

## 8. Részletes tervek

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

Változtatások .....	4
Bemeneti nyelv .....	4
Játék parancsai: .....	6
Kimeneti nyelv .....	9
8. Részletes tervek .....	12
8.1 Osztályok és metódusok tervei. ....	12
8.1.1 Alien .....	12
8.1.2 Asteroid .....	13
8.1.3 Base .....	15
8.1.4 Building .....	16
8.1.5 Coal .....	17
8.1.6 Drilling .....	18
8.1.7 Entity .....	18
8.1.8 Game .....	19
8.1.9 Ice .....	21
8.1.10 Inventory .....	21
8.1.11 Iron .....	22
8.1.12 Material .....	23
8.1.13 Mining .....	23
8.1.14 Moving .....	24
8.1.15 NonPlayer .....	24
8.1.16 Planet .....	26
8.1.17 Printable .....	27
8.1.18 ReactsToSunflare .....	27
8.1.19 Robot .....	28
8.1.20 Settler .....	29
8.1.21 StarGate .....	30
8.1.22 Sun .....	32
8.1.23 UI .....	33
8.1.24 Uran .....	33
8.2 A tesztek részletes tervei, leírásuk a teszt nyelven .....	34
8.2.1 Bányászás telepessel .....	34
8.2.2 Bányászás telepessel, teli tárolóval .....	35
8.2.3 Telepes visszahelyez egy nyersanyagot .....	37
8.2.4 Telepes nem tud visszahelyezni egy nyersanyagot .....	38
8.2.5 Telepes visszahelyez egy uránt napközelben .....	39
8.2.6 Telepes visszahelyez egy vízjeget napközelben .....	40
8.2.7 Alien bányászik .....	41
8.2.8 Alien bányászna .....	42
8.1.1 Nem elvégezhető Robot Crafting .....	42
8.1.2 Elvégezhető Robot Crafting .....	44
8.1.3 Nem elvégezhető Stargate Crafting .....	45
8.1.4 Elvégezhető Stargate Crafting .....	47
8.1.5 Nem elvégezhető Base Crafting egy telepessel .....	48
8.1.6 Nem elvégezhető Base Crafting telepesek külön aszteroidán .....	49
8.1.7 Elvégezhető Base Crafting .....	52
8.1.8 Első Stargate lehelyezése a párból .....	54

8.1.9	Második Stargate lehelyezése a párból .....	56
8.2	Napvihar kitörése nem átfűrt aszteroidán .....	58
8.3	Napvihar kitörése átfűrt aszteroidán .....	60
8.4	Fúrás robottal .....	63
8.5	Fúrás robottal, elpárolog .....	64
8.6	Fúrás robottal, felrobban .....	66
8.7	Fúrás robottal, felrobban, nincs szomszéd .....	67
8.8	Fúrás telepessel .....	69
8.9	Fúrás telepessel, elpárolog .....	71
8.10	Fúrás telepessel, felrobban .....	72
8.11	Mozgás robottal, kudarc .....	73
8.12	Mozgás robottal, siker .....	75
8.13	Mozgás telepessel, kudarc .....	77
8.14	Mozgás telepessel, siker .....	78
8.15	Telepes átlép egy kapun. ....	80
8.16	Teleportkapu elromlik. ....	81
8.3	A tesztelést támogató programok tervei .....	83
8.4	Napló .....	84

## Változtatások

A game osztályban már nem külön planetek meg robotok vannak egy-egy listában hanem ömlesztve controllablek.

A game osztály destroy(s: Settler) függvényét a formaiság miatt átneveztük remove(s: Settler)-é.

A buildingeknek illetve planeteknek mostmár van ownerje.

A moving interfésznek mostmár nem csak move(destination: Asteroid) hanem move(destination: StarGate) függvénye is van.

Az aszteroid neighbours listája mostantól csak azokat az aszteroidákat tartalmazza melyekkel eredetileg is szomszédságban volt ez az aszteroida (minusz amik elpusztultak).

Az aszteroidában lévő nyersanyag maximális mennyiségét a capacity tagváltozó jelöli mely random értékkel lesz inicializálva.

A Drilling függvényénél csak akkor hívódik meg az exposedAndCloseToSun() ha a kéreg éppen 0-ra változik, ellenkező esetben egy 0 vastagságú aszteroida fúrásánál is öregedne az aszteroidában található urán.

A stargate működését mostmár (a mozgás miatt) lehet állítani a determinisztikus s a random között.

A tesztesetek fejlesztése során változtatásokat eszközöltünk a nyelven. A változtatások mennyisége miatt érdemesnek találtuk rögzíteni a végleges nyelvet a dokumentum elején, hogy a módosítások egyértelműek legyenek.

## Bemeneti nyelv

### Inicializálás:

*init game <manually, random>*

**Leírás:** Játék létrehozása.

**manually:** Alapértelmezésben mindent 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 (ekkor a new value t/f)

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

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

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

**property name:** egy aszteroida tulajdonsága, lehetséges értékei:

closetosun (ekkor 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 telepet 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 szomszédot ad hozzá az aszteroidához.

neighbour id: A szomszéd aszteroida azonosítója.

*add asteroid building <asteroid id> <building id>*

Leírás: Egy szomszédot 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 telepest 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 teleppel 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 telepés azonsítója.

*drop settler <id> material <material id>*

**Leírás:** Az adott telepés eldobja a megjelölt nyersanyagot.

**Opciók:**

**-id:** A telepés azonsítója.

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

*build settler <id> <obj.>*

**Leírás:** Az adott telepés felépíti a kijelölt objektumot az aszteroidáján.

**Opciók:**

**-id:** A telepés azonsítója.

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

*place gate settler <id> <gate id>*

**Leírás:** Az adott telepés lehelyezi a megadott kaput a jelenlegi aszteroidájára.

**Opciók:**

**-id:** A telepés 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 alien 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 ellentettére 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:**

&lt;material id&gt;

&lt;material id&gt;

...

**stargates**

&lt;stargate id&gt;

&lt;stargate id&gt;

...

;

**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:**

&lt;neighbour id&gt;



```

...
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.

**Kimeneti nyelv*****list turn events***

**Leírás:** A parancs hatására kiíródik egy rövid összefoglalás az elmúlt kör eseményeiről. Ennek tartalma a következő bejegyzésekből 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 objektumon az alább leírt list info parancsokat.  
 Először a napokon, aztán aszteroidákon iterál végig, majd következnek az épületek, a telepések, 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>

## 8. Részletes tervek

### 8.1 Osztályok és metódusok tervei.

#### 8.1.1 Alien

##### 8.1.1.1 Felelősség

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

- **-id: int:** Ez az adott Alien id-je, ezt arra használjuk fel, hogy az Alien-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.
- **-asteroid:Asteroid:** Ezen az aszteroidán van az Alien.
- **-owner:Game:** Ez az osztály tárolja le az Alien-t
- **-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 asteroida-ja nem igaz értékkel tér vissza az isHollow függvényre
- **+move(destination: Asteroid):** egy függvény, aszteroidáról aszteroidára való mozgást végzi.

```
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 kivethető 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 kiválasztót szomszédjával a move függvényét. Majd ezt ismételteti.
- **+mine():** egy függvény, amely kivesz egy Material-t az aszteroidából, ha az aszteroida kérge 0.Osztály2

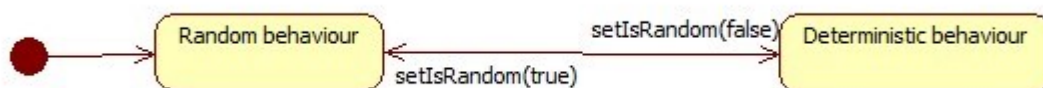
## 8.1.2 Asteroid

### 8.1.2.1 Felelősség

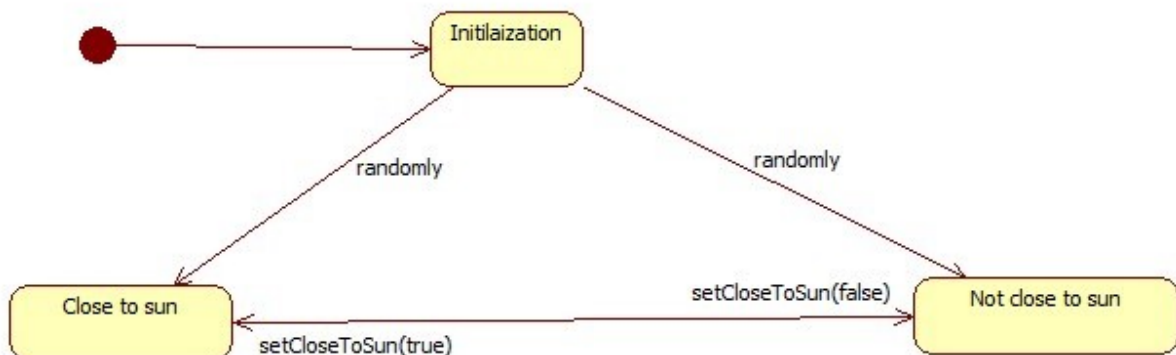
Ez az osztály felel a játékban található aszteroidák feladataiért, pl.: amikor napkitörés van ez az osztály hívja meg a rajta található entitásokon s épületeken a getNotifiedAboutSunflare() függvényt, ezzel jelezve nekik hogy épp napkitörés van.

### 8.1.2.2 Állapotdiagrammok

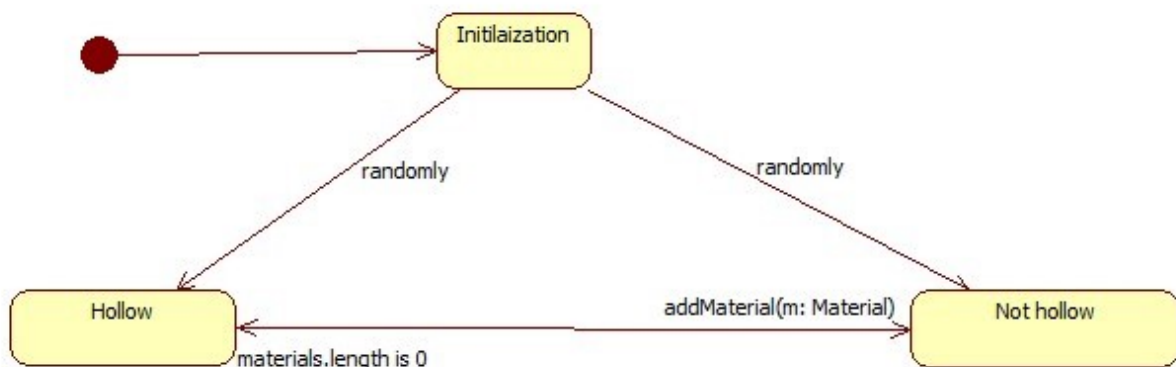
- Viselkedés



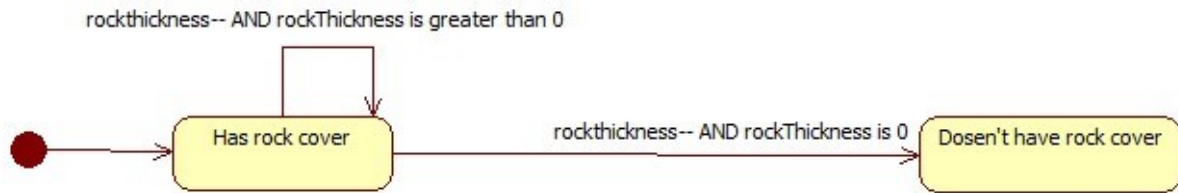
- Napközelség



- Üreges



- Kifűrt



(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 kérgének 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 (napközelségbe kerülés esély számlálás közben), ez csak a teszteléshez kell.
- **-capacity: int:** Ez az int azt jelzi, hogy az aszteroidába mennyi darab nyersanyag fér bele.

### 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 napközelségét (ha napközelen van akkor kikerül-e onnan s ha nincs napközelen 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 materialon (ha van) meghívja az exposedAndCloseToSun() függvényt.
- **+getNotifiedAboutSunflare():** Ez a függvény akkor hívódik meg az aszteroidán amikor azt napvihar éri, ilyenkor az aszteroida meghívja a getNotifiedAboutSunflare() függvényt az összes épületen, telepesen s nonPlayeren ami rajta van.
- **+addSettler(s: Settler):** Hozzáadja a settlers listához az s Settlet.
- **+addNonPlayer(np: NonPlayer):** Hozzáadja a nonPlayers listához az np NonPlayert.
- **+removeSettler(s: Settler):** Kivesszük a settlers listából az s Settlet.
- **+removeNonPlayer(np: NonPlayer):** Kivesszük 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 fúr az aszteroidán, az aszteroida kérge csökken egyel (ha tud), illetve ha épp 0-ra csökken akkor meghívja az aszteroidában található nyersanyagok egyikén az exposedAndCloseToSun() függvényt.
- **+removeMaterial(): Material:** Ez a függvény eltávolít egy nyersanyagot az aszteroida belsejéből (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):** Kivesszük a neighbours listából az a Asteroidot.
- **+addNeighbour(a: Asteroid):** Hozzáadja a neighbours listához az a Asteroidot.
- **+removeBuilding(b: Building):** Kivesszük 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 elpárologtatja a belsejében található nyersanyagokat. Tehát kivesszi 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 napközelen 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 napvihar 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

### 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.
- 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.
- **+onPlace(a: Asteroid):** Ez a függvény végrehajtja az épület feladatait abban a pillanatban amikor ráhelyezzük 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

- **-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 napközbe kerül
- **-exposedCounter:int:** Ez tárolja számon, hogy milyen hosszan volt kitéve a napnak az adott nyersanyag.

#### 8.1.5.5 Metódusok

- **+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.
- **+exposedAndCloseToSun(a:Asteroid):** Ez a függvény hívódik, amikor napközben 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 interfészt 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 fúrássra 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 s 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 játékosal 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 Settler-t.
- **void removeControllable(c: Controllable):** Kiveszi a controllables listából az s Controllable-t.
- **void addSettler(s: Settler):** Hozzáadja a settlers listához az s Settler-t.
- **void addControllable(c: Controllable):** Hozzáadja a controllables listához a c Controllable-t.
- **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 elvesztve amikor az összes settler meghalt, tehát 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 telepéseket s űrlényeket s ezeket ráhelyezi random módon az aszteroidákra. Az aszteroidát ezek után még inicializálja is az initialize() függvényével, ez azért kell mert az aszteroidáknak pl. többféle nyersanyag lehet a belsejében, s ezek eldöntéséért 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 visszafejtése alapján fog működni.
- **void executeCommand(arguments: String):** Ez a függvény végrehajt egy Controllable objektummal egy parancsot. Az id-vel specifikált controllable objektumon végrehajtjuk a megfelelő parancsot.
- **void startGame():** Ez a függvény elindítja a játékot. Ez jelzi az inicializálás befejeztét. Mivel a settlerek mindig a körben 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 információit 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 napközbe 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 napközben 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 típusu objektumokat tárol. Ebben van számontartva, 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 térési értéként.
- **StarGate removeStarGate():** kivesz egy StarGate-t az inventory-ból és azt adja vissza mint visszatérési érték.
- **Inventory subSet(i: Inventory):** ezt a függvényét 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 adott inventory-ból a releváns nyersanyagokat.
- **void addStarGate(sg:StarGate):** hozzá ad egy StarGate-t az inventory-hoz.
- **void addMaterial(m:Material):** hozzá adja a megadott Material-t az Inventory-hoz.

### 8.1.11 Iron

#### 8.1.11.1 Felelősség

Ez az osztály felel a vas viselkedésért.

#### 8.1.11.2 Ősosztályok

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 napközbe 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 hívodik, amikor napközben 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

- **-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 napközbe kerül
- **-exposedCounter:int:** Ez tárolja számon, hogy milyen hosszan volt kitéve a napnak az adott nyersanyag.

##### 8.1.12.5 Metódusok

- **+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.
- **+exposedAndCloseToSun(a:Asteroid):** Ez a függvény hívóik, amikor napközben 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 interfészt 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 interfészt 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 által irányított karakterekért



### 8.1.15.2 Ősosztályok

Printable->Entity->NonPlayer

### 8.1.15.3 Interfészek

ReactsToSunflare, Moving, Controllable

### 8.1.15.4 Attribútumok

- **-id: int:** Ez az adott NonPlayer id-je, ezt arra használjuk fel, hogy az NonPlayer-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.
- **-asteroid:Asteroid:** Ezen az aszteroidán van a NonPlayer.
- **-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 végzi.

```
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 űrosztály 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 űrosztályok

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 s 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 őszosztály 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 Őszosztályok

-

#### 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 fűr, akkor a settler id-jét megadva választhatja ki melyikkel kíván fűrni.
- **-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 interfészt azok az osztályok valósítják meg, amik reagálnak a napviharra

#### 8.1.18.2 Őszosztályok

-

#### 8.1.18.3 Interfészek

-

#### 8.1.18.4 Attribútumok

-

#### 8.1.18.5 Metódusok

- **+getNotifiedAboutSunflare():** Egy absztrakt függvény amely elvégzi azokat a függvény hívásokat, amiket a napvihar 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

- **-id: int:** Ez az adott Robot id-je, ezt arra használjuk fel, hogy az Robot-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.
- **-asteroid:Asteroid:** Ezen az aszteroidán van a Robot.
- **-owner:Game:** Ez az osztály tárolja le a Robot-t
- **-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 asteroida-ja nem igaz értékkel tér vissza az isHollow függvényre
- **+move(destination: Asteroid):** egy függvény, aszteroidáról aszteroidára való mozgást végzi.

```
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 kivethető 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 aszteroida-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 telepésért.

### 8.1.20.2 Ősosztályok

Printable->Entity->Settler

### 8.1.20.3 Interfészek

ReactsToSunflare, Moving, Drilling, Mining

### 8.1.20.4 Attribútumok

- **-id: int:** Ez az adott Settler id-je, ezt arra használjuk fel, hogy az Settler-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.
- **-asteroid:Asteroid:** Ezen az aszteroidán van a Settler.
- **-owner:Game:** Ez az osztály tárolja le a Robot-t
- **-inventory:Inventory:** Ebben tárolja a Settler általa össze gyűjtött nyersanyagokat.
- **-hadActionThisTurn:Boolean:** Ez tárolja, hogy lehet e ezzel a teleppel lépni az adott körben.
- **-baseRecipe:** ebben van benne, hogy mi kell egy bázis megépítéséhez.
- **-robotRecipe:** ebben van benne, hogy mi kell egy robot megépítéséhez.
- **-stargateRecipe:** ebben van benne, hogy mi kell egy kapu 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 s 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 végzi.

```
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 kivethető 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 tartozkodik, 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 átfűrt felszínnel), nyersanyagot helyez az aszteroidába az inventory-ból.
- **+placeStarGate():** Amennyiben van az Inventory-ben StarGate, lehelyezi azt az adott aszteroidára.

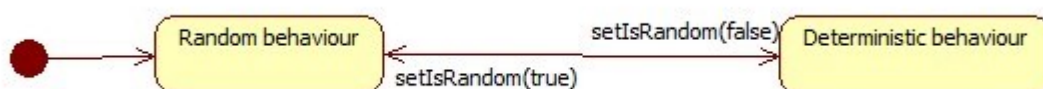
## 8.1.21 StarGate

### 8.1.21.1 Felelősség

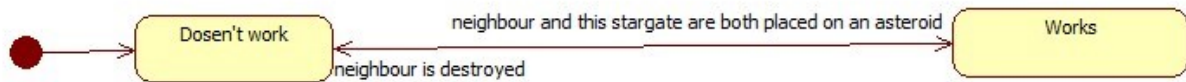
Ez az osztály felel a teleportkapukért s azok feladataiért, pl: hogy azok mindig jól össze legyenek párosítva, illetve ha a párból az egyik elpusztult akkor erről tudjon a másik (ha van rá szüksége).

### 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, Controllable. 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 napvihar.
- **-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 napvihar éri, ilyenkor a StarGateben true-ba billen a wasInSunFlare változó, innentől a StarGate elkezd mozogni.
- **+onTurn():** Egy függvény mely mozgatja a StarGatet minden körben, ha az már volt napviharban. Mozgás során a StarGate rálép az aszteroida neighbours listájában legelső helyen szereplő aszteroidára (ha van ilyen, ha nincs akkor nem mozog).
- **+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 a függvényben a StarGate rálép a destination StarGate-re s azon keresztül annak párjának 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.

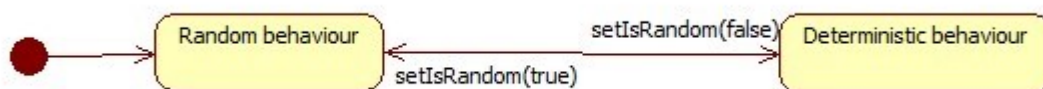
## 8.1.22 Sun

### 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 eséllyel tör ki, nem random esetben pedig a felhasználó hívja meg a sunFlare()-t amikor szeretné, 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 játékkal 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 belüli 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

- **-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 napközbe kerül.
- **-exposedCounter:int:** Ez tárolja számon, hogy milyen hosszan volt kitéve a napnak az adott nyersanyag.

#### 8.1.24.5 Metódusok

- **+getString(): 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.
- **+exposedAndCloseToSun(a:Asteroid):** Ez a függvény hívodik, amikor napközben 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 tervei, 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 mindent determinisztikus állapotra állít alapértelmezésben, ennek ellenére szemléltetésképpen egyes helyeken külön beállításra kerül.

#### 8.2.1 Bányászás telepessel

- **Leírás**

Egy telepes bányász az aszteroidájából.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Annak ellenőrzése, hogy a telepes valóban kibányássza-e a nyersanyagot egy átfűrt 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 settler S1
create inventory I1
create asteroid A1
set settler asteroid S1 A1
add asteroid settler A1 S1
set settler inventory S1 I1
create coal C1
set asteroid material A1 C1
```

```

set asteroid rockthickness A1 0
set asteroid deterministic behaviour A1 t
start game
mine settler S1
list turn events
list all

```

- **Elvárt kimenet**

```

#list turn events
material mined C1
asteroid mined A1

```

```

#list all
Asteroid id: A1
rockThickness: 0
closeToSun: f
isRandom: f
material -
neighbours:
    neighbour: -
buildings:
    building: -
settlers:
    settler: S1
...
nonPlayers:
    nonPlayer: -

Settler id: S1
hadActionThisTurn: t
asteroid: A1
inventory:
materials:
    material: C1
starGates:
    starGate:-

```

## 8.2.2 Bányászás telepessel, teli tárolóval

- **Leírás**

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

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A telepés 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 settler S1
create inventory I1
create asteroid A1
set settler asteroid S1 A1

```

```

add asteroid settler A1 S1
set settler inventory S1 I1
create coal C1
create iron IR1
create iron IR2
create iron IR3
create iron IR4
create iron IR5
create iron IR6
create iron IR7
create iron IR8
create iron IR9
create iron IR10
add inventory material I1 IR1
add inventory material I1 IR2
add inventory material I1 IR3
add inventory material I1 IR4
add inventory material I1 IR5
add inventory material I1 IR6
add inventory material I1 IR7
add inventory material I1 IR8
add inventory material I1 IR9
add inventory material I1 IR10
set asteroid material A1 C1
set asteroid rockthickness A1 0
set asteroid deterministic behaviour A1 t
start game
mine settler S1
list turn events
list all

```

- **Elvárt kimenet**

```

#list turn events
-

```

```

#list all
Asteroid id: A1
rockThickness: 0
closeToSun: f
isRandom: f
material C1
neighbours:
neighbour: -
buildings:
    building: -
settlers:
    settler: S1
nonPlayers:
    nonPlayer: -

Settler id: S1
hadActionThisTurn: t

```

```

asteroid: A1
inventory:
materials:
    material: IR1
    material: IR2
    material: IR3
    material: IR4
    material: IR5
    material: IR6
    material: IR7
    material: IR8
    material: IR9
    material: IR10
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 settler S1
create asteroid A1
create inventory I1
create coal C1
set settler asteroid S1 A1
set settler inventory S1 I1
add asteroid settler A1 S1
add inventory material I1 C1
set asteroid rockthickness A1 0
set asteroid deterministic behaviour A1 t
start game
drop settler S1 material C1
list turn events
list all

```

- **Elvárt kimenet**

```

#list turn events
material dropped A1

```

```

#list all
Asteroid id: A1
rockThickness: 0
closeToSun: f
isRandom: f
material C1
neighbours:

```

```

neighbour: -
buildings:
  building: -
settlers:
  settler: S1
nonPlayers:
  nonPlayer: -

Settler id: S1
hadActionThisTurn: t
asteroid: A1
inventory:
materials:
  material: -

```

### 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 átfűrva, 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 settler S1
create asteroid A1
create inventory I1
create coal C1
set settler asteroid S1 A1
set settler inventory S1 I1
add asteroid settler A1 S1
add inventory material I1 C1
set asteroid rockthickness A1 1
set asteroid deterministic behaviour A1 t
start game
drop settler S1 material C1
list turn events
list all

```

- **Elvárt kimenet**

```

#list turn events
-

#list all
Asteroid id: A1
rockThickness: 1
closeToSun: f
isRandom: f
material -
neighbours:
neighbour: -

```

```

buildings:
    building: -
settlers:
    settler: S1
nonPlayers:
    nonPlayer: -
Settler id: S1
hadActionThisTurn: t
asteroid: A1
inventory:
materials:
    material: C1

```

### 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 settler S1
create asteroid A1
create inventory I1
create uran U1
set settler asteroid S1 A1
set settler inventory S1 I1
add asteroid settler A1 S1
add inventory material I1 U1
set asteroid rockthickness A1 0
set asteroid closetosun t
set asteroid deterministic behaviour A1 t
start game
drop settler S1 material U1
start next turn
mine settler S1
start next turn
drop settler S1 material U1
start next turn
mine settler S1
start next turn
drop settler S1 material U1
list turn events
list all
Elvárt kimenet
#list turn events:

```

```
material dropped U1
material exploded U1
asteroid exploded A1
settler died S1
```

```
#list all:
-
```

### 8.2.6 Telepes visszahelyez egy vízjeget napközben

- **Leírás**

Egy telepes visszahelyez egy vízjeget az aszteroidájába, napközben.

- **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 settler S1
create asteroid A1
create inventory I1
create ice IC1
set settler asteroid S1 A1
set settler inventory S1 I1
add asteroid settler A1 S1
add inventory material I1 IC1
set asteroid rockthickness A1 0
set asteroid closetosun t
set asteroid deterministic behaviour A1 t
start game
drop settler S1 material IC1
```

- **Elvárt kimenet**

```
#list turn events
material dropped IC1
material evaporated IC1
```

```
#list all
Asteroid id: A1
rockThickness: 0
closeToSun: t
isRandom: f
material -
neighbours:
neighbour: -
buildings:
    building: -
settlers:
    settler: S1
nonPlayers:
    nonPlayer: -
```



```

Settler id: S1
hadActionThisTurn: t
asteroid: A1
inventory:
materials:
    material: -

```

### 8.2.7 Alien bányászik

- **Leírás**

Egy alien kibányászik egy nyersanyagot az aszteroidájából.

- **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 aszteroidájából.

- **Bemenet**

```

init game manually
create alien AL1
create asteroid A1
create coal C1
set alien asteroid AL1 A1
add asteroid nonplayer A1 AL1
add asteroid material A1 C1
set asteroid rockthickness A1 0
set asteroid deterministic behaviour A1 t
set alien deterministic behaviour AL1 t
start game
mine alien AL1
list turn events
list all

```

- **Elvárt kimenet**

```

#list turn events
material mined C1
asteroid mined A1

#list all
Asteroid id: A1
rockThickness: 0
closeToSun: f
isRandom: f
material -
neighbours:
neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: AL1

Alien id: AL1
isRandom: f

```

asteroid: A1

### 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 alien AL1
create asteroid A1
create coal C1
set alien asteroid AL1 A1
add asteroid nonplayer A1 AL1
add asteroid material A1 C1
set asteroid rockthickness A1 1
set asteroid deterministic behaviour A1 t
set alien deterministic behaviour AL1 t
start game
mine alien AL1
list turn events
list all
```

- **Elvárt kimenet**

```
#list turn events
-

#list all
Asteroid id: A1
rockThickness: 1
closeToSun: f
isRandom: f
material C1
neighbours:
neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: AL1

Alien id: AL1
isRandom: f
asteroid: A1
```

### 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 megíiusulását eredményezheti.

- **Bemenet**

```
init game manually
```

```
create asteroid a1
create settler s1
create inventory i1
create iron iron1
create coal coal1
```

```
#s1
add inventory material i1 iron1
add inventory material i1 coal1
set settler inventory s1 i1
set settler asteroid s1 a1
#a1
add asteroid settler a1 s1
```

```
#-----
#Végrehajtás:
```

```
start game
build settler s1 robot
start next turn
list turn events
list all
```

- **Elvárt kimenet**

```
#list turn events
-
```

```
#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
-
buildings:
-
settlers:
  settler: s1
```

```

nonPlayers:
    -

Settler id:s1:
hadActionThisTurn: t
asteroid: a1
inventory:
    materials:
        material: iron1
        material: coal1
    starGates:
    -

```

### 8.1.2 Elvégezhető Robot Crafting

- **Leírás**

A játékos a telepest 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 telepesnél 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 megghiúsulását eredményezheti.

- **Bemenet**

```

init game manually

create asteroid a1
create settler s1
create inventory i1
create iron iron1
create uran uran1
create coal coal1

#s1
add inventory material i1 iron1
add inventory material i1 uran1
add inventory material i1 coal1
set settler inventory s1 i1
set settler asteroid s1 a1

#a1
add asteroid settler a1 s1
#-----
#Végrehajtás:

start game
build settler s1 robot
start next turn
list turn events

```

```
list all
  • Elvárt kimenet
#list turn events
robot crafted r1

#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
  -
buildings:
  -
settlers:
  settler: s1
nonPlayers:
  nonPlayer: r1

Settler id: s1:
hadActionThisTurn: t
asteroid: a1
inventory:
  materials:
    -
  starGates:
    -

Robot id: r1
isRandom: f
asteroid: a1
```

### 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 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 megghiúsulását eredményezheti.

- **Bemenet**

```
init game manually

create asteroid a1
create settler s1
```

```

create inventory i1
create iron iron1
create uran uran1
create ice ice1

#s1
add inventory material i1 iron1
add inventory material i1 uran1
add inventory material i1 ice1
set settler inventory s1 i1
set settler asteroid s1 a1

#a1
add asteroid settler a1 s1

#-----
#Végrehajtás:

start game
build settler s1 stargate
start next turn
list turn events
list all
    • Elvárt kimenet
#list turn events
-

#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
-
buildings:
-
settlers:
    settler: s1
nonPlayers:
-

Settler id: s1:
hadActionThisTurn: t
asteroid: a1
inventory:
    materials:
        material: iron1
        material: uran1
        material: ice1
    starGates:

```

-

### 8.1.4 Elvégezhető Stargate Crafting

- **Leírás**

A játékos a telepest 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 megíúsulását eredményezheti.

- **Bemenet**

```
init game manually
```

```
create asteroid a1
create settler s1
create inventory i1
create iron iron1
create iron iron2
create uran uran1
create ice ice1
#s1
add inventory material i1 iron1
add inventory material i1 iron2
add inventory material i1 uran1
add inventory material i1 ice1
set settler inventory s1 i1
set settler asteroid s1 a1

#a1
add asteroid settler a1 s1
```

```
#-----
#Végrehajtás:
```

```
start game
build settler s1 stargate
start next turn
list turn events
list all
```

- **Elvárt kimenet**

```
#list turn events
stargate crafted sg1
stargate crafted sg2
```

```
#list all
Asteroid id: a1
rockThickness: -
```

```

closeToSun: -
isRandom: f
material: -
neighbours:
  -
buildings:
  -
settlers:
  settler: s1
nonPlayers:
  -

Settler id: s1:
hadActionThisTurn: t
asteroid: a1
inventory:
  materials:
    -
  starGates:
    starGate: sg1
    starGate: sg2

```

### 8.1.5 Nem elvégezhető Base Crafting egy telepessel

- **Leírás**

A játékos a telepest irányítva egy aszteroidán állva próbál meg elkészíteni egy bázist. A bázis elkészítése alapvetően akkor akadhat meg amikor a játékos a telepest 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 telepes 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 megghiúsulását eredményezhetik. A vizsgálat itt az aszteroidán lévő összes telepes inventory-án végig futhat, így itt több hiba is felléphet.

- **Bemenet**

```

init game manually

create asteroid a1
create settler s1
create inventory i1
create iron iron1
create coal coal1
create uran uran1
create ice ice1

#s1
add inventory material i1 iron1

```



```

add inventory material i1 coal1
add inventory material i1 uran1
add inventory material i1 ice1
set settler inventory s1 i1
set settler asteroid s1 a1

#a1
add asteroid settler a1 s1

#-----
#Végrehajtás:

start game
build settler s1 base
start next turn
list turn events
list all
    • Elvárt kimenet
#list turn events
-

#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
-
buildings:
-
settlers:
    settler: s1
nonPlayers:
-

Settler id: s1:
hadActionThisTurn: t
asteroid: a1
inventory:
    materials:
        material: iron1
        material: coal1
        material: uran1
        material: ice1
    starGates:
-

```

### 8.1.6 Nem elvégezhető Base Crafting telepések külön aszteroidán

- **Leírás**

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

A bázis elkészítése alapvetően akkor akadhat meg amikor a játékos a telepest irányítva úgy próbálja meg létrehozni, hogy a telepeske rendelkezzen 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 megghiúsulását eredményezheti. A vizsgálat itt az aszteroidán lévő összes telepes inventory-án végig futhat, így itt több hiba is felléphet.

- **Bemenet**

```
init game manually
```

```
create asteroid a1
create asteroid a2
create settler s1
create inventory i1
create iron iron1
create iron iron2
create iron iron3
create coal coal1
create coal coal2
create coal coal3
create settler s2
create inventory i2
create uran uran1
create uran uran2
create uran uran3
create ice ice1
create ice ice2
create ice ice3

#s1
add inventory material i1 iron1
add inventory material i1 iron2
add inventory material i1 iron3
add inventory material i1 coal1
add inventory material i1 coal2
add inventory material i1 coal3
set settler inventory s1 i1
set settler asteroid s1 a1

#a1
add asteroid settler a1 s1

#s2
add inventory material i2 uran1
```

```

add inventory material i2 iron2
add inventory material i2 iron3
add inventory material i2 coal1
add inventory material i2 coal2
add inventory material i2 coal3
set settler inventory s2 i2
set settler asteroid s2 a2

#a1
add asteroid settler a2 s2

#-----
#Végrehajtás:

start game
build settler s1 base
start next turn
list turn events
list all
    • Elvárt kimenet
#list turn events
-

#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
-
buildings:
-
settlers:
    settler: s1
nonPlayers:
-

Asteroid id: a2
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
-
buildings:
-
settlers:
    settler: s2
nonPlayers:
-

```

```

Settler id: s1:
hadActionThisTurn: t
asteroid: a1
inventory:
  materials:
    material: iron1
    material: iron2
    material: iron3
    material: coal1
    material: coal2
    material: coal3
  starGates:
    -

```

```

Settler id: s2:
hadActionThisTurn: t
asteroid: a2
inventory:
  materials:
    material: uran1
    material: uran2
    material: uran3
    material: ice1
    material: ice2
    material: ice3
  starGates:
    -

```

### 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 telepesenél és nála megvannak a megfelelő nyersanyagok. A bázis elkészítését vizsgálja abban az esetben amikor a létrehozáshoz szükséges nyersanyagokkal rendelkeznek a játékos által irányított telepések, 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 megghiúsulását eredményezheti. A vizsgálat itt az aszteroidán lévő összes telepes inventory-án végig futhat, így itt több hiba is fellelhető.

- **Bemenet**

```

init game manually

create asteroid a1
create settler s1
create inventory i1
create iron iron1

```

```

create iron iron2
create iron iron3
create coal coal1
create coal coal2
create coal coal3
create settler s2
create inventory i2
create uran uran1
create uran uran2
create uran uran3
create ice ice1
create ice ice2
create ice ice3

```

```

#s1
add inventory material i1 iron1
add inventory material i1 iron2
add inventory material i1 iron3
add inventory material i1 coal1
add inventory material i1 coal2
add inventory material i1 coal3
set settler inventory s1 i1
set settler asteroid s1 a1

```

```

#a1
add asteroid settler a1 s1

```

```

#s2
add inventory material i2 uran1
add inventory material i2 iron2
add inventory material i2 iron3
add inventory material i2 coal1
add inventory material i2 coal2
add inventory material i2 coal3
set settler inventory s2 i2
set settler asteroid s2 a1

```

```

#a2
add asteroid settler a1 s2

```

```

#-----
#Végrehajtás:

```

```

start game
build settler s1 base
start next turn
list turn events
list all

```

- **Elvárt kimenet**

```

#list turn events
base crafted b1

```

```
#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
  -
buildings:
  building: b1
settlers:
  settler: s1
  settler: s2
nonPlayers:
  -
Base id: b1
asteroid: a1

Settler id: s1:
hadActionThisTurn: t
asteroid: a1
inventory:
  materials:
    -
  starGates:
    -

Settler id: s2:
hadActionThisTurn: t
asteroid: a1
inventory:
  materials:
    -
  starGates:
    -
```

### 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 asteroid a1
```

```
create settler s1
create inventory i1
create stargate g11
create stargate g12

#s1
add inventory stargate i1 g11
add inventory stargate i1 g12
set settler inventory s1 i1
set settler asteroid s1 a1

#g11
set stargate neighbour g11 g12

#g12
set stargate neighbour g12 g11

#a1
add asteroid settler a1 s1

#-----
#Végrehajtás:

start game
place gate settler s1 g11
start next turn
list turn events
list all
  • Elvárt kimenet
#list turn events
place gate a1 g11

#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
  -
buildings:
  building: g11

settlers:
  settler: s1
nonPlayers:
  -

StarGate id: g11
isRandom: f
works: t
```

```

wasInSunFlare: f
neighbour: g12
asteroid: a1

StarGate id: g12
isRandom: f
works: f
wasInSunFlare: f
neighbour: g11
asteroid: -

Settler id: s1:
hadActionThisTurn: t
asteroid: a1
inventory:
  materials:
    -
  starGates:
    starGate: g12

```

### 8.1.9 Második Stargate lehelyezése a párból

- **Leírás**

A játékos a telepest 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 mehet tönkre, ha a tartalmazó aszteroida felrobban, de ekkor a pár minkét tagja törlésre kerül.

A tesztelésben lehetséges hibahely a kapupár működésének a vizsgálata.

- **Bemenet**

```

init game manually

create asteroid a1
create asteroid a2
create settler s1
create inventory i1
create stargate g11
create stargate g12

#g11
set stargate neighbour g11 g12

#g12
set stargate neighbour g12 g11

#s1

```



```
add inventory stargate g12
set settler inventory s1 i1
set settler asteroid s1 a1
```

```
#a1
add asteroid building g11
add asteroid settler a1 s1
```

```
#-----
#Végrehajtás:
```

```
start game
place gate settler s1 g12
start next turn
list turn events
list all
```

- **Elvárt kimenet**

```
#list turn events
place gate a1 g11
```

```
#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
-
buildings:
  building: g11
```

```
settlers:
-
nonPlayers:
-
```

```
Asteroid id: a2
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
-
buildings:
  building: g12
settlers:
  settler: s1
nonPlayers:
-
```

```
StarGate id: g11
```

```

isRandom: f
works: t
wasInSunFlare: f
neighbour: g12
asteroid: a2

StarGate id: g12
isRandom: f
works: t
wasInSunFlare: f
neighbour: g11
asteroid: a1

Settler id: s1:
hadActionThisTurn: t
asteroid: a1
inventory:
  materials:
    -
  starGates:
    -

```

## 8.2 Napvihar kitörése nem átfűrt 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 asteroid a1
create asteroid a2
create settler s1
create robot r1
create alien al1
create stargate g1
create stargate g2
create inventory iv1

#sun1
add sun asteroid sun1 a1

```

```

set sun deterministic behaviour sun1 t

#a1
set asteroid property a1 rockthickness 1
add asteroid settler a1 s1
add asteroid nonplayer a1 r1
add asteroid nonplayer a1 all
add asteroid neighbour a1 a2
add asteroid building a1 g1
set asteroid deterministic behaviour a1 t

#a2
add asteroid neighbour a2 a1
set asteroid deterministic behaviour a2 t

#s1
set settler property s1 asteroid a1
set settler property s1 inventory iv1

#r1
set robot property r1 asteroid a1
set robot deterministic behaviour r1 t

#all
set alien property all asteroid a1
set alien deterministic behaviour all t

#g1
set stargate neighbour g1 g2
set stargate asteroid g1 a1
set stargate deterministic behaviour g1 t

#g2
set stargate neighbour g2 g1
set stargate deterministic behaviour g2 t

#iv1
add inventory stargate iv1 g2

#-----
#Vegrehajtas:

start game
sunflare sun1
start next turn
list turn events
list all

    • Elvárt kimeneti

#list turn events

```

```
sunflare on sun sun1
settler died s1
starGate broke g1
starGate broke g2
robot died r1
alien died a11
```

```
#list all
```

```
Sun id: sun1
isRandom: f
asteroids:
```

```
-
```

```
Asteroid id: a1
rockThickness: 1
closeToSun: -
isRandom: f
material: -
neighbours:
    neighbour: a2
buildings:
    building: g1
settlers:
    -
nonPlayers:
    -
```

```
Asteroid id: a2
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
    neighbour: a1
buildings:
    -
settlers:
    -
nonPlayers:
    -
```

### 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 asteroid a1
create asteroid a2
create settler s1
create robot r1
create alien al1
create stargate g1
create stargate g2
create inventory iv1
```

```
#sun1
add sun asteroid sun1 a1
set sun deterministic behaviour sun1 t
```

```
#a1
set asteroid property a1 rockthickness 1
add asteroid settler a1 s1
add asteroid nonplayer a1 r1
add asteroid nonplayer a1 al1
add asteroid neighbour a1 a2
add asteroid building a1 g1
set asteroid deterministic behaviour a1 t
```

```
#a2
add asteroid neighbour a2 a1
set asteroid deterministic behaviour a2 t
```

```
#s1
set settler property s1 asteroid a1
set settler property s1 inventory iv1
```

```
#r1
set robot property r1 asteroid a1
set robot deterministic behaviour r1 t
```

```
#al1
set alien property al1 asteroid a1
set alien deterministic behaviour al1 t
```

```
#g1
set stargate neighbour g1 g2
```

```

set stargate asteroid g1 a1
set stargate deterministic behaviour g1 t

#g2
set stargate neighbour g2 g1
set stargate deterministic behaviour g2 t

#iv1
add inventory stargate iv1 g2

#-----
#Vegrehajtas:

start game
sunflare sun1
start next turn
list turn events
list all

```

- **Elvárt kimeneti**

```

#list turn events

sunflare on sun sun1
settler died s1
starGate broke g1
starGate broke g2
robot died r1
alien died a1

#list all

Sun id: sun1
isRandom: f
asteroids:
    -

Asteroid id: a1
rockThickness: 1
closeToSun: -
isRandom: f
material: -
neighbours:
    neighbour: a2
buildings:
    building: g1
settlers:
    -
nonPlayers:
    -

```

```

Asteroid id: a2
rockThickness: -
closeToSun: -
isRandom: f
material: -
neighbours:
    neighbour: a1
buildings:
    -
settlers:
    -
nonPlayers:
    -

```

#### 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 fúrás már nem tudja megtenni.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A Fúrás robottal use-case megvalósítását teszteli.

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

- **Bemenet**

```

init game manually

create asteroid a1
create robot r1

#a1
set asteroid rockThickness a1 2
add asteroid nonPlayers a1 r1

#r1
set robot asteroid r1 a1
set robot deterministic behaviour r1 t

#-----
#Vegrehajtás:

start game
drill robot r1

start next turn
list turn events
drill robot r1

```

```
start next turn
list turn events
drill robot r1
```

```
start next turn
list turn events
```

```
list all
```

- **Elvárt kimeneti**

```
#list turn events 1:
asteroid drilled a1
```

```
#list turn events 2:
asteroid drilled a1
```

```
#list turn events 3:
-
```

```
#list all:
Asteroid id: a1
rockThickness: 0
closeToSun: -
isRandom: -
material: -
neighbours:
-
buildings:
-
settlers:
-
nonPlayers:
  nonPlayer: r1
```

```
Robot id: r1
isRandom: f
asteroid: a1
```

## 8.5 Fúrás robottal, elpárolog

- **Leírás**

A robot fúr egyet, egy aszteroidán. Az aszteroida kéregvastagsága egy volt a fúrás előtt. Az aszteroida a tesztelés végéig folyamatosan napközben 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 átfúrni a kérget, a jég pedig nem párolog el.

- **Bemenet**

```
init game manually

create asteroid a1
create robot r1
create ice e1

#a1
set asteroid closeToSun a1 t
set asteroid rockThickness a1 1
add asteroid nonPlayer a1 r1
add asteroid material a1 e1
set asteroid deterministic behaviour a1 t

#r1
set robot asteroid r1 a1
set robot deterministic behaviour r1 t

#e1
set material exposed e1 0

#Vegrehajtás:

start game

drill robot r1

start next turn
list turn events

list all

#-----
```

- **Elvárt kimenet**

```
#list turn events
asteroid drilled a1
material evaporated e1

#list all
Asteroid id: a1
rockThickness: 0
closeToSun: t
isRandom: f
materials:
-
neighbours:
-
```

```

buildings:
  -
settlers:
  -
nonPlayers:
  nonPlayer: r1

Robot id: r1
isRandom: f
asteroid: a1

```

## 8.6 Fúrás robottal, felrobban

- **Leírás**

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

Ez három kör felfedettséggel 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 asteroid a1
create asteroid a2
create robot r1
create uran u1

#a1
set asteroid closeToSun a1 t
set asteroid rockThickness a1 1
add asteroid nonPlayer a1 r1
add asteroid neighbour a1 a2
add asteroid material a1 u1
set asteroid deterministic behaviour a1 t

#a2
add asteroid neighbour a2 a1

#r1
set robot asteroid r1 a1
set robot deterministic behaviour r1 t

```

```
#u1
set material exposed u1 0

#Vegrehajtás:

start game

drill robot r1

start next turn
list turn events

start next turn

start next turn
list turn events

list all
```

- **Elvárt kimenet**

```
#list turn events 1
asteroid drilled a1

#list turn events 2
material exploded u1
asteroid exploded a1
robot move r1 a2

#list all
Asteroid id: a2
rockThickness: -
closeToSun: -
isRandom: -
materials:
  -
neighbours:
  -
buildings:
  -
settlers:
  -
nonPlayers:
  nonPlayer: r1

Robot id: r1
isRandom: f
asteroid: a2
```

## 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 tesztet végéig folyamatosan napközeli áll. A magjában egy radioaktív urán található.

Ez három kör felfedezés 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 átfúrni a kérget, az urán pedig nem robban fel 3 kör múlva, vagy esetleg korábban robban fel.

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

- **Bemenet**

```
init game manually

create asteroid a1
create asteroid a2
create robot r1
create uran u1

#a1
set asteroid closeToSun a1 t
set asteroid rockThickness a1 1
add asteroid nonPlayer a1 r1
add asteroid material a1 u1
set asteroid deterministic behaviour a1 t

#a2
#-

#r1
set robot asteroid r1 a1
set robot deterministic behaviour r1 t

#u1
set material exposed u1 0

#Vegrehajtás:

start game

drill robot r1

start next turn
list turn events

start next turn

start next turn
list turn events
```

```
list all
```

- **Elvárt kimenet**

```
#list turn events 1
asteroid drilled a1

#list turn events 2
material exploded u1
asteroid exploded a1
robot died r1

#list all
Asteroid id: a2
rockThickness: -
closeToSun: -
isRandom: -
materials:
  -
neighbours:
  -
buildings:
  -
settlers:
  -
nonPlayers:
  -
```

## 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 asteroid a1
create settler s1

#a1
set asteroid rockThickness a1 2
add asteroid settler a1 s1
```

```
#s1
set settler hadActionThisTurn s1 f
set settler asteroid s1 a1

#-----
#Vegrehajtas:

start game
drill settler s1

start next turn
list turn events
drill settler s1

start next turn
list turn events
drill settler s1

start next turn
list turn events

list all
```

- **Elvárt kimenet**

```
#list turn events 1:
asteroid drilled a1

#list turn events 2:
asteroid drilled a1

#list turn events 3:
-

#list all:
Asteroid id: a1
rockThickness: 0
closeToSun: -
isRandom: -
material: -
neighbours:
-
buildings:
-
settlers:
  settler: s1
nonPlayers:
-

Settler id: s1
```

```
hadActionThisTurn: f
asteroid: a1
inventory:
  -
```

## 8.9 Fúrás telepessel, elpárolog

- **Leírás**

A telepes fúr egyet, egy aszteroidán. Az aszteroida kéregvastagsága egy volt a fúrás előtt. Az aszteroida a tesztet végéig folyamatosan napközben 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 asteroid a1
create settler s11
create ice e1

#a1
set asteroid closeToSun a1 t
set asteroid rockThickness a1 1
add asteroid settler a1 s11
add asteroid material a1 e1
set asteroid deterministic behaviour a1 t

#s11
set settler hadactionthisturn s11 f
set settler asteroid s11 a1

#e1
set material exposed e1 0

#Vegrehajtas:

start game

drill settler s11

start next turn
list turn events

list all
```

- **Elvárt kimenet**

```
#list turn events
asteroid drilled a1
material evaporated e1
```

```
#list all
Asteroid id: a1
rockThickness: 0
closeToSun: t
isRandom: f
materials:
    -
neighbours:
    -
buildings:
    -
settlers:
    settler: s11
nonPlayers:
    -
```

```
Settler id: s11
hadActionThisTurn: f
asteroid: a1
inventory:
    -
```

## 8.10 Fúrás telepessel, felrobban

- **Leírás**

A telepes fúr egyet, egy aszteroidán. Az aszteroida kéregvastagsága egy volt a fúrás előtt. Az aszteroida a tesztet végéig folyamatosan napközben 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 átfúrní a kérget, az urán pedig nem robban fel 3 kör múlva, vagy esetleg korábban robban fel.

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

- **Bemenet**

```
init game manually
```

```
create asteroid a1
create settler s11
create uran u1
```

```
#a1
```



```

set asteroid closeToSun a1 t
set asteroid rockThickness a1 1
add asteroid settler a1 s11
add asteroid material a1 u1
set asteroid deterministic behaviour a1 t

#s11
set settler hadactionthisturn s11 f
set settler asteroid s11 a1

#u1
set material exposed u1 0

#Vegrehajtas:

start game

drill settler s11

start next turn
list turn events

start next turn

start next turn
list turn events

list all

```

- **Elvárt kimenet**

```

#list turn events 1
asteroid drilled a1

#list turn events 2
material exploded u1
asteroid exploded a1
settler died s11

#list all
-

```

## 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 asteroid a1
create asteroid a2
create robot r1

#a1
add asteroid nonplayer a1 r1

#a2

#r1
set robot asteroid r1 a1
set robot deterministic behaviour r1 t

#Vegrehajtás:

start game

move robot r1 dest a2

start next turn
list turn events

list all
```

- **Elvárt kimenet**

```
#list turn events
-

#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: -
materials:
-
neighbours:
-
buildings:
-
settlers:
-
nonPlayers:
  nonPlayer: r1
```

```

Asteroid id: a2
rockThickness: -
closeToSun: -
isRandom: -
materials:
  -
neighbours:
  -
buildings:
  -
settlers:
  -
nonPlayers:
  -

```

```

Robot id: r1
isRandom: f
asteroid: a1

```

## 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 asteroid a1
create asteroid a2
create robot r1

#a1
add asteroid nonplayer a1 r1
add asteroid neighbour a1 a2

#a2
add asteroid neighbour a2 a1

#r1
set robot asteroid r1 a1
set robot deterministic behaviour r1 t

#Vegrehajtás:

```

```
start game

move robot r1 dest a2

start next turn
list turn events

list all
```

- **Elvárt kimenet**

```
#list turn events
robot move r1 a2

#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: -
materials:
  -
neighbours:
  neighbour: a2
buildings:
  -
settlers:
  -
nonPlayers:
  -

Asteroid id: a2
rockThickness: -
closeToSun: -
isRandom: -
materials:
  -
neighbours:
  neighbour: a1
buildings:
  -
settlers:
  -
nonPlayers:
  nonPlayer: r1

Robot id: r1
isRandom: f
asteroid: a2
```

### 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 asteroid a1
create asteroid a2
create settler s11

#a1
add asteroid settler a1 s11

#a2

#s11
set settler asteroid s11 a1
set settler hadactionthisturn s1 f

#Vegrehajtás:

start game

move settler s11 dest a2

start next turn
list turn events

list all
```

- **Elvárt kimenet**

```
#list turn events
-

#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: -
materials:
-
```

```

neighbours:
  -
buildings:
  -
settlers:
  settler: s11
nonPlayers:
  -

```

```

Asteroid id: a2
rockThickness: -
closeToSun: -
isRandom: -
materials:
  -
neighbours:
  -
buildings:
  -
settlers:
  settler: s11
nonPlayers:
  -

```

```

Settler id: s11
hadActionThisTurn: f
asteroid: a1
inventory:
  -

```

#### 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 asteroid a1
create asteroid a2
create settler s11

#a1
add asteroid settler a1 s11

```

```

add asteroid neighbour a1 a2

#a2
add asteroid neighbour a2 a1

#s11
set settler hadactionthisturn s11 f
set settler asteroid s11 a1

#Vegrehajtas:

start game

move settler s11 dest a2

start next turn
list turn events

list all

```

- **Elvárt kimenet**

```

#list turn events
settler move s11 a2

#list all
Asteroid id: a1
rockThickness: -
closeToSun: -
isRandom: -
materials:
  -
neighbours:
  neighbour: a2
buildings:
  -
settlers:
  -
nonPlayers:
  -

Asteroid id: a2
rockThickness: -
closeToSun: -
isRandom: -
materials:
  -
neighbours:
  neighbour: a1
buildings:
  -

```

```

settlers:
    settler: s11
nonPlayers:
    -

Settler id: s11
hadActionThisTurn: f
asteroid: a2
inventory:
    -

```

### 8.15 Telepes átlép egy kapun.

- **Leírás**

Egy telepes átlép egy kapun, az őt 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 settler S1
create asteroid A1
create asteroid A2
create stargate G1
create stargate G2
set asteroid deterministic behaviour A1 t
set asteroid deterministic behaviour A2 t
set stargate deterministic behaviour G1 t
set stargate deterministic behaviour G2 t
set settler asteroid S1 A1
add asteroid settler A1 S1
add asteroid building A1 G1
add asteroid building A2 G2
set asteroid rockthickness A1 1
set asteroid rockthickness A2 1
add stargate asteroid G1 A1
add stargate asteroid G2 A2
add stargate neighbour A1 A2
add stargate neighbour A2 A1
start game
move settler S1 dest G1
list turn events
list all

```

- **Elvárt kimenet**

```

# list turn events
settler move S1 A2
# list all
Asteroid id: A1
rockThickness: 1

```



```

closeToSun: f
isRandom: f
material -
neighbours:
neighbour: -
buildings:
    building: G1
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
Asteroid id: A2
rockThickness: 1
closeToSun: f
isRandom: f
material -
neighbours:
neighbour: -
buildings:
    building: G2
settlers:
    settler: S1
nonPlayers:
    nonPlayer: -
Settler id: S1
hadActionThisTurn: t
asteroid: A1
inventory:
materials:
    material: -

```

### 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 asteroid A1
create asteroid A2
create asteroid A3
create stargate G1
create stargate G2
create sun SU1
set asteroid deterministic behaviour A1 t
set asteroid deterministic behaviour A2 t

```

```

set asteroid deterministic behaviour A3 t
set stargate deterministic behaviour G1 t
set stargate deterministic behaviour G2 t
set sun deterministic behaviour SU1 t
add asteroid building A1 G1
add asteroid building A2 G2
set asteroid rockthickness A1 1
set asteroid rockthickness A2 1
set asteroid rockthickness A3 1
add stargate asteroid G1 A1
add stargate asteroid G2 A2
add stargate neighbour A1 A2
add stargate neighbour A2 A1
add sun asteroid SU1 A1
start game
sunflare SU1
list turn events
list all

```

- **Elvárt kimenet**

```

# list turn events
sunflare on sun SU1
stargate broke G1
stargate broke G2
stargate move G1 A3
# list all
Asteroid id: A1
rockThickness: 1
closeToSun: f
isRandom: f
material -
neighbours:
neighbour: -
buildings:
    building: -
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
Asteroid id: A2
rockThickness: 1
closeToSun: f
isRandom: f
material -
neighbours:
neighbour: -
buildings:
    building: G2
settlers:
    settler: S1
nonPlayers:
    nonPlayer: -

```

```
Asteroid id: A3
rockThickness: 1
closeToSun: f
isRandom: f
material -
neighbours:
neighbour: -
buildings:
    building: G1
settlers:
    settler: -
nonPlayers:
    nonPlayer: -
```

### 8.3 A tesztelést támogató programok tervei

Lehetőség van szöveges fájlból beolvasott tesztek végrehajtani. (load game) Ezek a fájlok a korábban megadott tesztek 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 máris 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	Részvevők	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 munkákat. Kárpáti és Mohácsi vállalta az osztályleírásokat, míg a tesztekét Bárkányi, Simon és Tóth.
2021.04.05. 14:00	2 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet: Elkezdett feladatokban felvetett hibákat megbeszéltük. A tesztesetek 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 átbeszéltünk 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. Elkezdtek a dokumentumot egységesen összeállítani. Behelyezésre kerültek átbeszélve a tesztesetek.
2021.04.12. 10:00	4 óra	Tóth Bárkányi Simon Mohácsi Kárpáti	Értekezlet. Véglegesre formáztuk közösen a dokumentumot.