ebben a tesztesetben több kört hajtunk végre, egymás után 2-szer eldobjuk, majd kibányásszuk az uránt, majd a harmadik eldobás után kell robbannia. Várható hiba, hogy a telepes tárolójából nem tűnik el a nyersanyag, nem kerül be az aszteroidába, a napközelben lévő, felfedett urán nem öregszik és ezáltal nem robban fel. Emiatt nem semmisül meg az aszteroida, vagy nem pusztul el a telepes.

- **Bemenet**

```
init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create uran ural settler set1
set asteroid capacity ast1 10
set asteroid rockThickness ast1 0
set asteroid closeToSun ast1 t
start game
drop settler set1 material ural
start next turn
mine settler set1
```

```

start next turn
drop settler set1 material ural
start next turn
list turn events
list all
-

```

8.2.6 Telepes visszahelyez egy vízjeget napközelben

- **Leírás**

Egy telepes visszahelyez egy vízjeget az aszteroidájába, napközelben.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Annak vizsgálata, hogy miután a telepes visszahelyezte a vízjeget az aszteroidájába, az valóban elpárolgott-e. Lehetséges hiba, ha a kör végén a vízjég bárhol megjelenik.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create ice icel settler set1
set asteroid capacity ast1 10
set asteroid rockThickness ast1 0
set asteroid closeToSun ast1 t
start game
drop settler set1 material icel
list turn events
list all

```

- **Elvárt kimenet**

```

# turn events:
material droppedice1
material evaporated icel

```

```

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: t
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -

```

```

settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: -
starGates:
    starGate: -
-----
```

8.2.7 Alien bányászik

- **Leírás**

Egy alien kibányászik egy nyersanyagot az aszteroidából.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Annak vizsgálata, hogy egy alien valóban tud-e bányászni az aszteroidájából. Lehetséges hiba, hogy a kibányászott nyersanyag nem tűnik el az aszteroidából.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create alien al1 asteroid ast1
create inventory inv1 alien al1
create coal coal asteroid ast1
set asteroid capacity ast1 10
set asteroid rockThickness ast1 0
start game
mine alien al1
list turn events
list all
```

- **Elvárt kimenet**

The game is lost!

```
# turn events:
asteroid mined ast1
```

```

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
```

```

materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: alil
-----
Alien id: alil
isRandom: f
asteroid: ast1
-----
```

8.2.8 Alien bányászna

- **Leírás**

Egy alien szeretne kibányászni egy nyersanyagot az aszteroidájából.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Annak vizsgálata, hogy egy alien valóban nem tud-e bányászni az aszteroidájából abban az esetben, ha nincs átfúrva az aszteroida.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create alien alil asteroid ast1
create inventory inv1 alien alil
set asteroid capacity ast1 10
create coal coal asteroid ast1
set asteroid rockThickness ast1 1
start game
mine alien alil
list turn events
list all
```

- **Elvárt kimenet**

The game is lost!

```
# turn events:
-
```

```

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 1
```

```

closeToSun: f
isRandom: f
materials:
    material: coal
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: alil
-----
Alien id: alil
isRandom: f
asteroid: ast1
-----
```

8.1.1 Nem elvégezhető Robot Crafting

- **Leírás**

A játékos a telepest irányítva egy aszteroidán állva próbál meg elkészíteni egy robotot. A robot elkészítése alapvetően akkor akadhat meg amikor úgy próbálják meg létrehozni, hogy nincsenek meg a szükséges nyersanyagok. Ezen eset tesztelését végzi a leírás.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Robot építése use-case megvalósítását teszteli.

A robot nem elkészíthető, ha nincs a telepesnél 1 darab vas (Iron), 1 darab urán (Uran) és 1 db szén (Coal).

Ezen lefutás a megfelelő nyersanyagok meglétének az ellenőrzésében összpontosul, így ott léphet fel egyedül hiba, amely a sikeres teszt meghiúsulását eredményezheti.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create iron iro1 settler set1
create coal coal settler set1
start game
build settler set1 robot
list turn events
list all
```

- **Elvárt kimenet**

```
# turn events:
-
```

```
# list all:
SunId: sun1
```

```

isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: iro1
    material: coal
starGates:
    starGate: -
-----
```

8.1.2 Elvégezhető Robot Crafting

- **Leírás**

A játékos a telep est irányítva egy aszteroidán állva próbál meg elkészíteni egy robotot. A robot elkészítését vizsgálja abban az esetben amikor a létrehozáshoz szükséges nyersanyagokkal rendelkezik a telepes.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Robot építése use-case megvalósítását teszteli.

A robot elkészíthető, ha a telepénél van 1 darab vas (Iron), 1 darab urán (Uran) és 1 db szén (Coal).

Ezen lefutás a megfelelő nyersanyagok meglétének az ellenőrzésében összpontosul, így ott léphet fel egyedül hiba, amely a sikeres teszt meghiúsulását eredményezheti.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create iron iro1 settler set1
create uran ura1 settler set1
create coal coal settler set1
```

```

start game
build settler set1 robot
start next turn
list turn events
list all
    • Elvárt kimenet
# turn events:
-
-----  

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: rob11
-----
Settler id: set1
hadActionThisTurn: false
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: -
starGates:
    starGate: -
-----
Robot id: rob11
isRandom: false
asteroid: ast1
-----
```

8.1.3 Nem elvégezhető Stargate Crafting

- **Leírás**

A játékos a telepest irányítva egy aszteroidán állva próbál meg elkészíteni egy kapupárt.

A teleportkapu elkészítése alapvetően akkor akadhat meg amikor a játékos a telepest irányítva úgy próbálja meg létrehozni, hogy nincsenek nála a szükséges nyersanyagok. Ezen eset tesztelését végzi a leírás.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Teleportkapu építése use-case megvalósítását teszeli.

A teleportkapu elkészíthető, ha a telepesnél van 2 darab vas (Iron), 1 darab urán (Uran) és 1 db vízjég (Ice).

Ezen lefutás a megfelelő nyersanyagok meglétének az ellenőrzésében összpontosul, így ott léphet fel egyedül hiba, amely a sikeres teszt meghiúsulását eredményezheti.

- **Bemenet**

```
init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create iron iro1 settler set1
create uran ura1 settler set1
create ice ice1 settler set1
start game
build settler set1 stargate
list turn events
list all
```

- **Elvárt kimenet**

```
# turn events:
```

```
-
```

```
# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
```

```

asteroid: ast1
inventory: Inventory id: inv1
materials:
  material: irol
  material: ural
  material: ice1
starGates:
  starGate: -
-----
```

8.1.4 Elvégezhető Stargate Crafting

- **Leírás**

A játékos a telepést irányítva egy aszteroidán állva próbál meg elkészíteni egy kapupárt.

A teleportkapu elkészítését vizsgálja abban az esetben amikor a létrehozáshoz szükséges nyersanyagokkal rendelkezik a játékos által irányított telepes.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Teleportkapu építése use-case megvalósítását teszteli.

A teleportkapu elkészíthető, ha a telepesnél van 2 darab vas (Iron), 1 darab urán (Uran) és 1 db vízjég (Ice).

Ezen lefutás a megfelelő nyersanyagok meglétének az ellenőrzésében összpontosul, így ott léphet fel egyedül hiba, amely a sikeres teszt meghiúsulását eredményezheti.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create iron irol settler set1
create iron irol settler set1
create uran ural settler set1
create ice ice1 settler set1
start game
build settler set1 stargate
list turn events
list all
```

- **Elvárt kimenet**

```

# turn events:
stargate crafted sta12
stargate crafted sta13
```

```

# list all:
SunId: sun1
isRandom: f
asteroids:
  asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
```

```

closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: -
starGates:
    starGate: sta12
    starGate: sta13
-----
StarGate id: sta12
isRandom: f
works: f
wasInSunFlare: f
neighbour: sta13
asteroid: -
-----
StarGate id: sta13
isRandom: f
works: f
wasInSunFlare: f
neighbour: sta12
asteroid: -
-----
```

8.1.5 Nem elvégezhető Base Crafting egy telepessel

- **Leírás**

A játékos a telep est irányítva egy aszteroidán állva próbál meg elkészíteni egy bázist. A bázis elkészítése alapvetően akkor akadhat meg amikor a játékos a telep est irányítva úgy próbálja meg létrehozni, hogy az aszteroidán lévő telepesek nem rendelkeznek a szükséges nyersanyagokkal. Ezen belül azon eset tesztelését végzi a leírás, ahol egy telep esetben próbálja meg elkészíteni csak a saját anyagaiból a bázist.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Bázis építése use-case megvalósítását teszteli.

A bázis elkészíthető, ha az aszteroidán lévő telepeseknél van 3 darab vas (Iron), 3 darab urán (Uran), 3 db vízjég (Ice) és 3 darab szén (Coal).

Ezen lefutás a megfelelő nyersanyagok meglétének az ellenőrzésében összpontosul, így ott léphetnek fel alapvetően hibák, amelyek a sikeres teszt meghiúsulását eredményezhetik. A vizsgálat itt az aszteroidán lévő összes telepes inventory-án végig futhat, így itt több hiba is felléphet.

- **Bemenet**

```
init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create iron iro1 settler set1
create coal coal1 settler set1
create uran ura1 settler set1
create ice icel1 settler set1
start game
build settler set1 base
list turn events
list all
```

- **Elvárt kimenet**

```
# turn events:
-
```

```
# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: iro1
```

```

material: coal
material: ural
material: ice1
starGates:
  starGate: -
-----

```

8.1.6 Nem elvégezhető Base Crafting telepesek külön aszteroidán

- **Leírás**

A játékos a telepest irányítva egy aszteroidán állva próbál meg elkészíteni egy bázist, úgy hogy megszerezze a szükséges nyersanyagokat, de azok két különböző aszteroidán álló telepesénél vannak.

A bázis elkészítése alapvetően akkor akadhat meg amikor a játékos a telepest irányítva úgy próbálja meg létrehozni, hogy a telepeske rendelkeznek a megfelelő nyersanyagokkal, de különböző aszteroidán vannak. Ezen eset tesztelését végzi a leírás.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Bázis építése use-case megvalósítását teszteli.

A bázis elkészíthető, ha az aszteroidán lévő telepeseknél van 3 darab vas (Iron), 3 darab urán (Uran), 3 db vízjég (Ice) és 3 darab szén (Coal).

Ezen lefutás a megfelelő nyersanyagok meglétének az ellenőrzésében összpontosul, így ott léphet fel alapvetően hiba, amely a sikeres teszt meghiúsulását eredményezheti. A vizsgálat itt az aszteroidán lévő összes telepes inventory-án végig futhat, így itt több hiba is felléphet.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create asteroid ast2 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create iron iro1 settler set1
create iron iro2 settler set1
create iron iro3 settler set1
create coal coal1 settler set1
create coal coa2 settler set1
create coal coa3 settler set1
create settler set2 asteroid ast2
create inventory inv2 settler set2
create uran ural1 settler set2
create uran ura2 settler set2
create uran ura3 settler set2
create ice ice1 settler set2
create ice ice2 settler set2
create ice ice3 settler set2
start game
build settler set1 base
list turn events
list all

```

- **Elvárt kimenet**

```
# turn events:
```

```
-  
  
# list all:  
SunId: sun1  
isRandom: f  
asteroids:  
    asteroid: ast1 asteroid: ast  
-----  
Asteroid id: ast1  
rockThickness: 0  
closeToSun: f  
isRandom: f  
materials:  
    materials: -  
neighbours:  
    neighbour: -  
buildings:  
    building: -  
settlers:  
    settler: set1  
nonPlayers:  
    nonPlayer: -  
-----  
Asteroid id: ast2  
rockThickness: 0  
closeToSun: f  
isRandom: f  
materials:  
    materials: -  
neighbours:  
    neighbour: -  
buildings:  
    building: -  
settlers:  
    settler: set2  
nonPlayers:  
    nonPlayer: -  
-----  
Settler id: set1  
hadActionThisTurn: true  
asteroid: ast1  
inventory: Inventory id: inv1  
materials:  
    material: iro1  
    material: iro2  
    material: iro3  
    material: coal  
    material: coa2  
    material: coa3  
starGates:
```

```

starGate: -
-----
Settler id: set2
hadActionThisTurn: false
asteroid: ast2
inventory: Inventory id: inv2
materials:
material: ura1
material: ura2
material: ura3
material: ice1
material: ice2
material: ice3
starGates:
starGate: -
-----
```

8.1.7 Elvégezhető Base Crafting

- **Leírás**

A játékos a telepest irányítva egy aszteroidán állva próbál meg elkészíteni egy bázist, úgy hogy az aszteroidán álló másik telepesnél és nála megvannak a megfelelő nyersanyagok. A báziselkészítését vizsgálja abban az esetben amikor a létrehozáshoz szükséges nyersanyagokkal rendelkeznek a játékos által irányított telepesek, és ugyan azon az aszteroidán álnak.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Bázis építése use-case megvalósítását teszteli.

A bázis elkészíthető, ha az aszteroidán lévő telepeseknél van 3 darab vas (Iron), 3 darab urán (Uran), 3 db vízjég (Ice) és 3 darab szén (Coal).

Ezen lefutás a megfelelő nyersanyagok meglétének az ellenőrzésében összpontosul, így ott léphet fel alapvetően hiba, amely a sikeres teszt meghiúsulását eredményezheti. A vizsgálat itt az aszteroidán lévő összes telepes inventory-án végig futhat, így itt több hiba is felléphet.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create asteroid ast2 sun sun1
create settler set1 asteroid ast1
create inventory i1 settler set1
create iron iro1 settler set1
create iron iro2 settler set1
create iron iro3 settler set1
create coal coa1 settler set1
create coal coa2 settler set1
create coal coa3 settler set1
create settler set2 asteroid ast1
create inventory inv2 settler set2
create uran ura1 settler set2
create uran ura2 settler set2
create uran ura3 settler set2
create ice icel1 settler set2
```

```

create ice ice2 settler set2
create ice ice3 settler set2
start game
build settler set1 base
list turn events
list all
    • Elvárt kimenet
The game is won!
# turn events:
base crafted bas27

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast1 asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: bas27
settlers:
    settler: set1
    settler: set2
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast2
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true

```

```

asteroid: ast1
inventory: Inventory id: i1
materials:
  material: -
starGates:
  starGate: -
-----
Settler id: set2
hadActionThisTurn: false
asteroid: ast1
inventory: Inventory id: inv2
materials:
  material: -
starGates:
  starGate: -
-----
Base id: bas27
asteroid: ast1
-----
```

8.1.8 Első Stargate lehelyezése a párból

- **Leírás**

A játékos a telepest irányítva egy aszteroidán állva próbál meg lehelyezni egy teleportkaput, úgy, hogy a párja a kapunak még az inventory-ban van.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Teleportkapu lehelyezés, pár az inventoryban use-case megvalósítását teszteli.

A teleportkapu csak akkor helyezhető le, ha az aktuális telepes rendelkezik legalább eggyel az inventory-jában.

Ezen teszteset nem rendelkezik kivételesen várható hibahellyel.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
create inventory inv1 settler set1
create stargate sta1 inventory set1
create stargate sta2 inventory set1
set stargate neighbour sta1 sta2
set stargate neighbour sta2 sta1
start game
place settler set1 sta1
list turn events
list all
```

- **Elvárt kimenet**

```
# turn events:
stargate placed sta1 ast1
```

```
# list all:
SunId: sun1
```

```

isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: stal
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: -
starGates:
    starGate: sta2
-----
StarGate id: stal
isRandom: f
works: f
wasInSunFlare: f
neighbour: sta2
asteroid: ast1
-----
StarGate id: sta2
isRandom: f
works: f
wasInSunFlare: f
neighbour: stal
asteroid: -
-----
```

8.1.9 Második Stargate lehelyezése a páról

- **Leírás**

A játékos a telepést irányítva egy aszteroidán álva próbál meg lehelyezni egy teleportkaput, úgy, hogy a párja a kapunak egy másik aszteroidán lehelyezésre került már.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Teleportkapu lehelyezés, pár az inventoryban use-case megvalósítását teszteli.

A teleportkapu csak akkor helyezhető le, ha az aktuális telepes rendelkezik legalább eggyel az inventory-jában.

Ha a kapu párja le van helyezve, akkor biztosan működik, mivel csak akkor lehet tönkre, ha a tartalmazó aszteroida felrobban, de ekkor a pár minkét tagja törlésre kerül.

A teszesetben lehetséges hibahely a kapupár működésének a vizsgálata.

- **Bemenet**

```
init game manually
create sun sun1
create asteroid ast1 sun sun1
create asteroid ast2 sun sun1
create settler set1 asteroid ast2
create inventory inv1 settler set1
create stargate sta1 asteroid ast1
create stargate sta2 inventory set1
set stargate neighbour sta1 sta2
set stargate neighbour sta2 sta1
start game
place settler set1 sta2
list turn events
list all
```

- **Elvárt kimenet**

```
# turn events:
stargate placed sta2 ast2
```

```
# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast1 asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: sta1
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast2
rockThickness: 0
closeToSun: f
isRandom: f
```

```

materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: sta2
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadAcrtionThisTurn: true
asteroid: ast2
inventory: Inventory id: inv1
materials:
    material: -
starGates:
    starGate: -
-----
StarGate id: stal
isRandom: f
works: t
wasInSunFlare: f
neighbour: sta2
asteroid: ast1
-----
StarGate id: sta2
isRandom: f
works: t
wasInSunFlare: f
neighbour: stal
asteroid: ast2
-----
```

8.2 Napvihar kitörése nem átfúrt aszteroidán

- **Leírás**

Egy kör során az egyetlen Napon napvihar tör ki.

Ennek következtében az aszteroidán, ami ehhez a Naphoz tartozik a telepes, robot, űrlény megsemmisül és a lehelyezett kapu "Napvihar érte" állapotba kerül.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Napvihar történik use-case megvalósítását teszteli.

Hibát jelenthet, ha a napkitörés után az a1 aszteroidán maradnak entitások vagy a g1 kapu nem tárolja, hogy már volt napviharban.

- **Bemenet**

```
init game manually
create sun sun1
```

```

create sun sun2
create asteroid ast1 sun sun1
set asteroid rockThickness ast1 1
create asteroid ast2 sun sun2
add asteroid neighbour ast1 ast2
add asteroid neighbour ast2 ast1
create settler set1 asteroid ast1
create robot rob1 asteroid ast1
create alien alil asteroid ast1
create stargate stal asteroid ast1
set stargate isRandom stal t
create inventory inv1 settler set1
create stargate sta2 settler set1
set stargate neighbour stal sta2
set stargate neighbour sta2 stal
start game
sunflare sun1
list turn events
list all

```

- **Elvárt kimeneti**

```

# turn events:
sunflare on sun sun1
stargate broke stal
settler died set1
robot died rob1
alien died alil

```

```

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
SunId: sun2
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 1
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast2
buildings:

```

```
building: sta1
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast2
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast1
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: false
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: -
starGates:
    starGate: sta2
-----
Robot id: rob1
isRandom: false
asteroid: ast1
-----
Alien id: ali1
isRandom: f
asteroid: ast1
-----
StarGate id: sta1
isRandom: t
works: f
wasInSunFlare: t
neighbour: sta2
asteroid: ast1
-----
StarGate id: sta2
isRandom: f
works: f
wasInSunFlare: f
neighbour: sta1
asteroid: -
```

8.3 Napvihar kitörése átfúrt aszteroidán

- **Leírás**

Egy kör során az egyetlen Napon napvihar tör ki.

Ennek ellenére az aszteroidán, ami ehhez a Naphoz tartozik a telepes, robot és űrlény nem semmisül meg, mert elbújnak az aszteroidán.

A lehelyezett kapu "Napvihar érte" állapotba kerül.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Napvihar történik use-case megvalósítását teszteli.

Hibát jelenthet, ha a napkitörés után az a1 aszteroidán maradnak entitások vagy a g1 kapu nem tárolja, hogy már volt napviharban.

- **Bemenet**

```
init game manually
create sun sun1
create sun sun2
create asteroid ast1 sun sun1
set asteroid rockThickness ast1 0
create asteroid ast2 sun sun2
add asteroid neighbour ast1 ast2
add asteroid neighbour ast2 ast1
create settler set1 asteroid ast1
create robot rob1 asteroid ast1
create alien alil asteroid ast1
create stargate stal1 asteroid ast1
set stargate isRandom stal1 t
create inventory inv1 settler set1
create stargate sta2 settler set1
set stargate neighbour stal1 sta2
set stargate neighbour sta2 stal1
start game
sunflare sun1
list turn events
list all
```

- **Elvárt kimeneti**

```
# turn events:
sunflare on sun sun1
stargate broke stal1
```

```
# list all:
SunId: sun1
isRandom: f
asteroids:
```

```
    asteroid: ast
-----
SunId: sun2
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast2
buildings:
    building: stal
settlers:
    settler: set1
nonPlayers:
    nonPlayer: rob1
    nonPlayer: ali1
-----
Asteroid id: ast2
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast1
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: false
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: -
starGates:
    starGate: sta2
-----
Robot id: rob1
isRandom: false
asteroid: ast1
-----
```

```

Alien id: alil
isRandom: f
asteroid: ast1
-----
StarGate id: stal
isRandom: t
works: f
wasInSunFlare: t
neighbour: sta2
asteroid: ast1
-----
StarGate id: sta2
isRandom: f
works: f
wasInSunFlare: f
neighbour: stal
asteroid: -
-----
```

8.4 Fúrás robottal

- **Leírás**

A robot egy kezdetben 2 kéregvastagságú aszteroidán próbál fúrni egymás után három körben.

Ez körönként csökkenti az aszteroida kéregvastagságát, amíg a harmadik fúrást már nem tudja megtenni.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Fúrás robottal use-case megvalósítását teszteli.

Hibás működést jelent, ha az első két fúrást sem tudja megtenni, vagy tud harmadszor is fúrni. (Esetleg negatívba fordul a kéregvastagság.)

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
set asteroid rockThickness ast1 2
create robot rob1 asteroid ast1
start game
drill robot rob1
list turn events
start next turn
drill robot rob1
list turn events
start next turn
drill robot rob1
list turn events
start next turn
list all
```

- **Elvárt kimeneti**

```
The game is lost!
# turn events:
asteroid drilled ast1
asteroid drilled ast1
```

```
The game is lost!
# turn events:
asteroid drilled ast1
asteroid drilled ast1
```

```
The game is lost!
# turn events:
asteroid drilled ast1
asteroid drilled ast1
```

The game is lost!

```
# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: rob1
-----
Robot id: rob1
isRandom: false
asteroid: ast1
-----
```

8.5 Fúrás robottal, elpárolog

- **Leírás**

A robot fúr egyet, egy aszteroidán. Az aszteroida kéregvastagsága egy volt a fúrás előtt. Az aszteroida a teszeset végéig folyamatosan napközelben áll. A magjában egy vízjég található. Ez a kéreg átfúrásakor elpárolog.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ez a teszt a Fúrás robottal, elpárolog use-case megvalósítását teszteli.

Főbb hiba lehetőségek, ha a robot nem tudja átfürni a kérget, a jég pedig nem párolog el.

- **Bemenet**

```
init game manually
create sun sun1
create asteroid ast1 sun sun1
create robot rob1 asteroid ast1
set asteroid capacity ast1 10
create ice ice1 asteroid ast1
set material exposedCounter ice1 0
set asteroid rockThickness ast1 1
set asteroid closeToSun ast1 t
start game
drill robot rob1
list turn events
list all
```

- **Elvárt kimenet**

```
The game is lost!
# turn events:
asteroid drilled ast1
material evaporated ice1
asteroid drilled ast1
```

```
# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: t
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: rob1
-----
Robot id: rob1
```

```
isRandom: false
asteroid: ast1
-----
```

8.6 Fúrás robottal, felrobban

- **Leírás**

A robot fúr egyet, egy aszteroidán. Az aszteroida kéregvastagsága egy volt a fúrás előtt. Az aszteroida a teszeset végéig folyamatosan napközelben áll. A magjában egy radioaktív urán található.

Ez három kör felfedettség után felrobban, elpusztítva az aszteroidát és a robotot szomszédos aszteroidára repítve.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ez a teszt a Fúrás robottal, felrobban use-case megvalósítását teszteli.

Főbb hiba lehetőségek, ha a robot nem tudja átfúrni a kérget, az urán pedig nem robban fel 3 kör múlva, vagy esetleg korábban robban fel.

Hiba, ha az érintett aszteroida nem robban fel, vagy a robot nem repül a szomszédos aszteroidára.

- **Bemenet**

```
init game manually
create sun sun1
create asteroid ast1 sun sun1
create robot rob1 asteroid ast1
set asteroid capacity ast1 10
create uran ural asteroid ast1
set material exposedCounter ural 0
set asteroid rockThickness ast1 1
set asteroid closeToSun ast1 t
create asteroid ast2 sun sun1
add asteroid neighbour ast1 ast2
add asteroid neighbour ast2 ast1
start game
drill robot rob1
list turn events
list all
```

- **Elvárt kimenet**

```
The game is lost!
# turn events:
asteroid drilled ast1
asteroid drilled ast1
```

```
# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast1 asteroid: ast
```

```
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: t
isRandom: f
materials:
    material: ural
neighbours:
    neighbour: ast2
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: rob1
-----
Asteroid id: ast2
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast1
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Robot id: rob1
isRandom: false
asteroid: ast1
-----
```

8.7 Fúrás robottal, felrobban, nincs szomszéd

- **Leírás**

A robot fúr egyet, egy aszteroidán. Az aszteroida kéregvastagsága egy volt a fúrás előtt. Az aszteroida a teszeset végéig folyamatosan napközelben áll. A magjában egy radioaktív urán található.

Ez három kör felfedettség után felrobban, elpusztítva az aszteroidát. Mivel nincs szomszédos aszteroida, ezért a robot is megsemmisül.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ez a teszt a Fúrás robottal, felrobban, nincs szomszéd use-case megvalósítását teszteli. Főbb hiba lehetőségek, ha a robot nem tudja átfürni a kérget, az urán pedig nem robban fel 3 kör műlva, vagy esetleg korábban robban fel.

Hiba, ha az érintett aszteroida és robot nem robban fel.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create robot rob1 asteroid ast1
set asteroid capacity ast1 10
create uran ural asteroid ast1
set material exposedCounter ural 0
set asteroid rockThickness ast1 1
set asteroid closeToSun ast1 t
create asteroid ast2 sun sun1
start game
drill robot rob1
list turn events
start next turn
start next turn
list turn evnets
start next turn
list all

```

- **Elvárt kimenet**

```

The game is lost!
# turn events:
asteroid drilled ast1
asteroid drilled ast1

```

```

The game is lost!
The game is lost!
# turn events:
asteroid exploded ast1
robot died rob1
material exploded ural

```

```

The game is lost!
# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast2
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:

```

```

building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----

```

8.8 Fúrás telepessel

- **Leírás**

A telepes egy kezdetben 2 kéregvastagságú aszteroidán próbál fúrni egymás után három körben. Ez körönként csökkenti az aszteroida kéregvastagságát, amíg a harmadik fúrást már nem tudja megtenni.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Fúrás telepessel use-case megvalósítását teszteli.

Hibás működést jelent, ha az első két fúrást sem tudja megtenni, vagy tud harmadszor is fúrni. (Esetleg negatívba fordul a kéregvastagság.)

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
set asteroid rockThickness ast1 2
set settler hadActionThisTurn set1 f
start game
drill settler set1
list turn events
start next turn
drill settler set1
list turn events
start next turn
drill settler set1
list turn events
start next turn
list all

```

- **Elvárt kimenet**

```

# turn events:
asteroid drilled ast1

# turn events:
asteroid drilled ast1

# turn events:
asteroid drilled ast1

```

```

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: false
asteroid: ast1
inventory: Inventory id: inv2
materials:
    material: -
starGates:
    starGate: -
-----
```

8.9 Fúrás telepessel, elpárolog

- **Leírás**

A telepes fúr egyet, egy aszteroidán. Az aszteroida kéregvastagsága egy volt a fúrás előtt. Az aszteroida a teszeset végéig folyamatosan napközelben áll. A magjában egy vízjég található. Ez a kéreg átfúrásakor elpárolog.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ez a teszt a Fúrás telepessel, elpárolog use-case megvalósítását teszteli.
Főbb hiba lehetőségek, ha a telepes nem tudja átfúrni a kérget, a jég pedig nem párolog el.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
set settler hadActionThisTurn set1 f
set asteroid capacity ast1 10
```

```

create ice ice1 asteroid ast1
set material exposedCounter ice1 0
set asteroid rockThickness ast1 1
set asteroid closeToSun ast1 t
start game
drill settler set1
list turn events
list all

```

- **Elvárt kimenet**

```

# turn events:
asteroid drilled ast1
material evaporated ice1

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: t
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv2
materials:
    material: -
starGates:
    starGate: -
-----
```

8.10 Fúrás telepessel, felrobban

- **Leírás**

A telepes fúr egyet, egy aszteroidán. Az aszteroida kéregvastagsága egy volt a fúrás előtt.

Az aszteroida a teszeset végéig folyamatosan napközelben áll. A magjában egy radioaktív urán található.

Ez három kör felfedettség után felrobban, elpusztítva az aszteroidát és a rajta lévő telepest.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ez a teszt a Fúrás telepessel, felrobban use-case megvalósítását teszteli.

Főbb hiba lehetőségek, ha a telepes nem tudja átfürni a kérget, az urán pedig nem robban fel 3 kör műlva, vagy esetleg korábban robban fel.

Hiba, ha az érintett aszteroida és telepes nem robban fel.

- **Bemenet**

```
init game manually
create sun sun1
create asteroid ast1 sun sun1
create settler set1 asteroid ast1
set settler hadActionThisTurn set1 f
set asteroid capacity ast1 10
create uran ural asteroid ast1
set material exposedCounter ural 0
set asteroid rockThickness ast1 1
set asteroid closeToSun ast1 t
start game
drill settler set1
list turn events
start next turn
start next turn
list turn events
start next turn
list all
```

- **Elvárt kimenet**

```
# turn events:
asteroid drilled ast1
```

```
# turn events:
asteroid exploded ast1
settler died set1
material exploded ural
```

The game is lost!

```
# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid:
-----
-----
```

8.11 Mozgás robottal, kudarc

- **Leírás**

Egy robot az egyik aszteroidáról egy másik aszteroidára próbál mozogni, de a másik aszteroida nem szomszédos a jelenlegivel.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ez a teszt a Mozgás robottal use-case megvalósítását ellenőrzi.
Hiba, ha a robot át tud lépni a másik aszteroidára.

- **Bemenet**

```
init game manually
create sun sun1
create asteroid ast1 sun sun1
create asteroid ast2 sun sun1
create robot rob1 asteroid ast1
start game
move robot rob1 dest asteroid ast2
list turn events
list all
```

- **Elvárt kimenet**

```
The game is lost!
# turn events:
robot move rob1 ast2

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast1 asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: rob1
-----
Asteroid id: ast2
```

```

rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Robot id: rob1
isRandom: false
asteroid: ast1
-----
```

8.12 Mozgás robottal, siker

- **Leírás**

Egy robot az egyik aszteroidáról egy szomszédos aszteroidára mozog.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ez a teszt a Mozgás robottal use-case megvalósítását ellenőrzi.
Hiba, ha a robot nem tud szomszédos aszteroidára lépni.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create asteroid ast2 sun sun1
create robot rob1 asteroid ast1
add asteroid neighbour ast1 ast2
add asteroid neighbour ast2 ast1
start game
move robot rob1 dest asteroid ast2
list turn events
list all
```

- **Elvárt kimenet**

```

The game is lost!
# turn events:
robot move rob1 ast2

# list all:
SunId: sun1
```

```

isRandom: f
asteroids:
    asteroid: ast1 asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast2
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast2
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast1
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: rob1
-----
Robot id: rob1
isRandom: false
asteroid: ast2
-----
```

8.13 Mozgás telepessel, kudarc

- **Leírás**

Egy telepes az egyik aszteroidáról egy másik aszteroidára próbál mozogni, de a másik aszteroida nem szomszédos a jelenlegivel.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ez a teszt a Mozgás telepessel use-case megvalósítását ellenőrzi.
Hiba, ha a telepes át tud lépni a másik aszteroidára.

- **Bemenet**

```
init game manually
create sun sun1
create asteroid ast1 sun sun1
create asteroid ast2 sun sun1
create settler set1 asteroid ast1
set settler hadActionThisTurn set1 f
start game
move settler set1 dest asteroid ast2
list turn events
list all
```

- **Elvárt kimenet**

```
# turn events:
-
# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast1 asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast2
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
```

```

nonPlayer: -
-----
Settler id: set1
hadAcrtionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv3
materials:
material: -
starGates:
starGate: -
-----
```

8.14 Mozgás telepessel, siker

- **Leírás**

Egy telepes az egyik aszteroidáról egy szomszédos aszteroidára mozog.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ez a teszt a Mozgás telepessel use-case megvalósítását ellenőrzi.

Hiba, ha a telepes nem tud szomszédos aszteroidára lépni.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
create asteroid ast2 sun sun1
create settler set1 asteroid ast1
set settler hadActionThisTurn set1 f
add asteroid neighbour ast1 ast2
add asteroid neighbour ast2 ast1
start game
move settler set1 dest asteroid ast2
list turn events
list all
```

- **Elvárt kimenet**

```

# turn events:
settler move set1 ast2

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast1 asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
```

```

closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast2
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast2
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast1
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast2
inventory: Inventory id: inv3
materials:
    material: -
starGates:
    starGate: -
-----
```

8.15 Telepes átlép egy kapun.

- **Leírás**

Egy telepes átlép egy kapun, az öt azzal összekötő aszteroidára.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Annak vizsgálata, hogy egy telepes valóban át tud-e lépni egy teleportkapun a kapu másik felén lévő aszteroidára. Lehetséges hiba, hogy a telepes nem kerül át a teleportkapu párjának aszteroidájára, illetve a kezdeti aszteroidáról nem kerül le.

- **Bemenet**

```

init game manually
create sun sun1
create asteroid ast1 sun sun1
```

```

set asteroid rockThickness ast1 1
create asteroid ast2 sun sun1
set asteroid rockThickness ast2 1
create settler set1 asteroid ast1
create stargate sta1 asteroid ast1
set stargate works sta1 t
create stargate sta2 asteroid ast2
set stargate works sta2 t
set stargate neighbour sta1 sta2
set stargate neighbour sta2 sta1
start game
move settler set1 dest stargate sta1
list turn events
list all

```

- **Elvárt kimenet**

```

# turn events:
settler move set1 ast1

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast1 asteroid: ast
-----
Asteroid id: ast1
rockThickness: 1
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: sta1
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast2
rockThickness: 1
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: sta2
settlers:

```

```

settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast2
inventory: Inventory id: inv3
materials:
    material: -
starGates:
    starGate: -
-----
StarGate id: stal
isRandom: f
works: t
wasInSunFlare: f
neighbour: sta2
asteroid: ast1
-----
StarGate id: sta2
isRandom: f
works: t
wasInSunFlare: f
neighbour: sta1
asteroid: ast2
-----
```

8.16 Teleportkapu elromlik.

- **Leírás**

Egy létező és működő teleportkapu pár egyik tagját napvihar éri. Ennek hatására elromlik és elmozdul egy másik szomszédos aszteroidára.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Annak vizsgálata, hogy a teleportkapu valóban elromlik-e. Lehetséges hibahelyek: A pár másik fele valóban lekapcsolódik-e. Az elromlott teleportkapu elmozdulása során valóban lekerül-e jelenlegi aszteroidájáról, illetve felkerül-e a cél aszteroidára.

- **Bemenet**

```

init game manually
create sun sun1
create sun sun2
create asteroid ast1 sun sun1
set asteroid rockThickness ast1 1
create asteroid ast2 sun sun2
set asteroid rockThickness ast2 1
create asteroid ast3 sun sun2
set asteroid rockThickness ast3 1
add asteroid neighbour ast1 ast3
add asteroid neighbour ast3 ast1
create stargate stal asteroid ast1
```

```

set stargate isRandom stal t
create stargate sta2 asteroid ast2
set stargate isRandom sta2 t
set stargate neighbour stal sta2
set stargate neighbour sta2 stal
start game
sunflare sun1
list turn events
start next turn
list turn events
list all

```

- **Elvárt kimenet**

```

The game is lost!
# turn events:
sunflare on sun sun1
stargate broke stal

The game is lost!
# turn events:
stargate move stal ast3

```

```

# list all:
SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
SunId: sun2
isRandom: f
asteroids:
    asteroid: ast2 asteroid: ast
-----
Asteroid id: ast1
rockThickness: 1
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast3
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast2
rockThickness: 1
closeToSun: f

```

```

isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: sta2
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast3
rockThickness: 1
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast1
buildings:
    building: stal
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
StarGate id: sta2
isRandom: t
works: f
wasInSunFlare: f
neighbour: stal
asteroid: ast2
-----
StarGate id: stal
isRandom: t
works: f
wasInSunFlare: t
neighbour: sta2
asteroid: ast3
-----
```

8.3 A tesztelést támogató programok tervezése

Lehetőség van szöveges fájlból beolvasott teszteket végrehajtani. (load game) Ezek a fájlok a korábban megadott teszteket tartalmazzák majd. Ebben az esetben a kimenetet a program szintén fájlba menti. Ezen kívül parancssorban is megadhatjuk az egyes inicializáló és végrehajtási parancsokat. (init game manually) Ha a program egy parancsra választ ad, akkor azt a parancs után már kiírja a konzolba. A válaszok a fájlba írt és a konzolba írt megoldásnál megegyeznek. Lehetőség van randomizált inicializálásra, ebben az esetben a játék kezdő állapota teljes mértékben önállóan jön létre. (init game random)

8.4 Napló

Kezdet	Időtartam	Részttvevők	Leírás
2010.03.21. 18:00	2,5 óra	Horváth Németh Tóth Oláh	Értekezlet. Döntés: Horváth elkészíti az osztálydiagramot, Oláh a use-case leírásokat.
2010.03.23. 23:00	5 óra	Németh	Tevékenység: Németh implementálja a tesztelő programokat.
2021.04.04. 14:00	2 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet. Csapat az adott információk alapján szétosztott a amunkákat. Kárpáti és Mohácsi vállalta az osztályleírásokat, míg a teszteket Bárkányi, Simon és Tóth.
2021.04.05. 14:00	2 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet: Elkezdett feladatokban felvetett hibákat megbeszéltek. A teszesetek kapcsán egységesség szellemében megbeszéltük, hogy milyen formázásokat alkalmazunk.
2021.04.7. 16:00	1 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet: Felvetett hibák javítás megbeszélése. Megvitattunk tovább haladás tervezetét, valamint átbeszélünk állapotdiagrammokat.
2021.04.9. 19:00	2 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet. A csapat átbeszélte az osztályleírásokat és ütemeztük a maradék munka elkészítését.
2021.04.10. 19:00	1 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet. Elkezdtük a dokumentumot egységesen összeállítani. Behelyezésre kerültek átbeszélve a teszesetek.
2021.04.12. 10:00	4 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet. Végelegesre formáztuk közösen a dokumentumot.

10. Prototípus leadása

67 – brainstormers

Konzulens:
Simon Balázs

Csapattagok

Mohácsi Márton

Simon Zoltán

Kárpáti Márk András

Bárkányi Csaba

Tóth Ádám

T8ZT88

HRSNUG

O1BG0Z

BQI7QQ

QEYOW2

moha.mohacsi@gmail.com

simonzolika1999@gmail.com

mark.karpati1999@gmail.com

csaba.barkanyi.csaba@gmail.com

toth99adam@gmail.com

2021-04-19

Tartalom

Változtatások:	Hiba! A könyvjelző nem létezik.
Modell változtatásai:	Hiba! A könyvjelző nem létezik.
Nyelv változtatásai:	Hiba! A könyvjelző nem létezik.
10. Prototípus beadása.....	199
10.1 Fordítási és futtatási útmutató.....	199
10.1.1 Fájllista.....	199
10.1.2 Fordítás	201
10.1.3 Futtatás.....	201
10.2 Tesztek jegyzőkönyvei.....	201
10.2.1 Bányászás telepessel	201
10.2.2 Bányászás telepessel teli tárolóval	202
10.2.3 Telepes visszahelyez egy nyersanyagot.....	203
10.2.4 Telepes nem tud visszahelyezni egy nyersanyagot.....	205
10.2.5 Telepes visszahelyez egy uránt napközelben	206
10.2.6 Telepes visszahelyez egy vízjeget napközelben	206
10.2.7 Alien bányászik	207
10.2.8 Alien bányászna.....	208
10.2.9 Nem elvégezhető a robot crafting	209
10.2.10 Elvégezhető a robot crafting.....	210
10.2.11 Nem elvégezhető a StarGate crafting.....	212
10.2.12 Elvégezhető a StarGate Crafting.....	213
10.2.13 Nem elvégezhető a Base Crafting egy telepessel.....	214
10.2.14 Nem elvégezhető a Base Crafting, telepesek külön aszteroidán.....	215
10.2.15 Elvégezhető a Base Crafting.....	217
10.2.16 Első StarGate lehelyezése a párból	219
10.2.17 Második StarGate lehelyezése a párból.....	220
10.2.18 Napvihar kitörés nem átfűrt aszteroidán.....	222
10.2.19 Napvihar kitörés átfűrt aszteroidán	224
10.2.20 Fúrás robottal	226
10.2.21 Fúrás robottal elpárolog	228
10.2.22 Fúrás robottal, felrobban	229
10.2.23 Fúrás robottal, felrobban, nincs szomszéd.....	230
10.2.24 Fúrás telepessel.....	231
10.2.25 Fúrás telepessel, elpárolog	233
10.2.26 Fúrás telepessel, felrobban	234
10.2.27 Mozgás robottal, kudarc.....	235
10.2.28 Mozgás robottal, siker.....	236
10.2.29 Mozgás telepessel, kudarc.....	237
10.2.30 Mozgás telepessel, siker.....	238
10.2.31 Telepes átlép egy kapun	240
10.2.32 Teleportkapu elromlik.....	241
10.3 Értékelés.....	244
10.4 Napló	245

Prototípus beadása

5.4 Fordítási és futtatási útmutató

5.4.1 Fájllista

	Méret	Keletkezés ideje	Tartalom
Alien.java	3235 byte	2021/16/4	Az Alien osztályt tartalmazza
Asteroid.java	16,122 byte	2021/12/4	Az Asteroid osztályt tartalmazza
Base.java	1,264 byte	2021/12/4	A Base osztályt tartalmazza
Building.java	1,330 byte	2021/12/4	A Building osztályt tartalmazza
Coal.java	609 byte	2021/12/4	A Coal osztályt tartalmazza
Controllable.java	369 byte	2021/12/4	A Controllable interfacet tartalmazza
Drilling.java	81 byte	2021/16/4	A Drilling interfacet tartalmazza
Entity.java	2,246 byte	2021/12/4	A Entity osztályt tartalmazza
Game.java	40,059 byte	2021/12/4	A Game osztályt tartalmazza
Ice.java	996 byte	2021/12/4	Az Ice osztályt tartalmazza
Inventory.java	9,147 byte	2021/12/4	Az Inventory osztályt tartalmazza
Iron.java	609 byte	2021/12/4	Az Iron osztályt tartalmazza
Main.java	1,885 byte	2021/12/4	A Main osztályt tartalmazza
Material.java	3,642 byte	2021/12/4	A Material osztályt tartalmazza
Minig.java	78 byte	2021/16/4	A Mining interfacet tartalmazza
Moving.java	134 byte	2021/16/4	A Moving interfacet tartalmazza
NonPlayer.java	1,889 byte	2021/16/4	A NonPlayer osztályt tartalmazza
Planet.java	693 byte	2021/12/4	A Planet osztályt tartalmazza
Printable.java	1,144 byte	2021/14/4	A Printable osztályt tartalmazza
ReactsToSunFlare.java	102 byte	2021/16/4	A ReactsToSunFlare interfacet tartalmazza
Robot.java	3,111 byte	2021/12/4	A Robot osztályt tartalmazza
Settler.java	11,748 byte	2021/12/4	A Settler osztályt tartalmazza
StarGate.java	5,018 byte	2021/12/4	A StarGate osztályt tartalmazza
Sun.java	4,324 byte	2021/12/4	A Sun osztályt tartalmazza
Tools.java	210 byte	2021/16/4	A Tools osztályt tartalmazza
UI.java	15,740 byte	2021/14/4	A UI osztályt tartalmazza
Uran.java	1,062 byte	2021/12/4	Az Uran osztályt tartalmazza

ProtoBuildScript.bat	87 byte	2021/19/04	
ProtoRunScript.bat	19 byte	2021/19/04	

5.4.2 Fordítás

A fordításhoz szükséges Java Development Kit (JDK).

Csomagoljuk ki a Brainstormers_proto.zip-et.

Futtassul a ProtoBuildScript.bat fájlt.

Ez létrehoz egy bin könyvtárat a root könyvtárban és a src mappa tartalmát lefordítva a bin/-be írja.

(Ehhez a binben létrejön egy main alkönyvtár is.)

Végül a lefordított classokat egy Proto.jar-ba csomagolja. A Main osztályt határozza meg mint főosztály.

5.4.3 Futtatás

A futtastáshoz a ProtoRunScript.bat fájlt futtassuk!

Ez a korábban létrehozott Proto.jar csomagot futtatja.

A megnyílt parancssorban használhatjuk az elindult programot.

5.5 Tesztek jegyzőkönyvei

A tesztek bemenete esetén a megadott szintaktika végén lévő enter másolása és beillesztése is szükséges.

A kimenetek teljes megjelenítéséhez esetekben enter ütése is szükséges a bemeneten. Ha minden bemenet megjelent, akkor a „Could not parse command.” üzenet olvasható akimeneten.

5.5.1 Bányászás telepessel

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 set asteroid capacity ast1 10 create coal coa1 asteroid ast1 set asteroid rockThickness ast1 0 set asteroid closeToSun ast1 f start game mine settler set1 list turn events list all</pre>
Teszt kimenete	<pre># turn events: material mined coa1 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast1 ----- Asteroid id: ast1</pre>

	<pre> rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: true asteroid: ast1 inventory: Inventory id: inv1 materials: MaterialId: coalstarGate: - -----</pre>
--	---

5.5.2 Bányászás telepessel teli tárolóval

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 set asteroid capacity ast1 10 create coal coal1 asteroid ast1 create iron iro1 settler set1 create iron iro2 settler set1 create iron iro3 settler set1 create iron iro4 settler set1 create iron iro5 settler set1 create iron iro6 settler set1 create iron iro7 settler set1 create iron iro8 settler set1 create iron iro9 settler set1 create iron iro10 settler set1 set asteroid rockThickness ast1 0 start game mine settler set1 list turn events list all</pre>

Teszt kimenete	<pre> # turn events: - # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: material: coa1 neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: true asteroid: ast1 inventory: Inventory id: inv1 materials: material: iro1 material: iro2 material: iro3 material: iro4 material: iro5 material: iro6 material: iro7 material: iro8 material: iro9 material: iro10 starGates: starGate: - -----</pre>
-----------------------	---

5.5.3 Telepes visszahelyez egy nyersanyagot

Teszteleő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	init game manually

	<pre> create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 create coal coa1 settler set1 set asteroid capacity ast1 10 set asteroid rockThickness ast1 0 start game drop settler set1 material coa1 list turn events list all </pre>
Teszt kimenete	<pre> # turn events: material droppedcoa1 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: material: coa1 neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadActionThisTurn: true asteroid: ast1 inventory: Inventory id: inv1 materials: material: - starGates: starGate: - -----</pre>

5.5.4 Telepes nem tud visszahelyezni egy nyersanyagot

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 create coal coa1 settler set1 set asteroid capacity ast1 10 set asteroid rockThickness ast1 1 start game drop settler set1 material coa1 list turn events list all</pre>
Teszt kimenete	<pre># turn events: - # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 1 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadActionThisTurn: true asteroid: ast1 inventory: Inventory id: inv1 materials: material: coa1 starGates: starGate: -</pre>

5.5.5 Telepes visszahelyez egy uránt napközelben

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 create uran ura1 settler set1 set asteroid capacity ast1 10 set asteroid rockThickness ast1 0 set asteroid closeToSun ast1 t start game drop settler set1 material ura1 start next turn mine settler set1 start next turn drop settler set1 material ura1 start next turn list turn events list all</pre>
Teszt kimenete	<pre>The game is lost! # turn events: - # list all: SunId: sun1 isRandom: f asteroids: asteroid: -----</pre>

5.5.6 Telepes visszahelyez egy vízjeget napközelben

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 create ice ice1 settler set1 set asteroid capacity ast1 10 set asteroid rockThickness ast1 0 set asteroid closeToSun ast1 t</pre>

	<pre> start game drop settler set1 material ice1 list turn events list all </pre>
Teszt kimenete	<pre> # turn events: material droppedice1 material evaporated ice1 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: t isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: true asteroid: ast1 inventory: Inventory id: inv1 materials: material: - starGates: starGate: - -----</pre>

5.5.7 Alien bányászik

Teszteleő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create alien ali1 asteroid ast1 </pre>

	<pre> create inventory inv1 alien ali1 create coal coal1 asteroid ast1 set asteroid capacity ast1 10 set asteroid rockThickness ast1 0 start game mine alien ali1 list turn events list all </pre>
Teszt kimenete	<pre> The game is lost! # turn events: asteroid mined ast1 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: - nonPlayers: nonPlayer: ali1 ----- Alien id: ali1 isRandom: f asteroid: ast1 -----</pre>

5.5.8 Alien bányászna

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create alien ali1 asteroid ast1 create inventory inv1 alien ali1 set asteroid capacity ast1 10 </pre>

	<pre>create coal coa1 asteroid ast1 set asteroid rockThickness ast1 1 start game mine alien ali1 list turn events list all</pre>
Teszt kimenete	<pre>The game is lost! # turn events: - # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 1 closeToSun: f isRandom: f materials: material: coa1 neighbours: neighbour: - buildings: building: - settlers: settler: - nonPlayers: nonPlayer: ali1 ----- Alien id: ali1 isRandom: f asteroid: ast1 -----</pre>

5.5.9 Nem elvégezhető a robot crafting

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18.20:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 create iron iro1 settler set1 create coal coa1 settler set1</pre>

	start game build settler set1 robot list turn events list all
Teszt kimenete	<pre># turn events: - # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: true asteroid: ast1 inventory: Inventory id: inv1 materials: material: iro1 material: coal starGates: starGate: - -----</pre>

5.5.10 Elvégezhető a robot crafting

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1

	<pre> create inventory inv1 settler set1 create iron iro1 settler set1 create uran ura1 settler set1 create coal coal1 settler set1 start game build settler set1 robot start next turn list turn events list all </pre>
Teszt kimenete	<pre> # turn events: - # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: rob11 ----- Settler id: set1 hadActionThisTurn: false asteroid: ast1 inventory: Inventory id: inv1 materials: material: - starGates: starGate: - ----- Robot id: rob11 isRandom: false asteroid: ast1 -----</pre>

5.5.11 Nem elvégezhető a StarGate crafting

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 create iron iro1 settler set1 create uran ura1 settler set1 create ice ice1 settler set1 start game build settler set1 stargate list turn events list all</pre>
Teszt kimenete	<pre># turn events: - # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: true asteroid: ast1 inventory: Inventory id: inv1 materials: material: iro1 material: ura1 material: ice1</pre>

	<pre>starGates: starGate: - -----</pre>
--	---

5.5.12 Elvégezhető a StarGate Crafting

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 create iron iro1 settler set1 create iron iro1 settler set1 create uran ura1 settler set1 create ice ice1 settler set1 start game build settler set1 stargate list turn events list all</pre>
Teszt kimenete	<pre># turn events: stargate crafted sta12 stargate crafted sta13 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1</pre>

	<pre> hadActionThisTurn: true asteroid: ast1 inventory: Inventory id: inv1 materials: material: - starGates: starGate: sta12 starGate: sta13 ----- StarGate id: sta12 isRandom: f works: f wasInSunFlare: f neighbour: sta13 asteroid: - ----- StarGate id: sta13 isRandom: f works: f wasInSunFlare: f neighbour: sta12 asteroid: - -----</pre>
--	--

5.5.13 Nem elvégezhető a Base Crafting egy telepessel

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 create iron iro1 settler set1 create coal coal1 settler set1 create uran ura1 settler set1 create ice ice1 settler set1 start game build settler set1 base list turn events list all</pre>
Teszt kimenete	<pre> # turn events: - # list all: SunId: sun1 isRandom: f</pre>

	<pre> asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: true asteroid: ast1 inventory: Inventory id: inv1 materials: material: iro1 material: coal material: ura1 material: ice1 starGates: starGate: - -----</pre>
--	---

5.5.14 Nem elvégezhető a Base Crafting, telepesek külön aszteroidán.

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create asteroid ast2 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 create iron iro1 settler set1 create iron iro2 settler set1 create iron iro3 settler set1 create coal coal1 settler set1 create coal coa2 settler set1 create coal coa3 settler set1 create settler set2 asteroid ast2</pre>

	<pre> create inventory inv2 settler set2 create uran ura1 settler set2 create uran ura2 settler set2 create uran ura3 settler set2 create ice ice1 settler set2 create ice ice2 settler set2 create ice ice3 settler set2 start game build settler set1 base list turn events list all </pre>
Teszt kimenete	<pre> # turn events: - # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast1 asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Asteroid id: ast2 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set2 </pre>

	<pre> nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: true asteroid: ast1 inventory: Inventory id: inv1 materials: material: iro1 material: iro2 material: iro3 material: coa1 material: coa2 material: coa3 starGates: starGate: - ----- Settler id: set2 hadAcrtionThisTurn: false asteroid: ast2 inventory: Inventory id: inv2 materials: material: ura1 material: ura2 material: ura3 material: ice1 material: ice2 material: ice3 starGates: starGate: - -----</pre>
--	---

5.5.15 Elvégezhető a Base Crafting

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create asteroid ast2 sun sun1 create settler set1 asteroid ast1 create inventory i1 settler set1 create iron iro1 settler set1 create iron iro2 settler set1 create iron iro3 settler set1 create coal coa1 settler set1 create coal coa2 settler set1 create coal coa3 settler set1 create settler set2 asteroid ast1</pre>

	<pre> create inventory inv2 settler set2 create uran ura1 settler set2 create uran ura2 settler set2 create uran ura3 settler set2 create ice ice1 settler set2 create ice ice2 settler set2 create ice ice3 settler set2 start game build settler set1 base list turn events list all </pre>
Teszt kimenete	<pre> The game is won! # turn events: base crafted bas27 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast1 asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: bas27 settlers: settler: set1 settler: set2 nonPlayers: nonPlayer: - ----- Asteroid id: ast2 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - </pre>

	<pre> settlers: settler: - nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: true asteroid: ast1 inventory: Inventory id: i1 materials: material: - starGates: starGate: - ----- Settler id: set2 hadAcrtionThisTurn: false asteroid: ast1 inventory: Inventory id: inv2 materials: material: - starGates: starGate: - ----- Base id: bas27 asteroid: ast1 -----</pre>
--	--

5.5.16 Első StarGate lehelyezése a párból

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18.20:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 create inventory inv1 settler set1 create stargate sta1 inventory set1 create stargate sta2 inventory set1 set stargate neighbour sta1 sta2 set stargate neighbour sta2 sta1 start game place settler set1 sta1 list turn events list all</pre>
Teszt kimenete	<pre> # turn events: stargate placed sta1 ast1 # list all:</pre>

```

SunId: sun1
isRandom: f
asteroids:
    asteroid: ast
-----
Asteroid id: ast1
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: sta1
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadAcrtionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: -
starGates:
    starGate: sta2
-----
StarGate id: sta1
isRandom: f
works: f
wasInSunFlare: f
neighbour: sta2
asteroid: ast1
-----
StarGate id: sta2
isRandom: f
works: f
wasInSunFlare: f
neighbour: sta1
asteroid: -
-----
```

5.5.17 Második StarGate lehelyezése a párból

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	init game manually

	<pre> create sun sun1 create asteroid ast1 sun sun1 create asteroid ast2 sun sun1 create settler set1 asteroid ast2 create inventory inv1 settler set1 create stargate sta1 asteroid ast1 create stargate sta2 inventory set1 set stargate neighbour sta1 sta2 set stargate neighbour sta2 sta1 start game place settler set1 sta2 list turn events list all </pre>
Teszt kimenete	<pre> # turn events: stargate placed sta2 ast2 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast1 asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: sta1 settlers: settler: - nonPlayers: nonPlayer: - ----- Asteroid id: ast2 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: sta2 </pre>

	<pre> settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: true asteroid: ast2 inventory: Inventory id: inv1 materials: material: - starGates: starGate: - ----- StarGate id: sta1 isRandom: f works: t wasInSunFlare: f neighbour: sta2 asteroid: ast1 ----- StarGate id: sta2 isRandom: f works: t wasInSunFlare: f neighbour: sta1 asteroid: ast2 -----</pre>
--	---

5.5.18 Napvihar kitörés nem átfúrt aszteroidán

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create sun sun2 create asteroid ast1 sun sun1 set asteroid rockThickness ast1 1 create asteroid ast2 sun sun2 add asteroid neighbour ast1 ast2 add asteroid neighbour ast2 ast1 create settler set1 asteroid ast1 create robot rob1 asteroid ast1 create alien ali1 asteroid ast1 create stargate sta1 asteroid ast1 set stargate isRandom sta1 t create inventory inv1 settler set1 create stargate sta2 settler set1 set stargate neighbour sta1 sta2</pre>

	<pre>set stargate neighbour sta2 sta1 start game sunflare sun1 list turn events list all</pre>
Teszt kimenete	<pre># turn events: sunflare on sun sun1 stargate broke sta1 settler died set1 robot died rob1 alien died ali1 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- SunId: sun2 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 1 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: ast2 buildings: building: sta1 settlers: settler: - nonPlayers: nonPlayer: - ----- Asteroid id: ast2 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: ast1 buildings:</pre>

```

building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadAcrtionThisTurn: false
asteroid: ast1
inventory: Inventory id: inv1
materials:
    material: -
starGates:
    starGate: sta2
-----
Robot id: rob1
isRandom: false
asteroid: ast1
-----
Alien id: ali1
isRandom: f
asteroid: ast1
-----
StarGate id: sta1
isRandom: t
works: f
wasInSunFlare: t
neighbour: sta2
asteroid: ast1
-----
StarGate id: sta2
isRandom: f
works: f
wasInSunFlare: f
neighbour: sta1
asteroid: -
-----
```

5.5.19 Napvihar kitörés átfúrt aszteroidán

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	init game manually create sun sun1 create sun sun2 create asteroid ast1 sun sun1 set asteroid rockThickness ast1 0 create asteroid ast2 sun sun2 add asteroid neighbour ast1 ast2

	<pre> add asteroid neighbour ast2 ast1 create settler set1 asteroid ast1 create robot rob1 asteroid ast1 create alien ali1 asteroid ast1 create stargate sta1 asteroid ast1 set stargate isRandom sta1 t create inventory inv1 settler set1 create stargate sta2 settler set1 set stargate neighbour sta1 sta2 set stargate neighbour sta2 sta1 start game sunflare sun1 list turn events list all </pre>
Teszt kimenete	<pre> # turn events: sunflare on sun sun1 stargate broke sta1 # list all: SunId: sun2 isRandom: f asteroids: asteroid: ast ----- SunId: sun2 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: ast2 buildings: building: sta1 settlers: settler: set1 nonPlayers: nonPlayer: rob1 nonPlayer: ali1 ----- Asteroid id: ast2 rockThickness: 0 </pre>

	<pre> closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: ast1 buildings: building: - settlers: settler: - nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: false asteroid: ast1 inventory: Inventory id: inv1 materials: material: - starGates: starGate: sta2 ----- Robot id: rob1 isRandom: false asteroid: ast1 ----- Alien id: ali1 isRandom: f asteroid: ast1 ----- StarGate id: sta1 isRandom: t works: f wasInSunFlare: t neighbour: sta2 asteroid: ast1 ----- StarGate id: sta2 isRandom: f works: f wasInSunFlare: f neighbour: sta1 asteroid: - -----</pre>
--	--

5.5.20 Fúrás robottal

Tesztelő neve	Tóth Ádám
---------------	-----------

Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 set asteroid rockThickness ast1 2 create robot rob1 asteroid ast1 start game drill robot rob1 list turn events start next turn drill robot rob1 list turn events start next turn drill robot rob1 list turn events start next turn list all</pre>
Teszt kimenete	<pre>The game is lost! # turn events: asteroid drilled ast1 asteroid drilled ast1 The game is lost! # turn events: asteroid drilled ast1 asteroid drilled ast1 The game is lost! # turn events: asteroid drilled ast1 asteroid drilled ast1 The game is lost! # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings:</pre>

	<pre> building: - settlers: settler: - nonPlayers: nonPlayer: rob1 ----- Robot id: rob1 isRandom: false asteroid: ast1 -----</pre>
--	--

5.5.21 Fúrás robottal eltárolt

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create robot rob1 asteroid ast1 set asteroid capacity ast1 10 create ice ice1 asteroid ast1 set material exposedCounter ice1 0 set asteroid rockThickness ast1 1 set asteroid closeToSun ast1 t start game drill robot rob1 list turn events list all</pre>
Teszt kimenete	<pre> The game is lost! # turn events: asteroid drilled ast1 material evaporated ice1 asteroid drilled ast1 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: t isRandom: f materials: materials: - neighbours:</pre>

	<pre> neighbour: - buildings: building: - settlers: settler: - nonPlayers: nonPlayer: rob1 ----- Robot id: rob1 isRandom: false asteroid: ast1 -----</pre>
--	--

5.5.22 Fúrás robottal, felrobban

Tesztelő neve	Tóth Ádám
Teszt időpontja	2021.04.18. 20:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create robot rob1 asteroid ast1 set asteroid capacity ast1 10 create uran ura1 asteroid ast1 set material exposedCounter ura1 0 set asteroid rockThickness ast1 1 set asteroid closeToSun ast1 t create asteroid ast2 sun sun1 add asteroid neighbour ast1 ast2 add asteroid neighbour ast2 ast1 start game drill robot rob1 list turn events list all</pre>
Teszt kimenete	<pre> The game is lost! # turn events: asteroid drilled ast1 asteroid drilled ast1 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast1 asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: t</pre>

	<pre> isRandom: f materials: material: ura1 neighbours: neighbour: ast2 buildings: building: - settlers: settler: - nonPlayers: nonPlayer: rob1 ----- Asteroid id: ast2 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: ast1 buildings: building: - settlers: settler: - nonPlayers: nonPlayer: - ----- Robot id: rob1 isRandom: false asteroid: ast1 -----</pre>
--	---

5.5.23 Fúrás robottal, felrobban, nincs szomszéd

Tesztelő neve	Bárkányi Csaba
Teszt időpontja	2021.04.19.01:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create robot rob1 asteroid ast1 set asteroid capacity ast1 10 create uran ura1 asteroid ast1 set material exposedCounter ura1 0 set asteroid rockThickness ast1 1 set asteroid closeToSun ast1 t create asteroid ast2 sun sun1 start game drill robot rob1 list turn events</pre>

	start next turn start next turn list turn events start next turn list all
Teszt kimenete	<pre>The game is lost! # turn events: asteroid drilled ast1 asteroid drilled ast1 The game is lost! The game is lost! # turn events: asteroid exploded ast1 robot died rob1 material exploded ura1 The game is lost! # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast2 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: - nonPlayers: nonPlayer: - -----</pre>

5.5.24 Fúrás telepessel

Tesztelő neve	Bárkányi Csaba
Teszt időpontja	2021.04.19.01:00
Teszt szintaktikája	init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1

	<pre> set asteroid rockThickness ast1 2 set settler hadActionThisTurn set1 f start game drill settler set1 list turn events start next turn drill settler set1 list turn events start next turn drill settler set1 list turn events start next turn list all </pre>
Teszt kimenete	<pre> # turn events: asteroid drilled ast1 # turn events: asteroid drilled ast1 # turn events: asteroid drilled ast1 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadAcrtionThisTurn: false asteroid: ast1 inventory: Inventory id: inv2 materials: </pre>

	<pre>material: - starGates: starGate: - -----</pre>
--	---

5.5.25 Fúrás telepessel, elpárolog

Tesztelő neve	Bárkányi Csaba
Teszt időpontja	2021.04.19.01:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 set settler hadActionThisTurn set1 f set asteroid capacity ast1 10 create ice ice1 asteroid ast1 set material exposedCounter ice1 0 set asteroid rockThickness ast1 1 set asteroid closeToSun ast1 t start game drill settler set1 list turn events list all</pre>
Teszt kimenete	<pre># turn events: asteroid drilled ast1 material evaporated ice1 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: t isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: -</pre>

	<pre> Settler id: set1 hadActionThisTurn: true asteroid: ast1 inventory: Inventory id: inv2 materials: material: - starGates: starGate: - -----</pre>
--	---

5.5.26 Fúrás telepessel, felrobban

Tesztelő neve	Bárkányi Csaba
Teszt időpontja	2021.04.19. 01:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create settler set1 asteroid ast1 set settler hadActionThisTurn set1 f set asteroid capacity ast1 10 create uran ura1 asteroid ast1 set material exposedCounter ura1 0 set asteroid rockThickness ast1 1 set asteroid closeToSun ast1 t start game drill settler set1 list turn events start next turn start next turn list turn events start next turn list all</pre>
Teszt kimenete	<pre> # turn events: asteroid drilled ast1 # turn events: asteroid exploded ast1 settler died set1 material exploded ura1 The game is lost! # list all: SunId: sun1 isRandom: f asteroids: asteroid: -----</pre>

5.5.27 Mozgás robottal, kudarc

Tesztelő neve	Bárkányi Csaba
Teszt időpontja	2021.04.19. 01:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 create asteroid ast2 sun sun1 create robot rob1 asteroid ast1 start game move robot rob1 dest asteroid ast2 list turn events list all</pre>
Teszt kimenete	<pre>The game is lost! # turn events: robot move rob1 ast2 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast1 asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers: settler: - nonPlayers: nonPlayer: rob1 ----- Asteroid id: ast2 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: - settlers:</pre>

	<pre> settler: - nonPlayers: nonPlayer: - ----- Robot id: rob1 isRandom: false asteroid: ast1 -----</pre>
--	--

5.5.28 Mozgás robottal, siker

Tesztelő neve	Bárkányi Csaba
Teszt időpontja	2021.04.19.01:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create asteroid ast2 sun sun1 create robot rob1 asteroid ast1 add asteroid neighbour ast1 ast2 add asteroid neighbour ast2 ast1 start game move robot rob1 dest asteroid ast2 list turn events list all</pre>
Teszt kimenete	<pre> The game is lost! # turn events: robot move rob1 ast2 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast1 asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: ast2 buildings: building: - settlers: settler: - nonPlayers: nonPlayer: -</pre>

	<pre> ----- Asteroid id: ast2 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: ast1 buildings: building: - settlers: settler: - nonPlayers: nonPlayer: rob1 ----- Robot id: rob1 isRandom: false asteroid: ast2 -----</pre>
--	--

5.5.29 Mozgás telepessel, kudarc

Tesztelő neve	Bárkányi Csaba
Teszt időpontja	2021.04.19.01:00
Teszt szintaktikája	<pre> init game manually create sun sun1 create asteroid ast1 sun sun1 create asteroid ast2 sun sun1 create settler set1 asteroid ast1 set settler hadActionThisTurn set1 f start game move settler set1 dest asteroid ast2 list turn events list all</pre>
Teszt kimenete	<pre> # turn events: - # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast1 asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f</pre>

```

materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: set1
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast2
rockThickness: 0
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Settler id: set1
hadActionThisTurn: true
asteroid: ast1
inventory: Inventory id: inv3
materials:
    material: -
starGates:
    starGate: -
-----
```

5.5.30 Mozgás telepessel, sikér

Tesztelő neve	Bárkányi Csaba
Teszt időpontja	2021.04.19.01:00
Teszt szintaktikája	init game manually create sun sun1 create asteroid ast1 sun sun1 create asteroid ast2 sun sun1 create settler set1 asteroid ast1 set settler hadActionThisTurn set1 f add asteroid neighbour ast1 ast2 add asteroid neighbour ast2 ast1 start game

	move settler set1 dest asteroid ast2 list turn events list all
Teszt kimenete	<pre># turn events: settler move set1 ast2 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast1 asteroid: ast ----- Asteroid id: ast1 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: ast2 buildings: building: - settlers: settler: - nonPlayers: nonPlayer: - ----- Asteroid id: ast2 rockThickness: 0 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: ast1 buildings: building: - settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadActionThisTurn: true asteroid: ast2 inventory: Inventory id: inv3 materials: material: -</pre>

	<pre>starGates: starGate: - -----</pre>
--	---

5.5.31 Telepes átlép egy kapun

Tesztelő neve	Bárkányi Csaba
Teszt időpontja	2021.04.19. 01:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create asteroid ast1 sun sun1 set asteroid rockThickness ast1 1 create asteroid ast2 sun sun1 set asteroid rockThickness ast2 1 create settler set1 asteroid ast1 create stargate sta1 asteroid ast1 set stargate works sta1 t create stargate sta2 asteroid ast2 set stargate works sta2 t set stargate neighbour sta1 sta2 set stargate neighbour sta2 sta1 start game move settler set1 dest stargate sta1 list turn events list all</pre>
Teszt kimenete	<pre># turn events: settler move set1 ast1 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast1 asteroid: ast ----- Asteroid id: ast1 rockThickness: 1 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: sta1 settlers: settler: - nonPlayers: nonPlayer: -</pre>

	<pre> ----- Asteroid id: ast2 rockThickness: 1 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: - buildings: building: sta2 settlers: settler: set1 nonPlayers: nonPlayer: - ----- Settler id: set1 hadActionThisTurn: true asteroid: ast2 inventory: Inventory id: inv3 materials: material: - starGates: starGate: - ----- StarGate id: sta1 isRandom: f works: t wasInSunFlare: f neighbour: sta2 asteroid: ast1 ----- StarGate id: sta2 isRandom: f works: t wasInSunFlare: f neighbour: sta1 asteroid: ast2 -----</pre>
--	--

5.5.32 Teleportkapu elromlik

Tesztelő neve	Bárkányi Csaba
Teszt időpontja	2021.04.19.01:00
Teszt szintaktikája	<pre>init game manually create sun sun1 create sun sun2 create asteroid ast1 sun sun1 set asteroid rockThickness ast1 1</pre>

	<pre> create asteroid ast2 sun sun2 set asteroid rockThickness ast2 1 create asteroid ast3 sun sun2 set asteroid rockThickness ast3 1 add asteroid neighbour ast1 ast3 add asteroid neighbour ast3 ast1 create stargate sta1 asteroid ast1 set stargate isRandom sta1 t create stargate sta2 asteroid ast2 set stargate isRandom sta2 t set stargate neighbour sta1 sta2 set stargate neighbour sta2 sta1 start game sunflare sun1 list turn events start next turn list turn events list all </pre>
Teszt kimenete	<pre> The game is lost! # turn events: sunflare on sun sun1 stargate broke sta1 The game is lost! # turn events: stargate move sta1 ast3 # list all: SunId: sun1 isRandom: f asteroids: asteroid: ast ----- SunId: sun2 isRandom: f asteroids: asteroid: ast2 asteroid: ast ----- Asteroid id: ast1 rockThickness: 1 closeToSun: f isRandom: f materials: materials: - neighbours: neighbour: ast3 buildings: building: - settlers: </pre>

```

        settler: -
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast2
rockThickness: 1
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: -
buildings:
    building: sta2
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
Asteroid id: ast3
rockThickness: 1
closeToSun: f
isRandom: f
materials:
    materials: -
neighbours:
    neighbour: ast1
buildings:
    building: sta1
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
-----
StarGate id: sta2
isRandom: t
works: f
wasInSunFlare: f
neighbour: sta1
asteroid: ast2
-----
StarGate id: sta1
isRandom: t
works: f
wasInSunFlare: t
neighbour: sta2
asteroid: ast3
-----
```

5.6 Értékelés

Tag neve	Tag neptun	Munka százalékban
Mohácsi Márton	T8ZT88	20%
Simon Zoltán	HRSNUG	20%
Kárpáti Márk András	O1BG0Z	20%
Bárkányi Csaba	BQI7QQ	20%
Tóth Ádám	QEYOW2	20%

5.7 Napló

Kezdet	Időtartam	Résznevők	Leírás
2010.04.13. 16:00	2 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Értekezlet. Megbeszéltük a korábbi eredményeket. És szétosztottuk a feladatokat: Mohácsi: Game, UI, Controllable, Printable Simon: Building, StarGate, Base, Alien Kárpáti: Asteroid, Sun, Planet, ReactsToSunflare Bárkányi: Material (összes leszármazottjai is), Inventory Tóth: Entity, Mining, Driling, Moving, NonPlayer, Robot
2010.04.14. 11:00	1 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Apró pontosítások.
2010.04.16. 16:00	2 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Felvetések megvitatása. Programhibák kijavítása.
2010.04.17. 16:00	3 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Felvetések megvitatása. További hibák javítása.
2010.04.18. 20:00	4 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	További hibák javítása és tesztelés.
2010.04.19. 10:00	4 óra	Mohácsi Simon Kárpáti	További hibák javítása és tesztelés.

		Bárkányi Tóth	Dokumentum szerkesztése. Utolsó simítások.
--	--	------------------	--

11. Grafikus felület specifikációja

67 – brainstormers

Konzulens:
Simon Balázs

Csapattagok

Mohácsi Márton

Simon Zoltán

Kárpáti Márk András

Bárkányi Csaba

Tóth Ádám

T8ZT88

HRSNUG

O1BG0Z

BQI7QQ

QEYOW2

moha.mohacsi@gmail.com

simonzolika1999@gmail.com

mark.karpati1999@gmail.com

csaba.barkanyi.csaba@gmail.com

toth99adam@gmail.com

2021-04-26

Tartalomjegyzék

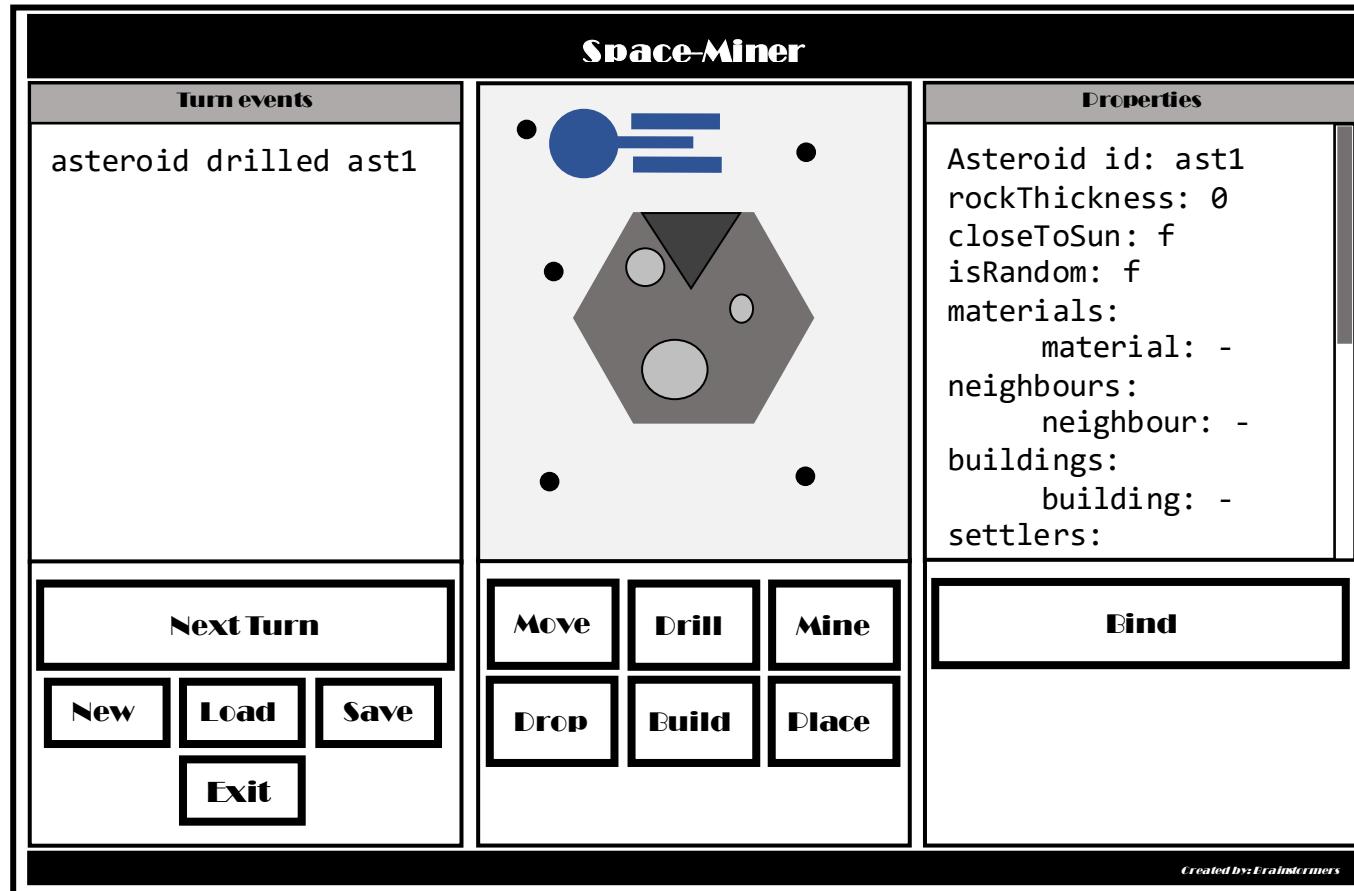
Tartalomjegyzék.....	248
11. Grafikus felület specifikációja.....	249
11.1 A grafikus interfész	249
11.2 A grafikus rendszer architektúrája.....	253
11.2.1 Struktúra diagram	253
11.2.2 A felület működési elve.....	254
11.2.3 A felület osztály-struktúrája.....	256
11.3 A grafikus objektumok felsorolása	256
11.3.1 BindDialog	256
11.1.2 BindDialog	257
11.1.3 Controller.....	258
11.1.4 DialogSelect1	258
11.1.5 DialogSelect2	259
11.1.6 DialogSelect3	260
11.1.7 DrillDialog	260
11.1.8 DropDialog.....	261
11.1.9 Game.....	261
11.1.10 Main	262
11.1.11 MineDialog.....	263
11.1.12 MoveDialog	263
11.1.13 PlaceDialog.....	264
11.1.14 View	265
11.2 Kapcsolat az alkalmazói rendszerrel.....	267
11.2.1 Build	267
11.3 Napló.....	272

9. Grafikus felület specifikációja

9.1 A grafikus interfész

9.1.1.1 Alkalmazás tervezett fő kezelőfelületi képe

A felületterv a jobb értelmezhetőség miatt példaadatokkal is ellátásra került.



9.1.1.2 Bind felugró ablak tervezett elrendezése

Bind	
Type:	
Id:	
Select	

9.1.1.3 Move felugró ablak tervezett elrendezése

Move with settler	
Destination Type:	
Id:	
SettlerId:	
Select	

9.1.1.4 Drill felugró ablak tervezett elrendezése

Drill with settler	
SettlerId:	
Select	

9.1.1.5 Mine felugró ablak tervezett elrendezése

Mine with settler	
SettlerId:	
Select	

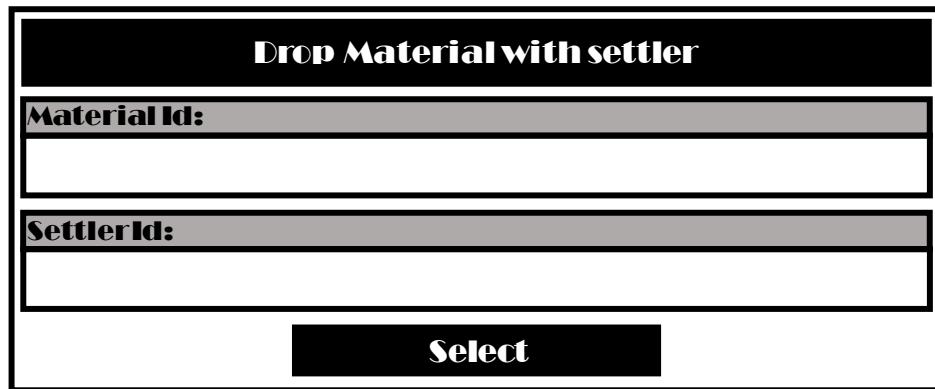
9.1.1.6 Build felugró ablak tervezett elrendezése

Build with settler	
Buildable:	
SettlerId:	
Select	

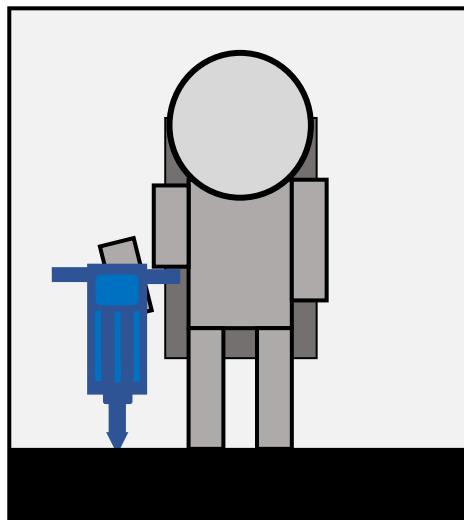
9.1.1.7 Place felugró ablak tervezett elrendezése

Place Gate with settler	
Gate Id:	
SettlerId:	
Select	

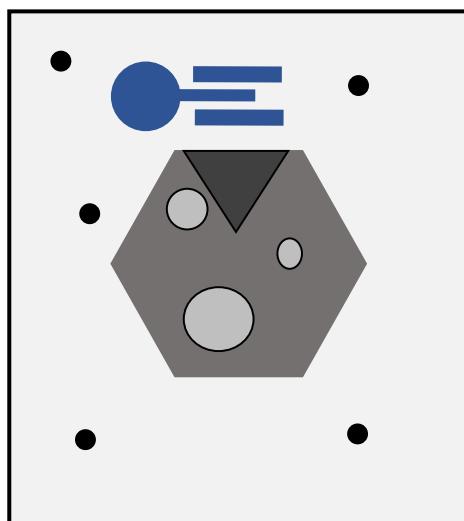
9.1.1.8 Drop felugró ablak tervezett elrendezése



9.1.1.9 Telepes tervezett megjelenítése



9.1.1.10 Aszteroida tervezett megjelenítése



9.2 A grafikus rendszer architektúrája

Grafikus elemek csoportjai:

9. EventFeed

10. GameControls

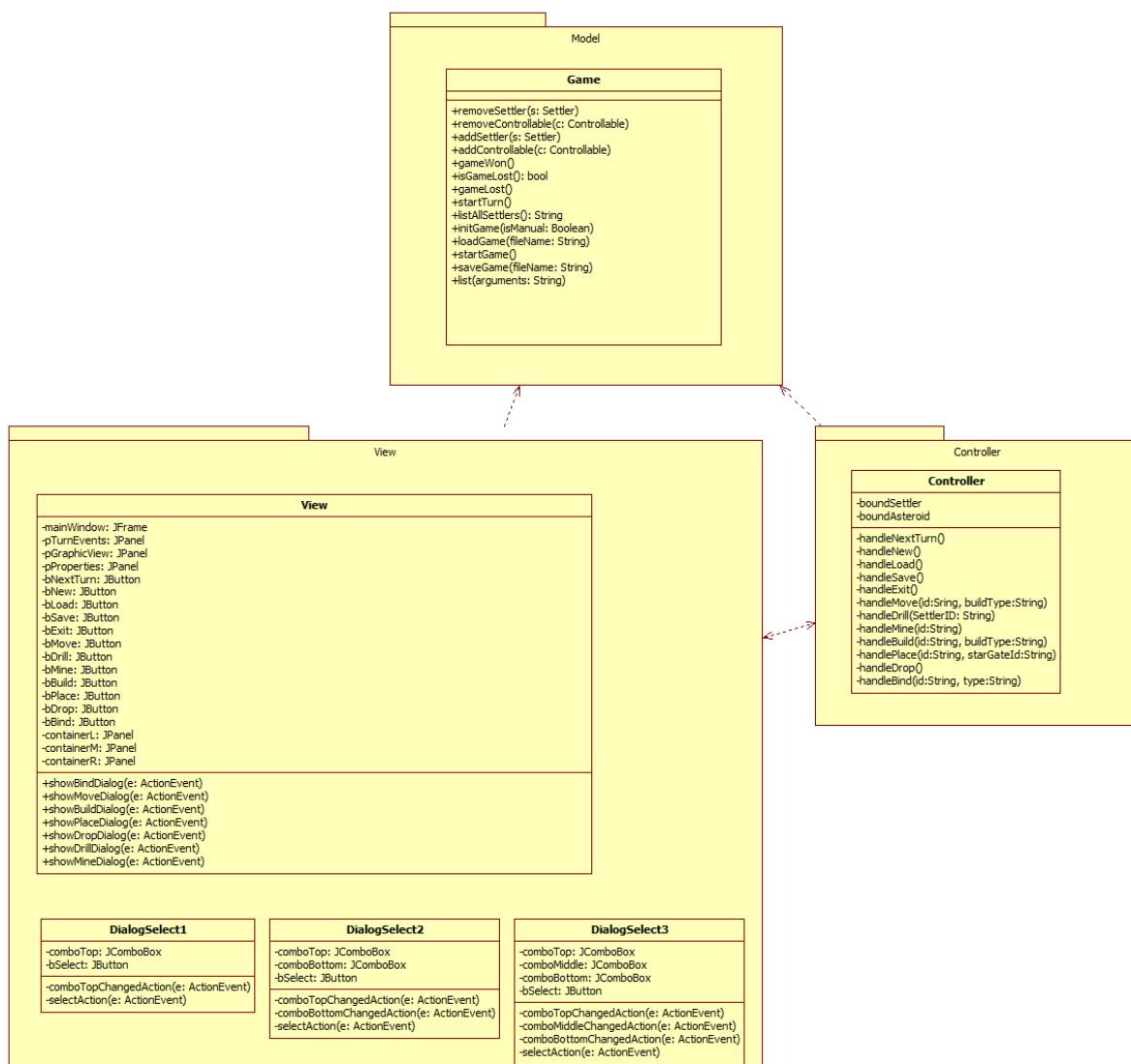
11. GraphicView

12. EntityControls

13. Properties

14. Bind

9.2.1 Struktúra diagram



9.2.2 A felület működési elve

9.2.2.1 Áttekintés:

Az applikáció MVC felépítésű. A felületelemek vizuálisan három csoportba szerveződnek. Ezek a köresemények listázása és játék kezelés, a kiválasztott objektum megjelenítése és annak kezelése és végül a kiválasztott objektum tulajdonságainak listázása és objektum választás. A felsoroltak tehát további két részre bomlanak: View és Controller. Az ablakban felül megjelenő mezők a View részei a lenti gombok pedig a Controller részei.

9.2.2.2 A Controller és View részletesebb bemutatása:

9.2.2.2.1 TurnEvents

Szemantikailag a view része. A köresemények listázása egy Java Swing szövegdoboz. Ebben az egyes játékkörök végén kiíródnak, az adott körben végbement események.
pl.: mozgás, bányászás, robbanás stb.

A Controller handleNextTurn() tölti fel új tartalommal.

9.2.2.2.2 GameControls

Szemantikailag a controller része. Ez A View-ban gombokként jelenik meg. A Next turn gomb hatására a Controller handleNextTurn()-je befejezi a kört és új kört indít Game hívásokkal. A New gombra kattintva a Controller handleNew() függvénye új játék kezd. A Load gombra kattintva a Controller handleLoad() függvénye betölt egy korábbi játékállást fájlból. A Save gombra kattintva a Controller handleSave() függvénye elmenti a játékállást. Az Exit gombra kattintva a Controller handleExit() függvénye leállítja a programot.

9.2.2.2.3 GraphicView

Szemantikailag a view része. Egy panel tartozik hozzá. Megjelenít egy ábrát, ami az éppen bindelt objetumtól függ, azt ábrázolja. A Controller handleBind() függvénye adja át a képet, amit meg kell jelenítenie.

9.2.2.2.4 EntityControls

Szemantikailag a controller része. A View-ben gombok tartoznak hozzá. A gombok lenyomására dialógusok jelennek meg. Ezek szintén a view részei.

A Move gomb lenyomásának hatására megjelenik egy dialógus, amiben kiválaszthatjuk, hogy melyik settler mozogjon, aszteroidára vagy kapura lépjen és konkrétan mi legyen a célja. A "Select" gombbal bezárul a dialógus és a Controller handleMove() függvénye továbbítja a Game felé a parancsot.

A Drill gomb lenyomásának hatására megjelenik egy dialógus, amiben kiválaszthatjuk, hogy melyik settler fúrjon.

A "Select" gombbal bezárul a dialógus és a Controller handleDrill() függvénye továbbítja a Game felé a parancsot.

A Mine gomb lenyomásának hatására megjelenik egy dialógus, amiben kiválaszthatjuk, hogy melyik settler bányásszon.

A "Select" gombbal bezárul a dialógus és a Controller handleMine() függvénye továbbítja a Game felé a parancsot.

A Drop gomb lenyomásának hatására megjelenik egy dialógus, amiben kiválaszthatjuk, hogy melyik settler tegyen le nyersanyagot és melyik nyersanyagot tegye le.

A "Select" gombbal bezárul a dialógus és a Controller handleDrop() függvénye továbbítja a Game felé a parancsot.

A Build gomb lenyomásának hatására megjelenik egy dialógus, amiben kiválaszthatjuk, hogy melyik settler építsen és mit építsen.

A "Select" gombbal bezárul a dialógus és a Controller handleBuild() függvénye továbbítja a Game felé a parancsot.

A Place gomb lenyomásának hatására megjelenik egy dialógus, amiben kiválaszthatjuk, hogy melyik settler rakjon le épületet és melyik már megépített épületet tegye le.

A "Select" gombbal bezárul a dialógus és a Controller handlePlace() függvénye továbbítja a Game felé a parancsot.

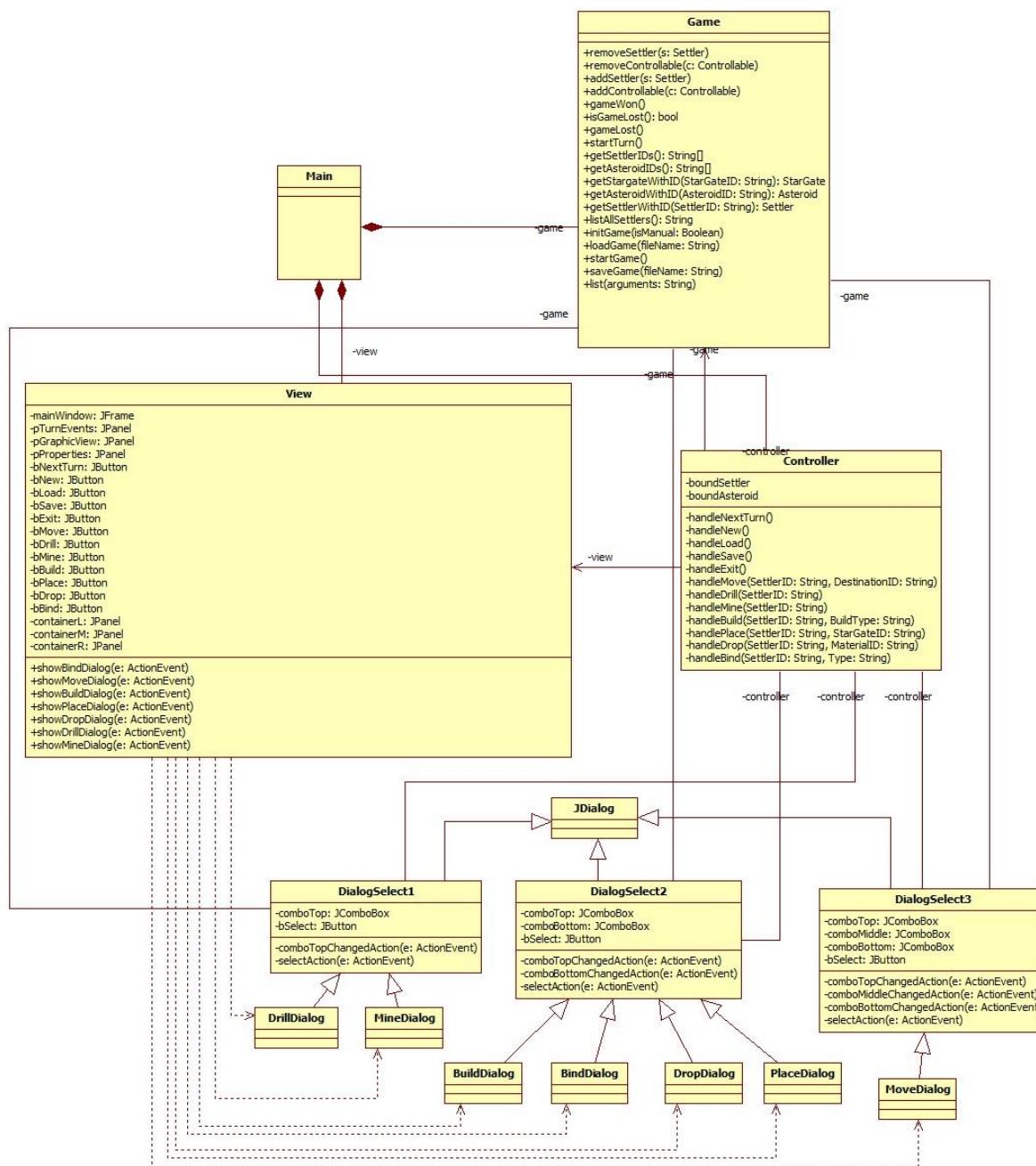
9.2.2.5 Properties

Szemantikailag a view része. A View-ban egy panel tartozik hozzá, ami kiírja az aktuálisan kiválasztott settler vagy asteroid adatait. A Controller handleBind() függvénye lekérdezi a Game-ból a bind-olt objektum adatait (pull), majd továbbadja a View-ba a Properties panelnek.

9.2.2.6 Bind

Szemantikailag a controller része. A View-ben egy gomb és egy megjelenő dialógus tartozik hozzá. A Bind gombra kattintva megjelenik egy dialógus, amiben kiválaszthatjuk, hogy aszteroidát vagy telepeset szeretnénk bind-olni. (Az aszteroidáról is megjeleníthetünk adatokat.) A "Select" gombbal bezárul a dialógus és a Controller handleBind() függvénye lekérdezi a Game-ból a bind-olt objektum adatait (pull), majd továbbadja a View-ba a Properties panelnek.

9.2.3 A felület osztály-struktúrája



9.3 A grafikus objektumok felsorolása

9.3.1 BindDialog

- Felelősség**

Ez az osztály a bind gomb által létrehozott dialógus ablakot valósítja meg.

- Ősosztályok**

`JDialog->DialogSelect2->BindDialog`

- **Interfészek**
 -
- **Attribútumok**
 - **game: Game:** Egy referencia a játék osztályra, innen tudjuk updatelni a dialógusablak megfelelő comboboxainak listáit, ha változik egy másik combobox szelekciója. Például, ha a felső comboboxban kiválasztjuk az aszteroida típust, az alsó comboboxot updatelni kell, hogy abban az aszteroidák idjei legyenek.
 - **comboTop:** JComboBox: A legfelső combobox a dialógusablakon. A bindolt objektum típusát lehet benne kiválasztani.
 - **comboBottom:** JComboBox: A legalasó combobox a dialógusablakon. A bindolni kívánt objektum idjét lehet benne kiválasztani.
 - **bSelect:** JButton: A gomb amivel leokézható a dialógusablak.
- **Metódusok**
 - **comboTopChangedAction(e: ActionEvent):** A comboTop szelekciójának megváltoztatásakor elsütődő event. Ha megváltozik a típus az alsó combobox id listáját updatelni kell.
 - **comboBottomChangedAction(e: ActionEvent):** A comboBottom szelekciójának megváltoztatásakor elsütődő event. Ez ebben a dialógus ablakban nincs használva.
 - **selectAction(e: ActionEvent):** A bSelect gomb lenyomásakor elsütődő esemény.
- **BindDialog**
 - **Felelősség**
Ez az osztály a build gomb által létrehozott dialógus ablakot valósítja meg.
 - **Ősosztályok**
JDialog->DialogSelect2->BuildDialog
 - **Interfészek**
 -
 - **Attribútumok**
 - **game: Game:** Egy referencia a játék osztályra, innen tudjuk updatelni a dialógusablak megfelelő comboboxainak listáit ha változik egy másik combobox szelekciója. Ez ebben a dialógus ablakban nincs használva.
 - **comboTop:** JComboBox: A legfelső combobox a dialógusablakon. A bindolni kívánt objektum típusát lehet benne kiválasztani.
 - **comboBottom:** JComboBox: A legalasó combobox a dialógusablakon. A bindolni kívánt objektum id-jét lehet benne kiválasztani
 - **bSelect:** JButton: A gomb amivel leokézható a dialógusablak.
 - **Metódusok**
 - **comboTopChangedAction(e: ActionEvent):** A comboTop szelekciójának megváltoztatásakor elsütődő event. Ez ebben a dialógus ablakban nincs használva.

- **comboBottonChangedAction(e: ActionEvent):** A comboBottom szelekciójának megváltoztatásakor elsütődő event. Ez ebben a dialógus ablakban nincs használva.
- **selectAction(e: ActionEvent):** A bSelect gomb lenyomásakor elsütődő esemény.

▪ Controller

• Felelősség

Ez az osztály felel a játékos View-en keresztül beadott utasításainak a gamehez (vagy modellhez) való eljuttatásáért. Mostmár a modellünkben nincs külön UI osztály, szét lett szedve a Controller-s a View osztályokra.

• Ősosztályok

-

• Interfészek

-

• Attribútumok

- **view: View:** Egy referencia a játékot megjelenítő View osztályhoz.
- **game: Game:** Egy referencia a játék modelljéhez, a gamehez.
- **boundSettler: Settler:** A legutóbb megjelenített (bebindolt) settllerre mutató referencia.
- **boundAsteroid: Asteroid:** A legutóbb megjelenített (bebindolt) asteroidra mutató referencia.

• Metódusok

- **handleNextTurn():** A következő kör utasítást a gamenek átfordító függvény.
- **handleNew():** Az új játék utasítást a gamenek átfordító függvény.
- **handleLoad():** A játék betöltése utasítást a gamenek átfordító függvény.
- **handleSave():** A játék mentése utasítást a gamenek átfordító függvény.
- **handleExit():** A kilépést végző függvény.
- **handleMove(id: String, destinationType: String, destination: String):** A mozgás utasítást a gamenek átfordító függvény.
- **handleDrill(id: String):** A fúrás utasítást a gamenek átfordító függvény.
- **handleMine(id: String):** A bányászást a gamenek átfordító függvény.
- **handleBuild(id: String, buildType: String):** Az építést a gamenek átfordító függvény
- **handlePlace(id: String, starGateId: String):** A StarGate lehelyezését a gamenek átfordító függvény.
- **handleBind(id: String, type: String):** A bind utasítást a gamenek átfordító függvény.

▪ DialogSelect1

• Felelősség

Ez az osztály egy generikus típusa az egy comboboxú dialógusablaknak.

- **Ősosztályok**

JDialog->DialogSelect1

- **Interfészek**

-

- **Attribútumok**

- **game: Game:** Egy referencia a játék osztályra, innen tudjuk updatelni a dialógusablak megfelelő comboboxainak listáit ha változik egy másik combobox szelekciója.
- **comboTop:** JComboBox: A legfelső combobox a dialógusablakon.
- **bSelect:** JButton: A gomb amivel leköthető a dialógusablak.

- **Metódusok**

- **comboTopChangedAction(e: ActionEvent):** A comboTop szelekciójának megváltoztatásakor elsütődő event.
- **selectAction(e: ActionEvent):** A bSelect gomb lenyomásakor elsütődő esemény.

- **DialogSelect2**

- **Felelősség**

Ez az osztály egy generikus típusa a két comboboxú dialógusablaknak.

- **Ősosztályok**

JDialog->DialogSelect2

- **Interfészek**

-

- **Attribútumok**

- **game: Game:** Egy referencia a játék osztályra, innen tudjuk updatelni a dialógusablak megfelelő comboboxainak listáit ha változik egy másik combobox szelekciója.
- **comboTop:** JComboBox: A legfelső combobox a dialógusablakon.
- **comboBottom:** JComboBox: A legalsó combobox a dialógusablakon.
- **bSelect:** JButton: A gomb amivel leköthető a dialógusablak.

- **Metódusok**

- **comboTopChangedAction(e: ActionEvent):** A comboTop szelekciójának megváltoztatásakor elsütődő event.
- **comboBottomChangedAction(e: ActionEvent):** A comboBottom szelekciójának megváltoztatásakor elsütődő event.
- **selectAction(e: ActionEvent):** A bSelect gomb lenyomásakor elsütődő esemény.

▪ DialogSelect3

• Felelősség

Ez az osztály egy generikus típusa a három comboboxú dialógusablaknak.

• Ősosztályok

JDialog->DialogSelect3

• Interfészek

-

• Attribútumok

- **game: Game:** Egy referencia a játék osztályra, innen tudjuk updatelni a dialógusablak megfelelő comboboxainak listáit ha változik egy másik combobox szelekciója.
- **comboTop:** JComboBox: A legfelső combobox a dialógusablakon.
- **comboMiddle:** JComboBox: A középső combobox a dialógusablakon.
- **comboBottom:** JComboBox: A legalsó combobox a dialógusablakon.
- **bSelect:** JButton: A gomb amivel lekövethető a dialógusablak.

• Metódusok

- **comboTopChangedAction(e: ActionEvent):** A comboTop szelekciójának megváltoztatásakor elsütődő esemény.
- **comboMiddleChangedAction(e: ActionEvent):** A comboMiddle szelekciójának megváltoztatásakor elsütődő esemény.
- **comboBottomChangedAction(e: ActionEvent):** A comboBottom szelekciójának megváltoztatásakor elsütődő esemény.
- **selectAction(e: ActionEvent):** A bSelect gomb lenyomásakor elsütődő esemény.

▪ DrillDialog

• Felelősség

Ez az osztály a drill gomb által létrehozott dialógus ablakot valósítja meg.

• Ősosztályok

JDialog->DialogSelect1->DrillDialog

• Interfészek

-

• Attribútumok

- **game: Game:** Egy referencia a játék osztályra, innen tudjuk updatelni a dialógusablak megfelelő comboboxainak listáit ha változik egy másik combobox szelekciója. Ez ebben a dialógus ablakban nincs használva.
- **comboTop:** JComboBox: A legfelső combobox a dialógusablakon. A settler id-jét lehet kiválasztani benne mellyel fűrni akarunk.
- **bSelect:** JButton: A gomb amivel lekövethető a dialógusablak.

- **Metódusok**
 - **comboTopChangedAction(e: ActionEvent):** A comboTop szelekciójának megváltoztatásakor elsütődő event. Ez ebben a dialógus ablakban nincs használva.
 - **selectAction(e: ActionEvent):** A bSelect gomb lenyomásakor elsütődő esemény.

■ DropDialog

- **Felelősség**

Ez az osztály a drop gomb által létrehozott dialógus ablakot valósítja meg.

- **Ősosztályok**

JDialog->DialogSelect2

- **Interfészek**
-

● Attribútumok

- **game: Game:** Egy referencia a játék osztályra, innen tudjuk updatelni a dialógusablak megfelelő comboboxainak listáit ha változik egy másik combobox szelekciója. Például ha a felső comboboxban megváltoztatjuk a settler id-t akkor az alsó comboboxban meg kell változtatnunk a material id-k listáját (itt minden azoknak a materialoknak kell lenniük melyek az adott settler inventoryjában vannak).
- **comboTop:** JComboBox: A legfelső combobox a dialógusablakon. A settler id-jét lehet benne kiválasztani amelyikkel el akarunk dobni egy nyersanyagot.
- **comboBottom:** JComboBox: A legalacso combobox a dialógusablakon. A material id-jét lehet benne kiválasztani amelyiket el akarjuk dobni.
- **bSelect:** JButton: A gomb amivel leköthető a dialógusablak.

● Metódusok

- **comboTopChangedAction(e: ActionEvent):** A comboTop szelekciójának megváltoztatásakor elsütődő event. Ha a settler id-je megváltoztatni az alsó combobox listáját updatelni kell az adott settler inventoryjában lévő materialok id-jével.
- **comboBottomChangedAction(e: ActionEvent):** A comboBottom szelekciójának megváltoztatásakor elsütődő event. Ez ebben a dialógus ablakban nincs használva.
- **selectAction(e: ActionEvent):** A bSelect gomb lenyomásakor elsütődő esemény.

■ Game

- **Felelősség**

A játék menetéért felelő osztály, nagyrészt változatlan de a grafikus megjelenítés érdekében új függvényekkel lett kiegészítve.

- **Ősosztályok**
-

- **Interfészek**
 -
- **Attribútumok**
 - **turnEvents: String[]:** Egy string tömb melyben a kör eventjeit tároljuk, ezt mindenféle osztály updateleheti az addTurnEvent függvényel, illetve a kör kezdetén törlődik.
- **Metódusok**
 - **addTurnEvent(s: String):** Hozzáadja az s stringet a turnEvents tömbhöz.
 - **listTurnEvents(): String** Visszaadjuk soronként kiírva a turnEvents tömböt.
 - **getSettlerIds(): String[]:** Egy string tömmbe szervezve visszaadjuk az összes játékban létező settler idjét.
 - **getAsteroidIds(): String[]:** Egy string tömmbe szervezve visszaadjuk az összes játékban létező asteroid idjét.
 - **getSettlerWithId(id: String): Settler:** Ez a függvény visszaadja az id id-jű Settlert, vagy nullt ha nincs ilyen.
 - **getAsteroidWithId(id: String): Asteroid:** Ez a függvény visszaadja az id id-jű Asteroidot, vagy nullt ha nincs ilyen.
 - **getSunWithId(id: String): Sun:** Ez a függvény visszaadja az id id-jű Sunt, vagy nullt ha nincs ilyen.
 - **getAlienWithId(id: String): Alien:** Ez a függvény visszaadja az id id-jű Alient, vagy nullt ha nincs ilyen.
 - **getRobotWithId(id: String): Robot:** Ez a függvény visszaadja az id id-jű Robotot, vagy nullt ha nincs ilyen.
 - **getStarGateWithId(id: String): StarGate:** Ez a függvény visszaadja az id id-jű StarGatet, vagy nullt ha nincs ilyen.
 - **getBaseWithId(id: String): Base:** Ez a függvény visszaadja az id id-jű Baset, vagy nullt ha nincs ilyen.
 - **getMaterialWithId(id: String): Material:** Ez a függvény visszaadja az id id-jű Materialt, vagy nullt ha nincs ilyen.
- **Main**
 - **Felelősség**

Ez az osztály inicializálja az egész programunk működését, innen fut a játék.
 - **Ősosztályok**
 -
 - **Interfészek**
 -
 - **Attribútumok**
 - **controller: Controller:** Egy referencia a View és a Modell közötti kommunikációt végző Controller osztályhoz.

- **view:** View: Egy referencia a játékot megjelenítő View osztályhoz.
- **game:** Game: Egy referencia a játék modelljéhez, a gamehez.

▪ **MineDialog**

- **Felelősség**

Ez az osztály a mine gomb által létrehozott dialógus ablakot valósítja meg.

- **Ősosztályok**

JDialog->DialogSelect1->MineDialog

- **Interfészek**

-

- **Attribútumok**

- **game: Game:** Egy referencia a játék osztályra, innen tudjuk updatelni a dialógusablak megfelelő comboboxainak listáit ha változik egy másik combobox szelekciója. Ez ebben a dialógus ablakban nincs használva.
- **comboTop:** JComboBox: A legfelső combobox a dialógusablakon. A settler id-jét lehet kiválasztani benne mellyel bányászni akarunk.
- **bSelect:** JButton: A gomb amivel leköthető a dialógusablak.

- **Metódusok**

- **comboTopChangedAction(e: ActionEvent):** A comboTop szelekciójának megváltoztatásakor elsütődő event. Ez ebben a dialógus ablakban nincs használva.
- **selectAction(e: ActionEvent):** A bSelect gomb lenyomásakor elsütődő esemény.

▪ **MoveDialog**

- **Felelősség**

Ez az osztály a move gomb által létrehozott dialógus ablakot valósítja meg.

- **Ősosztályok**

JDialog->DialogSelect3->MoveDialog

- **Interfészek**

-

- **Attribútumok**

- **game: Game:** Egy referencia a játék osztályra, innen tudjuk updatelni a dialógusablak megfelelő comboboxainak listáit ha változik egy másik combobox szelekciója. Például ha a settler id-je megváltozik akkor a harmadik combobox listáját (a második combobox szelekciójának függvényében) megfelelően updatelni kell.
- **comboTop:** JComboBox: A legfelső combobox a dialógusablakon. A mozgatni kívánt settler id-jét lehet benne kiválasztani.

- **comboMiddle:** JComboBox: A középső combobox a dialógusablakon. A destination típusát lehet benne kiválasztani (stargate/asteroid).
- **comboBottom:** JComboBox: A legalsó combobox a dialógusablakon. A destination id-jét lehet benne kiválasztani.
- **bSelect:** JButton: A gomb amivel leköthető a dialógusablak.

- **Metódusok**
- **comboTopChangedAction(e: ActionEvent):** A comboTop szelekciójának megváltoztatásakor elsütődő event. Ha megváltozik ez a combobox akkor a legalsó combobox listáját updatelni kell a második combobox függvényében. Ha a második combobox értéke asteroid akkor az ebben a comboboxban beállított id-jű settler aszteroidájának szomszédos aszteroidáinak id-jeit kell belehelyeznünk a harmadik comboboxba. Ha a második combobox érték stargate akkor az ebben a comboboxban beállított id-jű settler aszteroidáján található stargatek id-jeit kell belehelyeznünk a harmadik comboboxba.
- **comboMiddleChangedAction(e: ActionEvent):** A comboMiddle szelekciójának megváltoztatásakor elsütődő event. Ha megváltozik ez a combobox akkor a legalsó combobox listáját updatelni kell. Ha ez a combobox asteroidra változik akkor az első comboboxban specifikált id-jű settler aszteroidájának szomszédos aszteroidáinak id-jeit kell a legalsó comboboxba helyezni. Ha ez a combobox stargatere változik akkor az első comboboxban specifikált id-jű settler aszteroidáján található stargatek id-jeit kell a legalsó comboboxba helyezni.
- **comboBottomChangedAction(e: ActionEvent):** A comboBottom szelekciójának megváltoztatásakor elsütődő event. Ez ebben a dialógus ablakban nincs használva.
- **selectAction(e: ActionEvent):** A bSelect gomb lenyomásakor elsütődő esemény.

▪ PlaceDialog

• Felelősség

Ez az osztály a place gomb által létrehozott dialógus ablakot valósítja meg.

• Ősosztályok

JDialog->DialogSelect2->PlaceDialog

• Interfészek

-

• Attribútumok

- **game: Game:** Egy referencia a játék osztályra, innen tudjuk updatelni a dialógusablak megfelelő comboboxainak listáit ha változik egy másik combobox szelekciója. Például ha a felső comboboxban a settler id-ja meg lett változtatva, akkor az alsó comboboxban updatelni kell a stargatek listáját, úgy hogy azok szerepeljenek a listában amelyek a settler inventoryjában benne vannak.
- **comboTop:** JComboBox: A legfelső combobox a dialógusablakon. A settler id-jét lehet benne kiválasztani amivel le akarunk helyezni egy stargatet.
- **comboBottom:** JComboBox: A legalsó combobox a dialógusablakon. A stargate id-jét lehet benne kiválasztani amelyiket le szeretnénk helyezni.

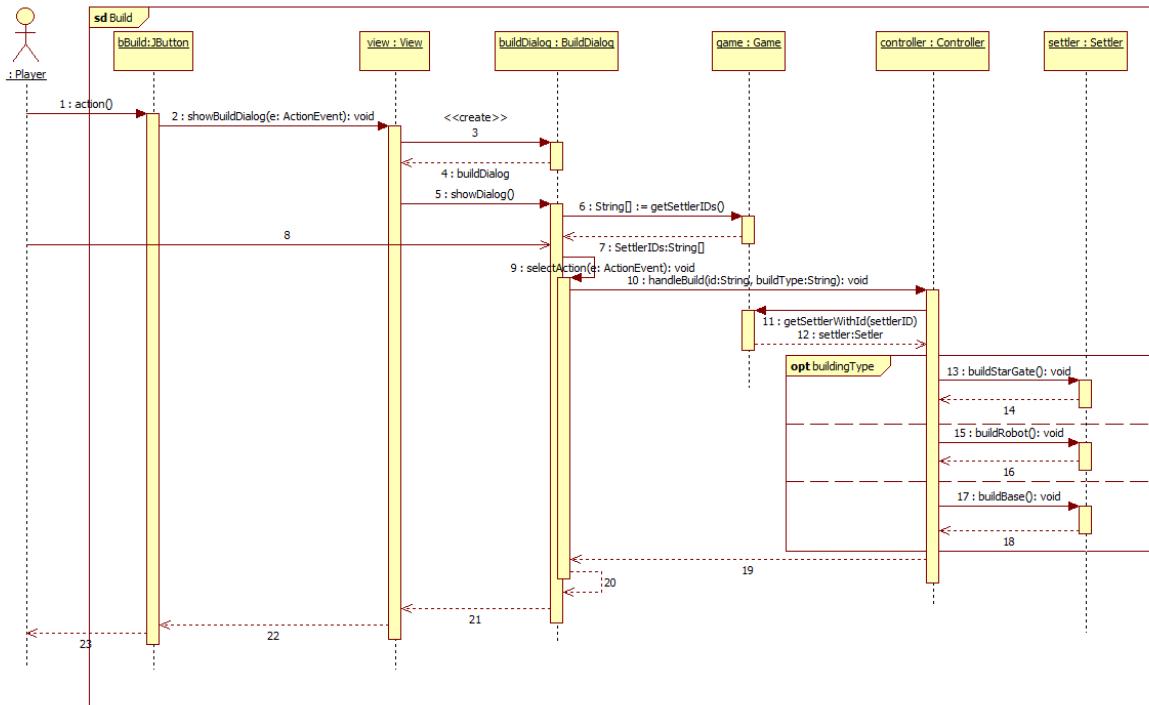
- **bSelect:** JButton: A gomb amivel leokézható a dialógusablak.
 - **Metódusok**
 - **comboTopChangedAction(e: ActionEvent):** A comboTop szelekciójának megváltoztatásakor elsütődő event. Ha a settler id-je megváltozik akkor az alsó comboboxot updatelni kell úgy, hogy a megfelelő id-jű settler inventoryjában található stargatek id-jei szerepeljenek benne.
 - **comboBottomChangedAction(e: ActionEvent):** A comboBottom szelekciójának megváltoztatásakor elsütődő event. Ez ebben a dialógus ablakban nincs használva.
 - **selectAction(e: ActionEvent):** A bSelect gomb lenyomásakor elsütődő esemény.
-
- **View**
 - **Felelősség**

A UI megjelenítéséért felelő osztály. Mostmár a modellünkben nincs külön UI osztály, szét lett szedve a Controller s a View osztályokra.
 - **Ősosztályok**
 -
 - **Interfészek**
 -
 - **Attribútumok**
 - **mainWindow: JFrame:** A játék fő ablaka.
 - **pTurnEvents: JPanel:** Egy szövegdoboz amelyik az adott körben történt dolgokat tartalmazza.
 - **pGraphicView: JPanel:** Egy megjelenítő amelyik az éppen bindolt settler/asteroid képét mutatja.
 - **pProperties: JPanel:** Egy szövegdoboz amelyik az éppen bindolt settler/asteroid tulajdonságait listázza.
 - **bNextTurn: JButton:** Egy gomb melynek hatására elindul a következő kör.
 - **bNew: JButton:** Egy gomb melynek hatására új játék kezdődik.
 - **bLoad: JButton:** Egy gomb melynek hatására betöltődik a játék állása a game.txt fileból.
 - **bSave: JButton:** Egy gomb melynek hatására elmentődik a játék állása a game.txt fileba.
 - **bExit: JButton:** Egy gomb melynek hatására kilépünk a játékból.
 - **bMove: JButton:** Egy gomb melynek hatására felugrik egy párbeszédablak melyben specifikálhatjuk, hogy melyik settler hogyan mozogjon.
 - **bDrill: JButton:** Egy gomb melynek hatására felugrik egy párbeszédablak melyben specifikálhatjuk, hogy melyik settler fúrjon.
 - **bMine: JButton:** Egy gomb melynek hatására felugrik egy párbeszédablak melyben specifikálhatjuk, hogy melyik settler bányásszon.
 - **bBuild: JButton:** Egy gomb melynek hatására felugrik egy párbeszédablak melyben specifikálhatjuk, hogy melyik settler s mit építsen.

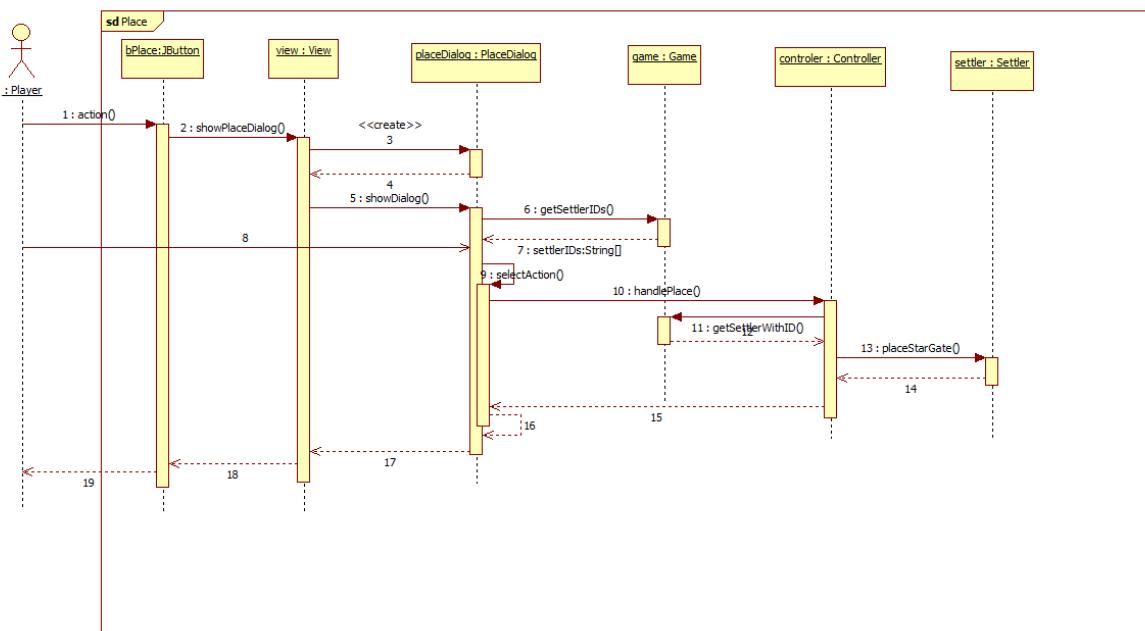
- **bPlace: JButton:** Egy gomb melynek hatására felugrik egy párbeszédablak melyben specifikálhatjuk, hogy melyik settler s melyik stargatejét helyezze le.
 - **bBind: JButton:** Egy gomb melynek hatására felugrik egy párbeszédablak melyben specifikálhatjuk, hogy mit (settler vagy aszteroidát) s abból melyiket bindoljuk.
 - **containerL: JPanel:** Egy container amiben a pTurnEvents, bNextTurn, bNew, bLoad, bSave, bExit elemek találhatók.
 - **containerM: JPanel:** Egy container amiben a pGraphicView, bMove, bDrill, bMine, bBuild, bPlace elemek találhatók.
 - **containerR: JPanel:** Egy container amiben a pProperties, bBind elemek találhatók.
- **Metódusok**
 - **showBindDialog(e: ActionEvent):** Megjeleníti a bBind által használt dialógusablakot.
 - **showMoveDialog(e: ActionEvent):** Megjeleníti a bMove által használt dialógusablakot.
 - **showBuildDialog(e: ActionEvent):** Megjeleníti a bBuild által használt dialógusablakot.
 - **showPlaceDialog(e: ActionEvent):** Megjeleníti a bPlace által használt dialógusablakot.
 - **showDropDialog(e: ActionEvent):** Megjeleníti a bDrop által használt dialógusablakot.
 - **showDrillDialog(e: ActionEvent):** Megjeleníti a bDrill által használt dialógusablakot.
 - **showMineDialog(e: ActionEvent):** Megjeleníti a bMine által használt dialógusablakot.

○ **Kapcsolat az alkalmazói rendszerrel**

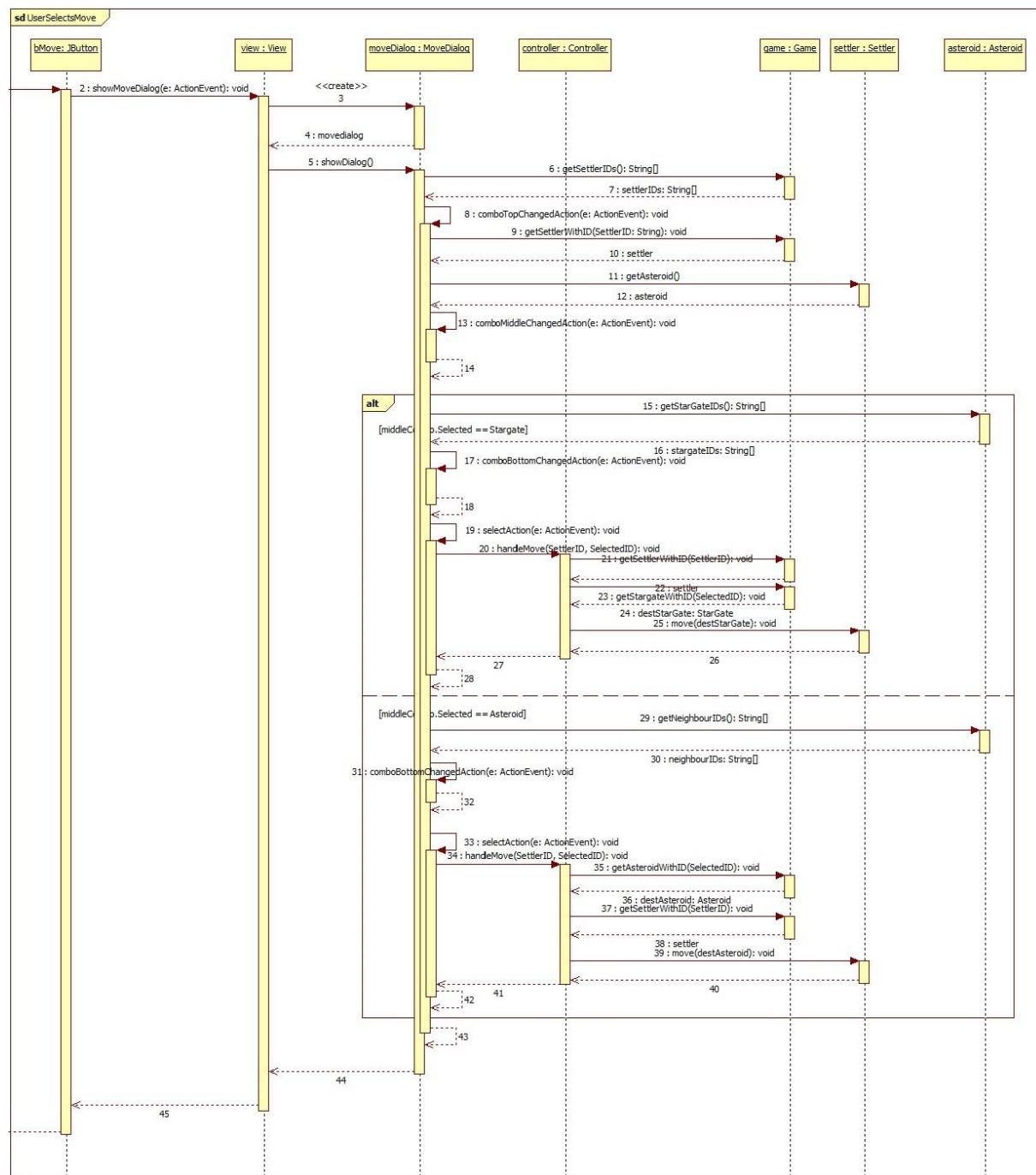
▪ Build



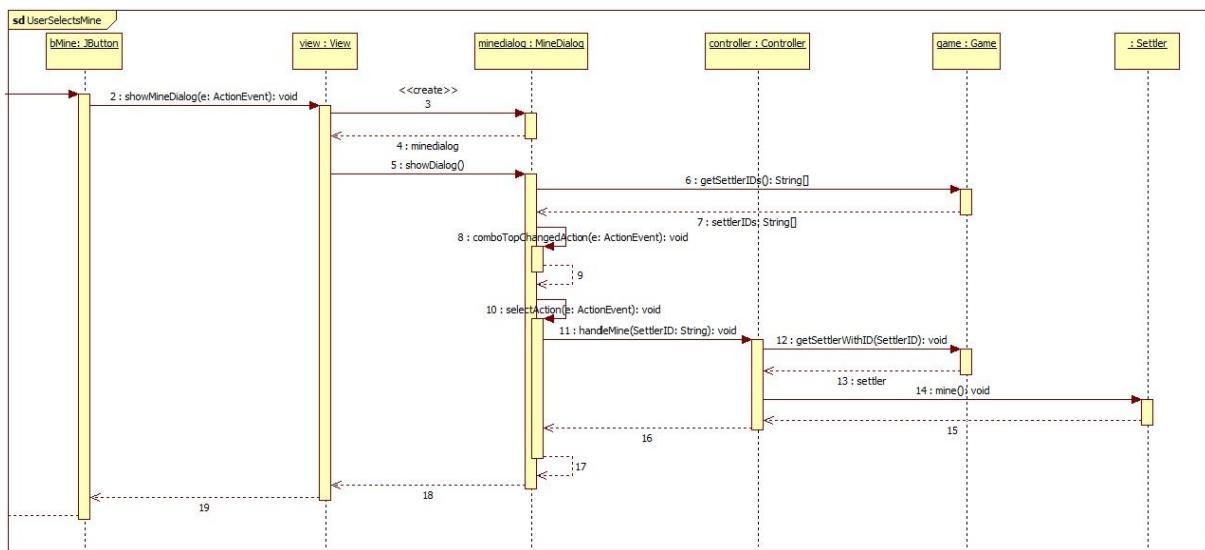
• Place



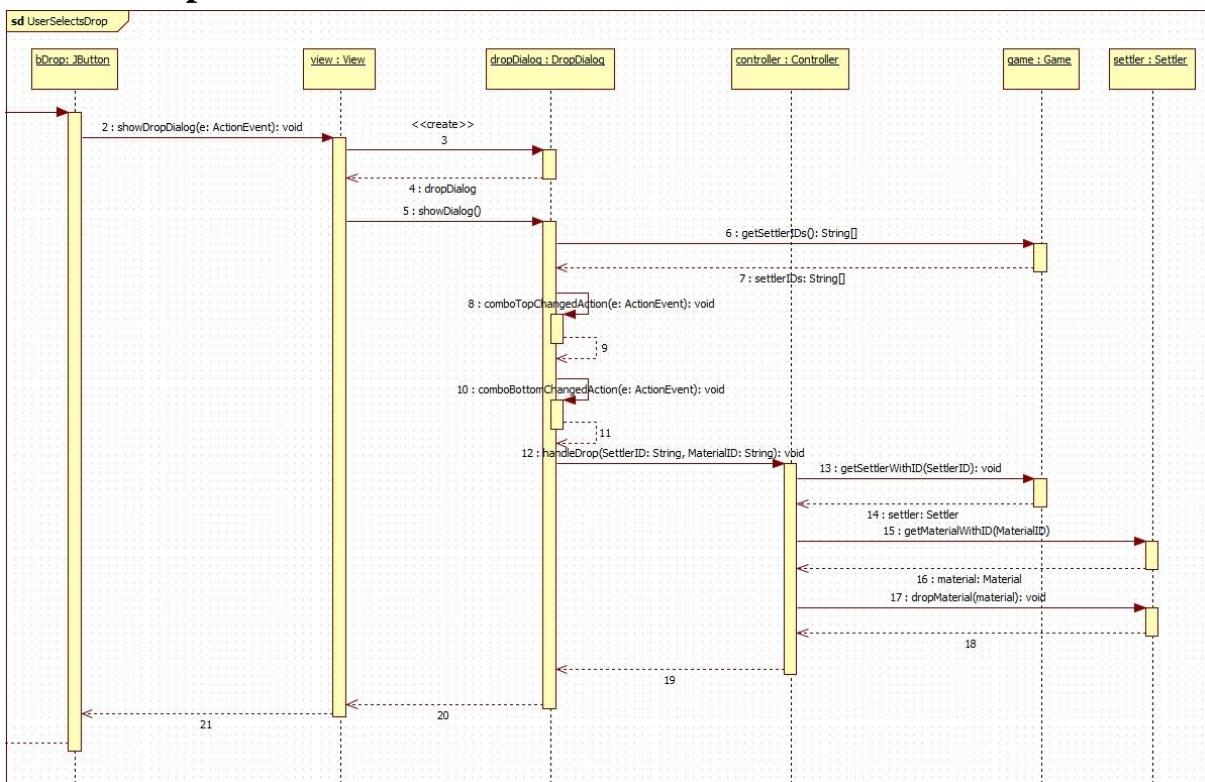
- Moves



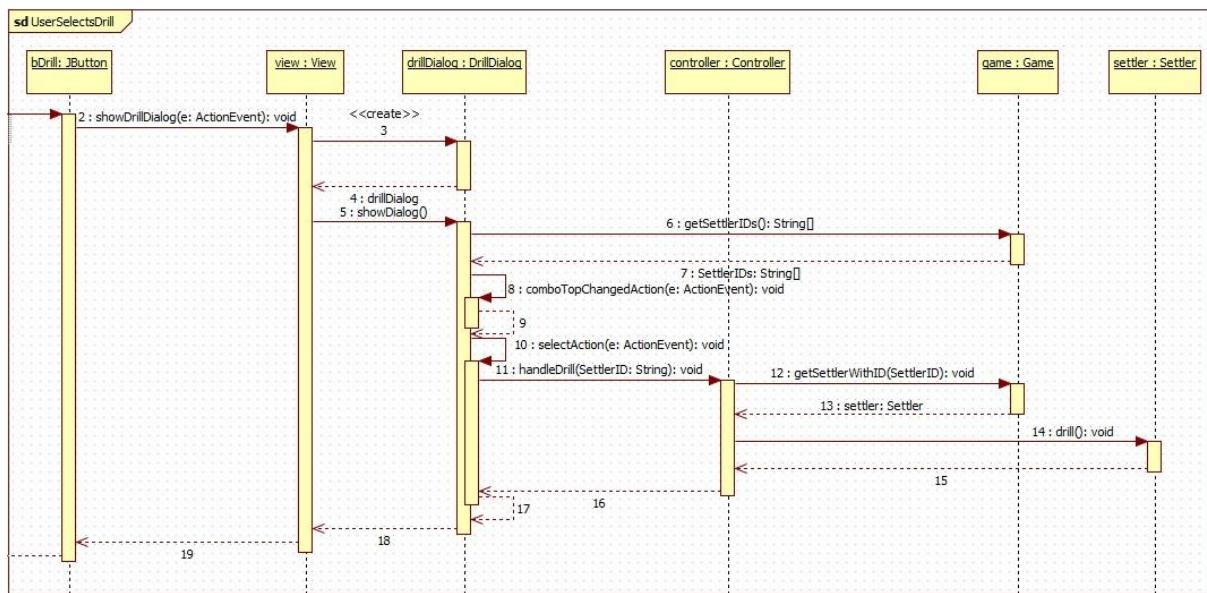
● Mine



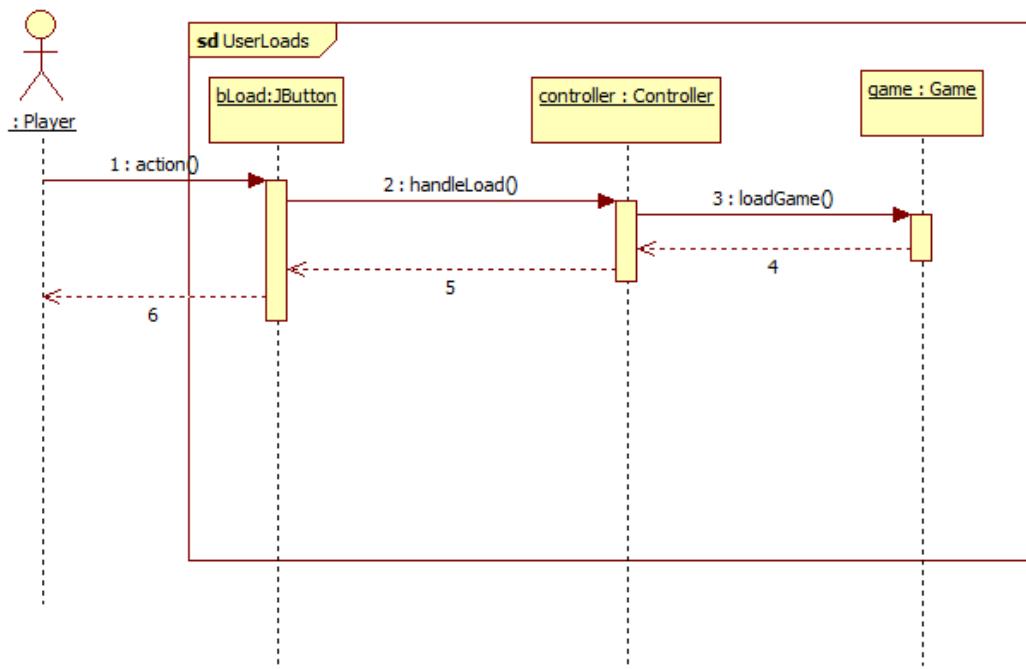
● Drop



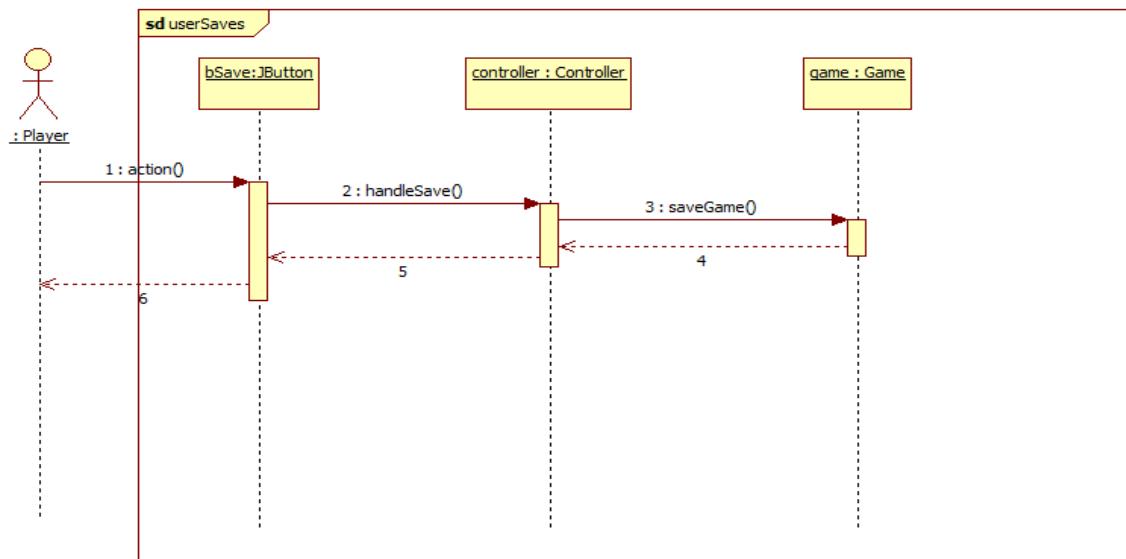
- **Drill**



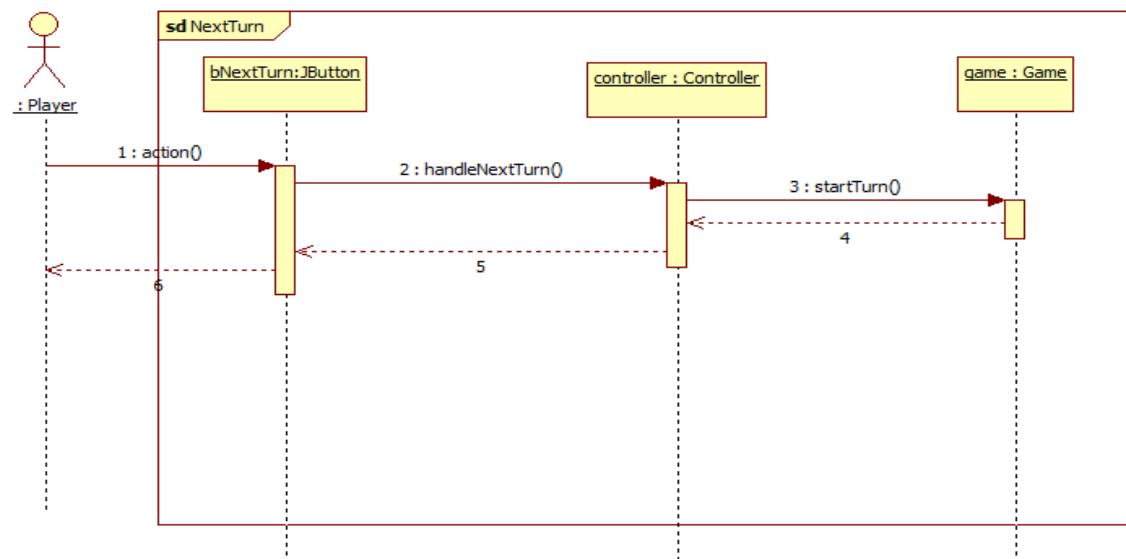
- **Load**



- Save



- Next turn



- **Napló**

Kezdet	Időtartam	Résztvevők	Leírás
2010.04.20. 20:00	1 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Értekezlet. Átnéztük az új feladatokat és összeszedtük a kérdéseinket a laboralkalomra.
2010.04.23. 16:00	1,5 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Részletesebben átbeszélgettük mit kell csinálnunk és szétosztottuk a feladatokat. Bárkányi: Grafikus interfész, Simon: Grafikus rendszer architektúrája, Kárpáti és Tóth: Kapcsolat az alkalmazói rendszerrel, Mohácsi: Osztályleírások
2010.04.24. 16:00	3 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Felvetések megvitatása. Programhibák kijavítása.
2010.04.25. 17:00	2,5 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Felvetések megvitatása. További hibák javítása.
2010.04.26. 10:00	3 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Utolsó felvetések megoldása és dokumentum összeállítása.

13. Grafikus változat beadása

67 – brainstormers

Konzulens:
Simon Balázs

Csapattagok

Mohácsi Márton

Simon Zoltán

Kárpáti Márk András

Bárkányi Csaba

Tóth Ádám

T8ZT88

HRSNUG

O1BG0Z

BQI7QQ

QEYOW2

moha.mohacsi@gmail.com

simonzolika1999@gmail.com

mark.karpati1999@gmail.com

csaba.barkanyi.csaba@gmail.com

toth99adam@gmail.com

2021-05-10

6. Grafikus változat beadása

6.1 Fordítási és futtatási útmutató

6.1.1 Fájllista

Fájl neve	Méret	Keletkezés ideje	Tartalom
Alien.java	3,277 byte	2021/04/16	Az Alien osztályt tartalmazza
Asteroid.java	16,140 byte	2021/04/12	Az Asteroid osztályt tartalmazza
Base.java	1,264 byte	2021/04/12	A Base osztályt tartalmazza
BindDialog.java	4,540 byte	2021/04/12	A BindDialog osztályt tartalmazza.
BuildDialog.java	3,036 byte	2021/05/02	A BuildDialog osztályt tartalmazza.
Building.java	1,330 byte	2021/05/02	A Building osztályt tartalmazza
Coal.java	609 byte	2021/04/12	A Coal osztályt tartalmazza
Controllable.java	369 byte	2021/04/12	A Controllable interfacet tartalmazza
Controller.java	2,486 byte	2021/05/02	A Controller osztályt tartalmazza.
DialogSelect1.java	3,270 byte	2021/05/06	A DialogSelect1 osztályt tartalmazza.
DialogSelect2.java	3,640 byte	2021/05/06	A DialogSelect2 osztályt tartalmazza.
DialogSelect3.java	3,882 byte	2021/05/06	A DialogSelect3 osztályt tartalmazza.
DrillDialog.java	2,608 byte	2021/05/06	A DrillDialog osztályt tartalmazza.
Drilling.java	81 byte	2021/04/16	A Drilling interfacet tartalmazza
DropDialog.java	4,200 byte	2021/05/06	A DropDialog osztályt tartalmazza.
Entity.java	2,246 byte	2021/04/12	Az Entity osztályt tartalmazza
Game.java	41,496 byte	2021/04/12	A Game osztályt tartalmazza
Ice.java	996 byte	2021/04/12	Az Ice osztályt tartalmazza
Inventory.java	9,147 byte	2021/04/12	Az Inventory osztályt tartalmazza
Iron.java	609 byte	2021/04/12	Az Iron osztályt tartalmazza
Main.java	2,007 byte	2021/04/12	A Main osztályt tartalmazza
Material.java	3,642 byte	2021/04/12	A Material osztályt tartalmazza
MineDialog.java	2,421 byte	2021/05/06	A MineDialog osztályt tartalmazza.
Minig.java	78 byte	2021/04/16	A Mining interfacet tartalmazza

MoveDialog.java	5,773 byte	2021/05/06	A MoveDialog osztályt tartalmazza.
Moving.java	134 byte	2021/04/16	A Moving interfacet tartalmazza
NonPlayer.java	1,889 byte	2021/04/16	A NonPlayer osztályt tartalmazza
PlaceDialog.java	4,325	2021/05/06	A PlaceDialog osztályt tartalmazza.
Planet.java	693 byte	2021/04/16	A Planet osztályt tartalmazza
Printable.java	1,144 byte	2021/04/14	A Printable osztályt tartalmazza
ReactsToSunFlare.java	102 byte	2021/04/6	A ReactsToSunFlare interfacet tartalmazza
Robot.java	3,111 byte	2021/04/12	A Robot osztályt tartalmazza
Settler.java	11,819 byte	2021/04/12	A Settler osztályt tartalmazza
StarGate.java	5,018 byte	2021/04/12	A StarGate osztályt tartalmazza
Sun.java	4,324 byte	2021/04/12	A Sun osztályt tartalmazza
Tools.java	210 byte	2021/04/12	A Tools osztályt tartalmazza
UI.java	15,740 byte	2021/04/12	A UI osztályt tartalmazza
Uran.java	1,062 byte	2021/04/12	Az Uran osztályt tartalmazza
View.java	23,019 byte	2021/04/30	A View osztályt tartalmazza.
FinalBuildScript.bat	87 byte	2021/04/19	Lefordítja a forrás forrásfájlokat. Létrehozza a Final.jar-t.
FinalRunScript.bat	19 byte	2021/04/19	Futtatja a Final.jar-t.

6.1.2 Fordítás és telepítés

A fordításhoz szükséges Java Development Kit (JDK). Csomagoljuk ki a Brainstormers_final.zip-et. Futtassul a FinalBuildScript.bat fájlt. Ez létrehoz egy bin könyvtárat a root könyvtárban és a src mappa tartalmát lefordítva a bin/-be írja. (Ehhez a binben létrejön egy main alkönyvtár is.) Végül a lefordított classokat egy Final.jar-ba csomagolja. A Main osztályt határozza meg mint főosztály.

6.1.3 Futtatás

A futtastáshoz a FinalRunScript.bat fájlt futtassuk!

Ez a korábban létrehozott Final.jar csomagot futtatja.

A megnyílt parancssorban használhatjuk az elindult programot.

A New gombra kattintva tudunk új játékot kezdeni. Ekkor automatikusan létrejön az új pályánk a rajta lévő telepesekkel, akiket a játék során irányítani tudunk.

A Bind gombra kattintva megjeleníthetjük egy telepes, vagy egy aszteroida tulajdonságait. Ezek a tulajdonságok a Properties ablakban jelennek meg.

Az egyes lépésekhez szükséges paramétereket dialógusablakokban adhatjuk meg. minden egyes lehetséges művelethez tartozik egy-egy dialógusablak, amit a neki megfelelő gombra kattintva érhetünk el.

Az egyes telepeseket mozgatni a Move gombra kattintva tudjuk. A gomb megnyomása után előugrik egy dialógusablak, ami bekéri, hogy melyik telepessel szeretnénk mozogni. Tudunk mozogni szomszédos aszteroidára, valamint teleportálhatunk az aszteroidán lévő teleportkapuval is. Előbbi esetben a Destination Type mezőnek „Asteroid” értékűnek kell lennie, utóbbi esetben, pedig „StarGate”-nek. Ezután megmondhatjuk hogy pontosan, melyik szomszédos aszteroidára szeretnénk mozogni, vagy melyik a jelenlegi aszteroidán lévő teleportkaput szeretnénk használni.

A telepesekkel fúrni a drill, Bányászni, pedig a Mine gombokkal tudunk. Mindkét esetben a felugró párbeszédablakon kell megadnunk, hogy melyik telepessel szeretnénk fúrni, vagy bányászni.

A Drop gombra kattintva kiválaszthatjuk, hogy melyik telepesnek, melyik nyersanyagát szeretnénk eldobni.

A Build gombra megjelenő dialógusablakon választhatjuk ki, hogy melyik telepessel szeretnénk építeni és hogy mit szeretnénk építeni.

A Place gombbal kiválaszthatjuk melyik telepessel szeretnénk kaput lehelyezni és hogy melyik nála lévő kaput szeretnénk lehelyezni.

A Next Turn gomb megnyomására kiíródnak a kör eseményei a Turn events dobozba és ekkor új kör kezdődik, ismét lehet minden telepessel lépni.

A Save gombbal elmenthetjük játékunkat, a Load gombbal, pedig visszatölthetjük annak állapotát.

6.2 Értékelés

Tag neve	Tag neptun	Munka százalékban
Mohácsi Márton	T8ZT88	20%
Simon Zoltán	HRSNUG	20%
Kárpáti Márk András	O1BG0Z	20%
Bárkányi Csaba	BQI7QQ	20%
Tóth Ádám	QEYOW2	20%

6.3 Napló

Kezdet	Időtartam	Résznevők	Leírás
2021.05.16:00	1 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Értekezlet a felhasználi felület megvalósításáról.
2021.05.04. 16:00	2 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Értekezlet a felhasználi felület osztályairól, ezek felosztásáról. Döntés: Bárkányi készíti el a View osztályt, Tóth

			a dialógusablakokat és Mohácsi a Controller-t írja át.
2021.05.06. 16:00	2 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Tesztelés, hibák keresése.
2021.05.08. 16:00	2 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Tesztelés, hibák keresése.
2021.05.09. 18:00	1 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Tesztelés, hibakeresés.
2021.05.10. 10:00	2 óra	Mohácsi Simon Kárpáti Bárkányi Tóth	Dokumentumok készítése.

14. Összefoglalás

67 – brainstormers

Konzulens:
Simon Balázs

Csapattagok

Mohácsi Márton

Simon Zoltán

Kárpáti Márk András

Bárkányi Csaba

Tóth Ádám

T8ZT88

HRSNUG

O1BG0Z

BQI7QQ

QEYOW2

moha.mohacsi@gmail.com

simonzolika1999@gmail.com

mark.karpati1999@gmail.com

csaba.barkanyi.csaba@gmail.com

toth99adam@gmail.com

2021-05-10

7. Összefoglalás

7.1 A projektre fordított összes munkaidő

Tag neve	Munkaidő (óra)
Mohácsi Márton	125
Simon Zoltán	125
Kárpáti Márk András	125
Bárkányi Csaba	125
Tóth Ádám	125
Összesen	125

- A feltöltött programok forrássorainak száma**

Fázis	KódSOROK száma
Szkeleton	3589
Prototípus	4899
Grafikus változat	7018
Összesen	15506

7.2 •Projekt összegzés

7.2.1 Mit tanultak a projektből konkrétan és általában?

A feladat során konkrétan képet kaptunk arról, hogy hogyan is néz ki egy csoportos project és ezen belül milyen elemi lépésekkel állhat. Lehetőségünk volt ezen kívül például az UML adta lehetőségek beépítésének módjainak a megismerésére a projekt életciklusa során.

Általánosságban véve megtanultunk a csaptatásainkkal együtt dolgozni és összehangoltan elvégezni a munkákat és a részfeladatokat az egyes személyes képességei és érdeklődési köre szerint szétosztani.

7.2.2 Mi volt a legnehezebb és a legkönnyebb?

Legnehezebbnek a feladat elején, amikor még csak a konkrét tervezés zajlott, viszont pont emiatt, hogy az nehéz volt és több időt szántunk rá a konkrét implementáció többek között zökkenőmentesen tudott zajlani. A legkönnyebb így a már megtervezett program implementálása volt.

7.2.3 Összhangban állt-e az idő és a pontszám az elvégzendő feladatokkal?

Szerintünk teljes mértékben összhangban áll.

7.2.4 Ha nem, akkor hol okozott ez nehézséget?

Nem okozott nehézséget.

7.2.5 Milyen változtatási javaslatuk van?

Nincs változtatási javaslatunk.

7.2.6 Milyen feladatot ajánlanának a projektre?

Egy olyan dungeon alapú játék tervezése, amelyben az egyes szintek a történelem és/vagy a jövő bizonyos helyen játszódnának.

7.2.7 Egyéb kritika és javaslat

Nincs kritikánk vagy javaslatunk.