**ASZTEROIDABÁNYÁSZAT**

Szoftvertechnológia házi feladat

Szabó Bence Sándor

NQB6UO

# A feladat leírása

# Az emberiség elhatározta, hogy kiaknázza az aszteroidabányászatban rejlő lehetőségeket. Ehhez a telepeseknek fel kell építeniük egy űrbázist az aszteroidaövben. Az építéshez szükséges erőforrásokat túlságosan drága lenne odavinni, így azokat az aszteroidákból kell kinyerni.

# A játékosok a telepeseket irányítják. A telepesek egyszemélyes űrhajókkal járják az aszteroidákat megfelelő nyersanyagok után kutatva. A mozgás atomi gyorsaságú.

# Az aszteroidák külsejét különböző vastagságú sziklarétegek borítják. A fontos nyersanyagok (vízjég, vas, szén, stb.) az aszteroidák magjában találhatók. Vannak veszélyes aszteroidák, amelyek magjának anyaga erősen radioaktív. Vannak üreges aszteroidák is, amelyek magjában nem található nyersanyag. Az aszteroidák magja mindig homogén, nem keverednek benne a különböző anyagok.

# Egy telepes egy lépésben egyfajta műveletet hajthat végre és ebből is csak egyet. A lehetséges műveletek: mozgás, fúrás, bányászat. Mozgás során a telepes az űrhajójával átmegy egy szomszédos aszteroidára(minden aszteroidának néhány, esetenként több száz szomszédja van). Fúrás során a telepes egy egységnyivel tudja mélyíteni az aszteroida köpenyébe fúrt lyukat. Bányászat során a telepes kinyeri a fúrt lyukon keresztül az aszteroida magjában található erőforrást, de ez a lépés csak akkor lehetséges, ha az aszteroida köpenyét már sikerült teljesen átfúrni.

# A telepesekre azonban veszélyek is leselkednek. A radioaktív aszteroidáknak, ha elérhető a belseje (van oda vezető lyuk) és napközelbe kerül, akkor az felrobban, a robbanás a rajta lévő telepeseket megöli. Veszélyt jelentenek a telepesekre a napviharok is, amelyek időnként elérik az aszteroidaövet, adott körönként van esély egyre, ennek esélye meg nem történt esetek hatására nő. A napvihar csak úgy élhető túl, ha az aszteroida, amin a telepes van, üreges és magja elérhető.

# A telepeseket mesterséges intelligencia által vezérelt autonóm robotok is segítik. Ezek a robotok csak szomszédos aszteroidák közötti mozgásra és fúrásra képesek. A robotok bányászni nem tudnak, mert nyersanyagot nem tudnak magukkal vinni. A robotok túlélik a radioaktív robbanást, és ilyenkor valamelyik szomszédos aszteroidán landolnak. A napvihar azonban a robotokat is tönkreteszi, ha nem egy olyan aszteroidán vannak, ami üreges és magja elérhető.

# A játék kétféleképpen érhet véget. Ha minden telepes meghalt, akkor a játékosok veszítettek. Ha azonban sikerült mindenfajta nyersanyagból legalább egy egységet kibányászni és egy közös aszteroidára összegyűjteni, akkor a telepesek felépíthetik a bázist és meg nyerik a játékot.

## Elsődleges követelmények

|  |  |  |
| --- | --- | --- |
| **Azonosító** | **Leírás** | **Use-case** |
| R01 | A játék körökre osztott, a játékos minden körben minden meglévő telepesek mindegyikének pontosan egy utasítást tud adni. | Telepesek irányítása |
| R02 | A játékos, telepest irányít egy aszteroida mezőben. | Telepesek irányítása |
| R03 | A telepesek képesek egy szomszédos aszteroidára mozogni. Emellett tud fúrni és bányászni velük. Minden körben ezek közül csak egyet. | Telepesek irányítása,  Mozgás,  Fúrás,  Bányászás |
| R04 | Játékos telepessel fúrással el kell jutni az aszteroida belsejéhez, mielőtt kinyeri a nyersanyagot belőle. Minden fúrással eggyel közelebb kerül a maghoz, az aszteroida rétegének vastagsága mutatja meg hányszor kell azon az aszteroidán fúrni, míg a magjáig elérünk. | Fúrás |
| R05 | A játékos nem választhatja a Fúrás opciót, ha már van az aszteroidán egy lyuk, ami már elért a maghoz. | Fúrás |
| R06 | A játékosnak meghatározott nyersanyagokat kell gyűjtenie, a győzelemhez. | Bányászás |
| R07 | 10 féle nyersanyag létezik: Alumínium, Kobalt, Nikkel, Vas, Réz, Jég, Urán (radioaktív), Plutónium (radioaktív), Szilikon, Szén.  Mindegyikből 1 egység kell a győzelemhez. | Bányászás |
| R08 | Az összegyűjtött nyersanyagokat egy aszteroidára kell vinnie, nem tudja ott hagyni a nyersanyagot, így egy közös aszteroidára kell irányitani a telepeseket, akiknél az anyagok összessége megfelel a bázis építés feltételének, ekkor megépül a bázis és a játékos nyer. | Telepesek irányítása |
| R09 | Játékos telepessel bányász, ha elérte az aszteroida közepét, ekkor nyersanyaghoz jut és az aszteroidát üregessé teszi. | Bányászás,  Telepesek irányítása |
| R10 | A telepesek hajóval mozognak, melyben akármennyi nyersanyag fér. | Telepesek irányítása |
| R11 | Ha a játékos megszerzi az összes nyersanyagot, ami kell, és elviszi az épülő bázisra, akkor nyer. | Bányászás |
| R12 | Elsőnek a 10.körben fixen történik egy.  Utána minden 3. körben van esélye (20%, ha nem történik a következő körben +20%, amint megtörténi, az esély resetelődik) egy napkitörés bekövetkeztének. Ennek során, aki nincs üregben elpusztul (robotokok és telepesek is). | Napkitörés vezérlése |
| R13 | Napkitörést egy üreges aszteroida belsejébe bújva túl lehet élni. | Napkitörés vezérlése,  RTelepesek irányítása,  Robotok irányítása |
| R14 | Léteznek radioaktív belsejű aszteroidák, amiknek, ha van a magjához vezető fúrat, a nap közelébe érve felrobbannak. | Aszteroidák vezérlése |
| R15 | Egy aszteroidán, lehet több telepes/robot, de nem fúrhatnak egyszerre. Ha üreges, akkor nem bújhatnak el egynél többen, ugyan abban a lyukban. | Telepesek irányítása,  Robotok irányítása |
| R16 | Ha a játékos összes telepese elpusztul, a játékos veszít. | Telepesek irányítása,  Napkitörés vezérlése |
| R17 | Az aszteroidák kéreg nagysága a lebontáshoz szükséges körök számában mérendők. Ez a szám 1-10 között lehet. | Aszteroidák vezérlése |
| R18 | A rendszer robotokat vezérel. | Robotok irányítása |
| R19 | A robotok célja, hogy segítsék a telepeseket, tehát a lehető legtöbb aszteroidát kifúrjanak, egészen a magjáig. | Robotok irányítása |
| R20 | A robotok nem képesek bányászni, csak szomszédos aszteroidára tudnak menni, fúrni, esetleg egy üregben elbújni. | Robotok irányítása |
| R21 | Az aszteroidák minden körben mozognak. | Aszteroidák vezérlése |
| R22 | A telepesek meghalnak, ha az az aszteroida felrobban, amin a robbanás pillanatában vannak. | Aszteroidák vezérlése |
| R23 | A robotok nem halnak meg, ha az az aszteroida felrobban, amin a robbanás pillanatában vannak. De egy véletlenszerűen választott szomszédra kerülnek. | Aszteroidák vezérlése,  Robotok irányítása |
| R24 | A radioaktív anyagok szállításkor, ugyan úgy viselkednek, mintha nem lennének azok. | Telepesek irányítása |
| R25 | Nincs kevert aszteroida, tehát minden aszteroidából csak egy fajta anyagot lehet kinyerni. | Aszteroidák vezérlése,  Telepesek irányítása |
| R26 | Egy Aszteroida egy egység anyagot tartalmaz, egy bányászás után, üregessé válik. | Aszteroidák vezérlése,  Bányászás |

## További követelmények

|  |  |  |
| --- | --- | --- |
| **Azonosító** | **Leírás** | **Use-case** |
| R27 | Egy aszteroidának több száz szomszédja is lehet, de legalább 1. | Aszteroidák vezérlése |
| R28 | Ha egy aszteroida megszűnik, minden szomszédja megkapja a többi szomszédját szomszédjául. | Aszteroidák vezérlése |

# Use-case-ek

## Use-case diagram

## Use-case leírások

|  |  |
| --- | --- |
| **Cím** | **Controll Settlers** |
| **Leírás** | Játékos mindegyik meglévő telepesnek egy körben, egy utasítást ad. |
| **Aktorok** | Player |
| **Főforgatókönyv** | **1.** A játékos 3 utasítás közül választ egyet: Mozgatás, Fúrás vagy Bányászás. |
| **Alternatív forgatókönyv** | **1.A.1.**  A játékos, ha üreges aszteroidán van és eléri az üreget, el tud bújni, ha az üreg üres. |
| **Alternatív forgatókönyv** | **1.A.1.A.1.** Ha már el van bújva választhat, hogy marad elbújva, vagy kimászik onnan. Ha kimászik onnan továbbra is válaszhat a főforgatókönyv szerint felsorolt műveletek közül. |

|  |  |
| --- | --- |
| **Cím** | **Move (Controll Settlers)** |
| **Leírás** | A játékos a kiválasztott telepest mozgatja. |
| **Aktorok** | Player |
| **Főforgatókönyv** | **1.** A játékos az adott telepessel, egy a telepes tartózkodási aszteroidának a szomszédjára mozog. |
| **Alternatív forgatókönyv** | **1.A.1.**  A játékos olyan aszteroidára lépteti a telepest, ahol vannak már más telepesek és így összeségben a telepeseknél megvan minden szükséges anyag, ekkor a játékos megnyeri a játékot. |

|  |  |
| --- | --- |
| **Cím** | **Drill (Controll Settlers)** |
| **Leírás** | A játékos a kiválasztott telepessel fúr. |
| **Aktorok** | Player |
| **Főforgatókönyv** | **1.** A telepes aktuális tartózkodási helyén réteggel lejjebb fúr. Ha még nem történt az aszteroidán, egy új lyukat kezd, ha már van lyuk, akkor mindenképp azt folytatja. |

|  |  |
| --- | --- |
| **Cím** | **Mine** |
| **Leírás** | A játékos a kiválasztott telepessel bányász az adott aszteroidán. |
| **Aktorok** | Player |
| **Főforgatókönyv** | **1.** A telepes aktuális tartózkodási aszteroidából kibányássza az aszteroidában lévő nyersanyagot és eltárolja az űrhajójában. Az aszteroida üregessé válik. |
| **Alternatív forgatókönyv** | **1.A.1.** Az aszteroidán, ahol kibányászta a nyersanyagot, a rajta lévő telepeseknél megtalálható a szükséges nyersanyagok, a játékos nyer. |

|  |  |
| --- | --- |
| **Cím** | **View Solar System** |
| **Leírás** | A játékos megtekinti a pályát. |
| **Aktorok** | Player |
| **Főforgatókönyv** | **1.** A rendszer megjeleníti az aszteroida övet.  **2.** A játékos meg tekinti az aszteroida övet.  **3.** A játékos értesül, mennyi idő múlva lesz napkitörés.  **4.** A játékos látja, melyik telepesénél, milyen anyagok vannak. |

|  |  |
| --- | --- |
| **Cím** | **Controll Robots** |
| **Leírás** | A rendszer utasításokat ad a robotoknak. |
| **Aktorok** | System |
| **Főforgatókönyv** | **1.** A Rendszer minden általa irányított robotnak utasítást adhat, két féle utasítás lehetséges: mozgás vagy fúrás. |

|  |  |
| --- | --- |
| **Cím** | **Move (Controll Robots)** |
| **Leírás** | A rendszer mozgatja a kiválasztott robotot. |
| **Aktorok** | System |
| **Főforgatókönyv** | **1.** A Rendszer úgy látja, hogy ahol a robot van már van fúrt lyuk a magig, így úgy dönt egy szomszédos aszteroidára irányítja.  **2.** A Rendszer úgy látja, hogy az aszteroidán már egy robot fúr (a rendszer utasítást adott, hogy ebben a körben fúrjon), így, hogy ne fúrjanak együtt odébb megy. |
| **Alternatív forgatókönyv** | **1.A.1.** Ha a rendszer érzékeli, hogy van esély nap kitörésre, keres egy aszteroidát ahova elbújhat és el is búj. |

|  |  |
| --- | --- |
| **Cím** | **Drill (Controll Robots)** |
| **Leírás** | A rendszer fúratja a robotot. |
| **Aktorok** | System |
| **Főforgatókönyv** | **1.** A Rendszer látja, hogy az aszteroidán, amin a robot van, még nincs magig nyúló lyuk, így fúrat a robottal, így a lyuk egy réteggel mélyebb lesz. |

|  |  |
| --- | --- |
| **Cím** | **Controll Solar Eruptions** |
| **Leírás** | A rendszer eldönti, hogy legyen e napkitörés. |
| **Aktorok** | System |
| **Főforgatókönyv** | **1.** A Rendszer látja, hogy még nincs itt az ideje (nem telt le a kör számláló), ahhoz, hogy napkitörés legyen.  **2.** A Rendszer látja, hogy a kör vissza számláló 0 így sorsol egyet az aktuális napkitörés esélynek megfelelően, ha nem sikeres a sorsolás a rendszer nem idéz elő napkitörést, a körszámlálót 3ra állítja és a napkitörés esélyét növeli 20 %-kal.  **3.** A Rendszer látja, hogy a kör vissza számláló 0 így sorsol egyet az aktuális napkitörés esélynek megfelelően, ha sikeres a sorsolás megtörténik a napkitörés, tehát mindenki elpusztul, aki nem bújt el aszteroidában. A számlálót 3ra, a kitörés valószínűségét 20%-ra állítja. |

|  |  |
| --- | --- |
| **Cím** | **Controll Asteroids** |
| **Leírás** | A rendszer vezényli az aszteroidákat. |
| **Aktorok** | System |
| **Főforgatókönyv** | **1.** Egy ellipszis pályán mozgatja az aszteroidákat. |
| **Alternatív forgatókönyv** | **1.A.1.** Ha az aszteroida, radioaktív és van a magjához vezető lyuk, valamint a nap közelében kerül, felrobban. |

# Strukturális leírás

## Az osztályok leírása

### TurnManager

**Felelősségek**

A játék körökre osztott tulajdonságáért felelős, számontartja, hogy hányadik kör van. Csak akkor lépteti a kört, ha az összes user befejezte.

**Attribútumok**

|  |  |
| --- | --- |
| -userNum: int | Számontartja a userek számát. |
| -doneNum: int | A már körükkel végzett userek számát tartja számon. |
| -turnNum: int | Az éppen aktuális kör sorszámát tárolja. |
| -users: TurnUser[1..\*] | A usereket tárolja. |

**Metódusok**

|  |  |
| --- | --- |
| +Done() | Egy user hívja meg ezt a metódust, ha végez a körével. Maga a metódus növeli a doneNum-ot és ja a doneNum=userNum változóval, meghívja az EndTurn() metódust. |
| -EndTurn() | Lezárja a kört, a doneNum-ot lenullázza valamint az összes useren (ami eleme a usersnek) meghívja a Reset() metódust. |

### *TurnUser*

**Felelősségek**

Absztrakt osztály, neki a leszármazottjai tudnak egy körben akciókat végrehajtani, addignincs vége egy teljes körnek, amíg az összes user nem cselekedett (lásd. feljebb).

**Attribútumok**

|  |  |
| --- | --- |
| #done: boolean | Azt mutatja meg, hogy végzett e már a körével. |
| #manager: TurnManager | Az őt irányító TurnManagerre hivatkozik. |

**Metódusok**

|  |  |
| --- | --- |
| +Reset() | Felüldefiniálható metódus, de mindenképp átállítja a done attribútumot false értékre. |
| -EndTurn() | Felüldefiniálható metódus, de mindenképp meghívja a manager Done() metódusát. |

### SolarSystem

**Felelősségek**

A naprendszer megvalósításáért felelős. Tárolja az aszteroidákat, a vezérlőket, illetve felelős a napkitörésekért is. A *TurnUser* leszármazottja, így a körökben tevékenységeket tud csinálni.

**Attribútumok**

|  |  |
| --- | --- |
| -solarEruptionChance: double | A napkitörés aktuális valószínűségét tárolja. |
| -turnLeft: int | A következő napkitörésig hátralévő körök száma. Ha eléri a 0-át, akkor van esélye, hogy napkitörés történik. |
| -myGame:Game | A game osztály referenciája. |
| -robotController: RobotController | A robotok vezérlője. |
| -settlerController: SettlerController | A telepesek vezérlője. |
| -asteroids: Asteroid[1..\*] | A naprendszerben lévő aszteroidák. |
| -neccesery: Material[1..\*] | A győzelemhez szükséges nyersanyagok listája. |

**Metódusok**

|  |  |
| --- | --- |
| -MoveAsteroids() | Minden aszteroidának ad egy új helyet, minden körben meghívódik ez a metódus. |
| -MakeSolarEruption() | A napkitörést valósítja meg, tehát ellenőrzi, hogy az adott körben van esély napkitörésre, ha van akkor sorsol az aktuális valószínűséggel (solarEruptionChance), ha sikeres létrejön a napkitörés és laphelyzetbe állítja a valószínűséget, ha sikertelen nem történik napkitörés, de az esélye nő. Mindkét esetben a turnLeft alaphelyzetbe áll. |
| -CheckLose() | Ellenőrzi a játékos vereségét, tehát hogy van e még élő telepese, ha nincs vesztett és a játéknak vége. |
| +Win() | A hozzátartozó objektumok tudják ezt meghívni, hogy jelezzék a győzelmet. Ő ezt továbbítja a myGamenek. |
| +EndTurn() | A *TurnUser* őstől származik és egykörét valósítja meg. |
| +RmAsteroid(removable:Asteroid) | Egy megadott aszteroidát eltávolít a astroids listából. |

### SolarEruption

**Felelősségek**

Egy napkitörésért felelős. Létrehozás után leszimulál egy kitörést, majd törli magát.

**Attribútumok**

|  |  |
| --- | --- |
| -allAsteroids | A naprendszerben lévő összes aszteroidának a referenciáját tárolja. |

**Metódusok**

|  |  |
| --- | --- |
| +KillEverything() | A napkitörés halálos hatását valósítja meg. Az összes aszteroidára applikálja a hatását. |
| -ESelfDestruct | Önmegsemmisítő metódus. |

### Game

**Felelősségek**

A játék vezérléséért felelős.

**Attribútumok**

|  |  |
| --- | --- |
| -solarSystem | Az éppen aktuális naprendszer adatait tárolja. |

**Metódusok**

|  |  |
| --- | --- |
| +StartGame() | Új játékot indít. |
| +EndGame(type: EndType) | A játékot befejezi. |
| -GenerateNewSolarSystem() | Új naprendszert generál. |

### EndType

**Felelősségek**

Ez egy enumeráció, ami a játék végének a típusának jelzésére kell. Két értéke van a Win és Lose, Win = győzelem, Lose = vereség.

### RobotController

**Felelősségek**

A robotok vezérléséért felelős. A *TurnUser* leszármazottja. Ő felel azért, hogy egy körben, egy robot, egy és csakis egy cselekvést tegyen.

**Attribútumok**

|  |  |
| --- | --- |
| -robots: Worker[0..\*] | Az „élő” robotokat tárolja. |
| -workGiven: int | A már kiadott utasítások számát tárolja. |
| -robotNum: int | Az „élő” robotok számát tárolja. |

**Metódusok**

|  |  |
| --- | --- |
| +Move(selected:Worker, destination: Asteroid) | Egy kiválasztott robotnak, kiadja az utasítást, hogy a megadott aszteroidára mozogjon. |
| +Drill(selected: Worker) | A Egy kiválasztott robotnak, kiadja az utasítást, hogy fúrjon a tartózkodási helyén. |
| +RmRobot(removable: Worker) | A kiválasztott robotot kiszedi a rotots listából. |
| +Reset() | A *TurnUser* ősől származik, minden robot madeActionjét falsera állítja vissza és a workGivent 0zza |
| +EndTurn | A *TurnUser* ősől származik, ha minden robotnak van már parancs adva, meghívja a manager Done fv.ét. |

### SettlerController

**Felelősségek**

A settlerek vezérléséért felelős. A *TurnUser* leszármazottja. Ő felel azért, hogy egy körben, egy settler, egy és csakis egy cselekvést tegyen.

**Attribútumok**

|  |  |
| --- | --- |
| -settlers: Settler[1..\*] | Az „élő” settlereket tárolja. |
| -workGiven: int | A már kiadott utasítások számát tárolja. |
| -settlerNum: int | Az „élő” settlerek számát tárolja. |

**Metódusok**

|  |  |
| --- | --- |
| +Move(selected:Settler, destination: Asteroid) | Egy kiválasztott settlernek, kiadja az utasítást, hogy a megadott aszteroidára mozogjon. |
| +Drill(selected: Settler) | Egy kiválasztott settlernek, kiadja az utasítást, hogy fúrjon a tartózkodási helyén. |
| +Mine(selected: Settler) | Egy kiválasztott settlernek, kiadja az utasítást, hogy kinyerje a tartózkodási helyén lévő nyersanyagot, csak akkor teljesül, ha a feltételek adottak. |
| +RmSettler(removable: Settler) | A kiválasztott settlert kiszedi a settlers listából. |
| +Reset() | A *TurnUser* ősől származik, minden settler madeActionjét falsera állítja vissza és a workGivent 0zza |
| +EndTurn | A *TurnUser* ősől származik, ha minden settlernek van már parancs adva, meghívja a manager Done fv.ét. |

### Worker

**Felelősségek**

A Settler ősosztálya, de tökéletesen használható, robotok adatainak tárolására, így ennek példányosításával egy robotnak megfelelő objektumot kapunk. A Move és Drill interfészt valósítja meg.

**Attribútumok**

|  |  |
| --- | --- |
| #madeAction: boolean | Tárolja, hogy végzett e munkát. |
| #location | A tartózkodási aszteroidát tárolja. |
| -myController: RobotController | A robotcontrollert tárolja, csak akkor érhető el ha közvetlenül innen példányosítunk. |

**Metódusok**

|  |  |
| --- | --- |
| +Move(destination: Asteroid) | A kapott aszteroidára mozog. |
| +Drill() | a location aszteroidán fúr, a feltételeknek megfelelően. |
| +Die | A meghalását valósítja meg. |
| +Explode | Az asteroida robbanáskor hívódik meg, itt kezelődik le, mi történik vele, ha a tartozkodási aszteroidára felrobban. |

### Settler

**Felelősségek**

A Worker leszármazottja, egy telepes adatai tárolására használatos osztály. A Move, Mine és Drill interfészt valósítja meg.

**Attribútumok**

|  |  |
| --- | --- |
| #madeAction: boolean | Tárolja, hogy végzett e munkát. |
| #location | A tartózkodási aszteroidát tárolja. |
| -storedItems: Material | A telepesnél lévő, már kibányászott nyersanyagok. |
| -myController: SettlerController | A settlercontrollert tárolja. |

**Metódusok**

|  |  |
| --- | --- |
| +Move(destination: Asteroid) | A kapott aszteroidára mozog. |
| +Mine() | A location aszteroidából kinyeri a nyersanyagot, a feltételeknek megfelelően. |
| +Drill() | A location aszteroidán fúr, a feltételeknek megfelelően. |
| +Die | A meghalását valósítja meg. |
| +Explode | Az asteroida robbanáskor hívódik meg, itt kezelődik le, mi történik vele, ha a tartozkodási aszteroidára felrobban. |

### Mine

**Felelősségek**

Ez egy interfész, a Mine metódust szolgáltatja.

**Attribútumok**

-

**Metódusok**

|  |  |
| --- | --- |
| +Mine() | Felüldefiniálandó. A bányászást teszi lehetővé. |

### Move

**Felelősségek**

Ez egy interfész, a Move metódust szolgáltatja.

**Attribútumok**

-

**Metódusok**

|  |  |
| --- | --- |
| +Move(destination: Asteroid) | Felüldefiniálandó. Egy adott aszteroidára való mozgást teszi lehetővé. |

### Drill

**Felelősségek**

Ez egy interfész, a Drill metódust szolgáltatja.

**Attribútumok**

-

**Metódusok**

|  |  |
| --- | --- |
| +Drill() | Felüldefiniálandó. A fúrást teszi lehetővé. |

### Material

**Felelősségek**

A nyersanyagok adatait tárolja.

**Attribútumok**

|  |  |
| --- | --- |
| -name | A neve. |
| -radiactive | Tárolja, hogy radioktív-e |

**Metódusok**

**-**

### Asteroid

**Felelősségek**

Egy aszteroidát valósít meg, kezeli a rajta lévő Workereket, robban ha kell.

**Attribútumok**

|  |  |
| --- | --- |
| -layer: int | A magját körül vevő kőzet réteg, a mag eléréséhez szükséges fúrások számát adja. |
| -deepestHole: int | Az eddig történt fúrások száma. |
| -distanceFromSun: double | A naptól való távolság. |
| -neighbours: Asteroid[1..\*] | A szomszédos aszteroidákat tárolja. |
| -workers: Worker[0..\*] | A rajta lévő workerek. |
| -mySystem: SolarSystem | A naprendszerének referenciája. |
| -core: Material | A magjának anyaga, ha null akkor üreges. |

**Metódusok**

|  |  |
| --- | --- |
| +AddNeighbour(addable: Asteroid) | A kapott aszteroidát hozzáadja a szomszédjaihoz. |
| +RmNeighbour(removable: Asteroid) | A kapott aszteroidát eltávolít a szomszédjaiból. |
| +MakeEmpty() | A közepét üregessé teszi. |
| +CoreReachable(): boolean | Visszatér, hogy a mag elérhető-e, (layer == deepestHole). |
| +Explode() | Az aszteroida felrobbanását kezeli, le, ha feltételek adottak. |
| +SetSunDistance(dis: double) | A naptól való távolságát állítja be. |
| -CheckWorkersSuccess(): boolean | Megnézi, hogy az aszteroidán lévő telepeseknél megvan-e az összes nyersanyak. |
| +AddWorker(worker: Worker) | A megadott Workert hozáadja a workers listához. |
| +RmWorker(worker: Worker) | A megadott Workert eltávolítja a workers listából. |
| +DrillOne() | Eggyel növeli a deepestHol változót. |
| +SunInDistance(): boolean | Megnézi, hogy közel van-e a nap. |
| +ApplySolarErupt() | A naprobbanás hatását szimulálja le. |

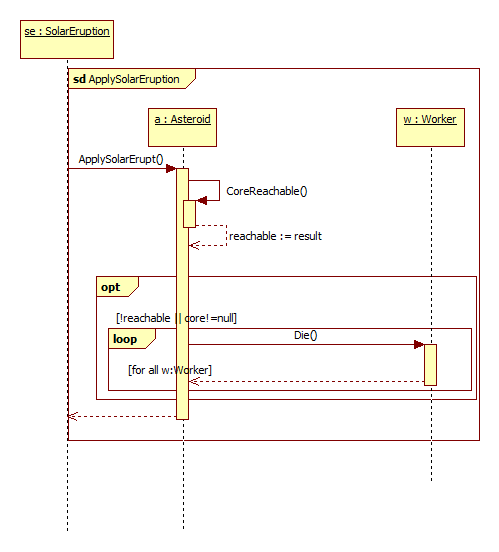
## Osztálydiagram

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

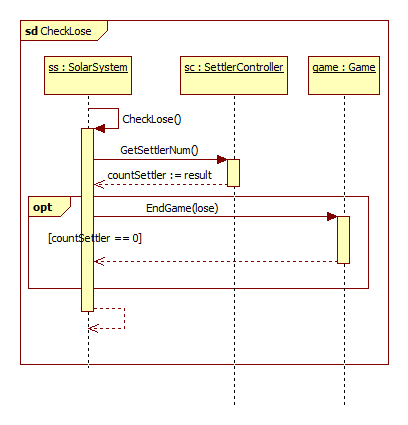
# Viselkedés leírása

## Szekvencia diagramok

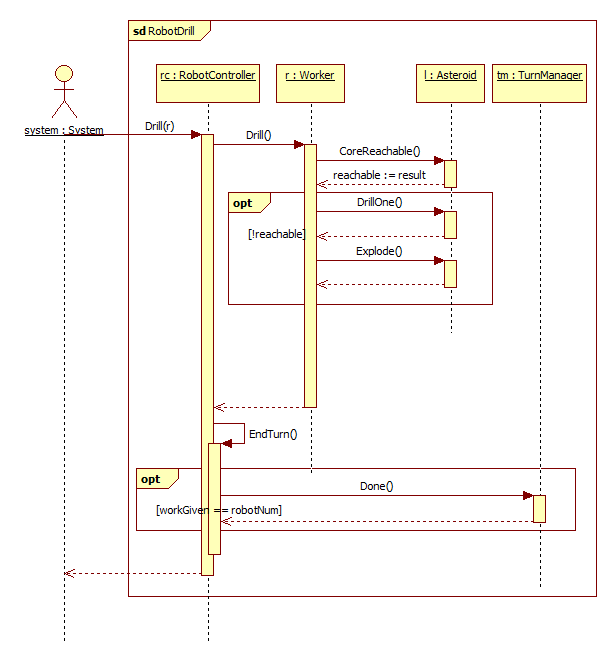
### Apply Solar Eruption



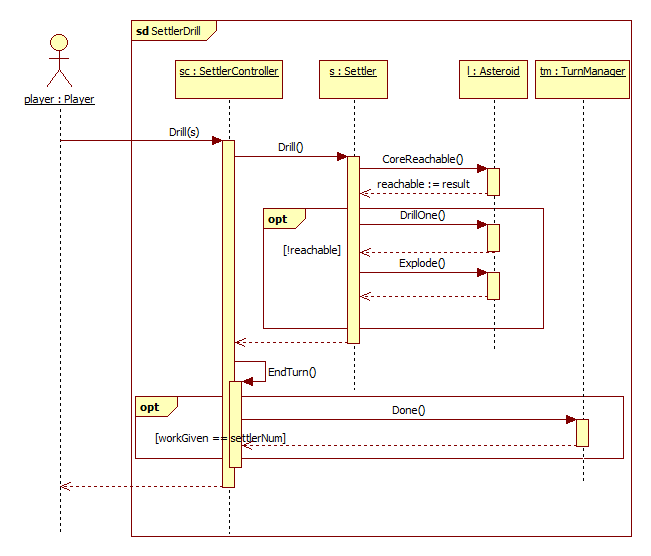
### Check Lose



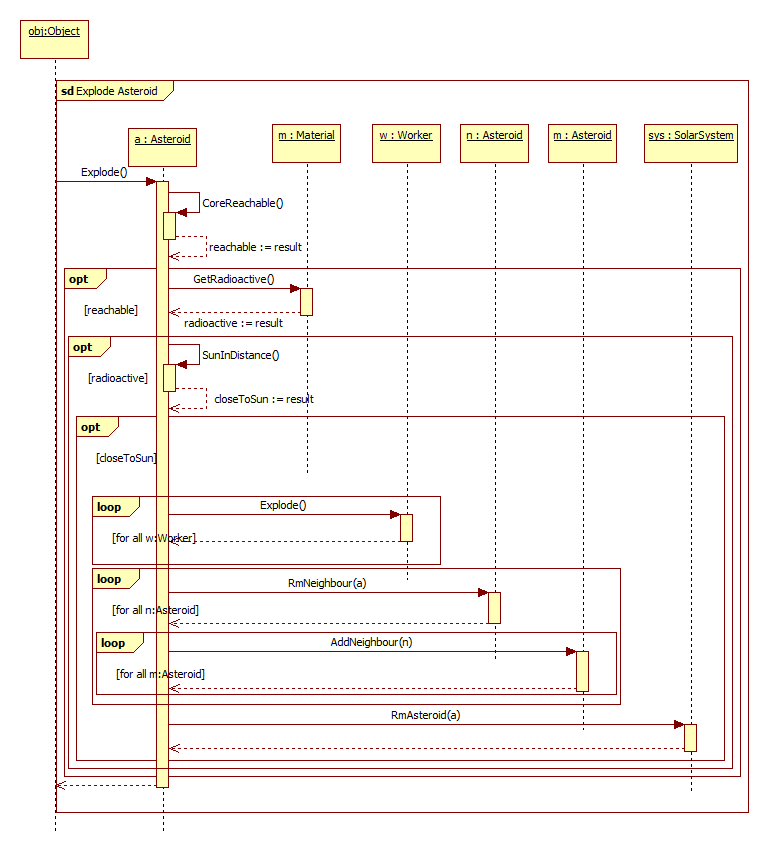
### Robot Drill



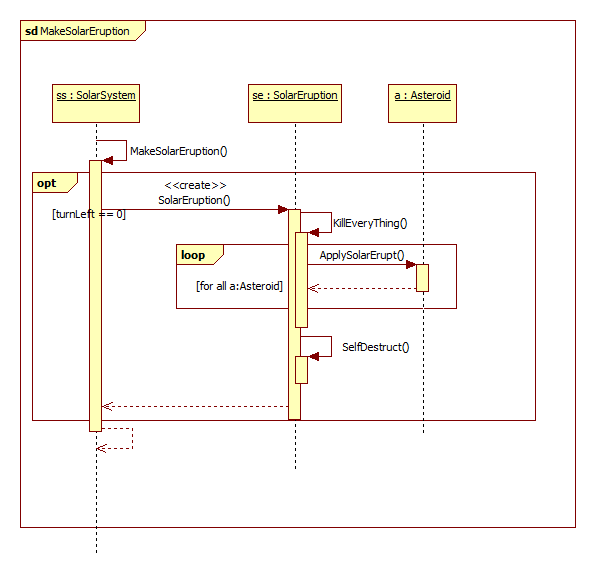
### Settler Drill



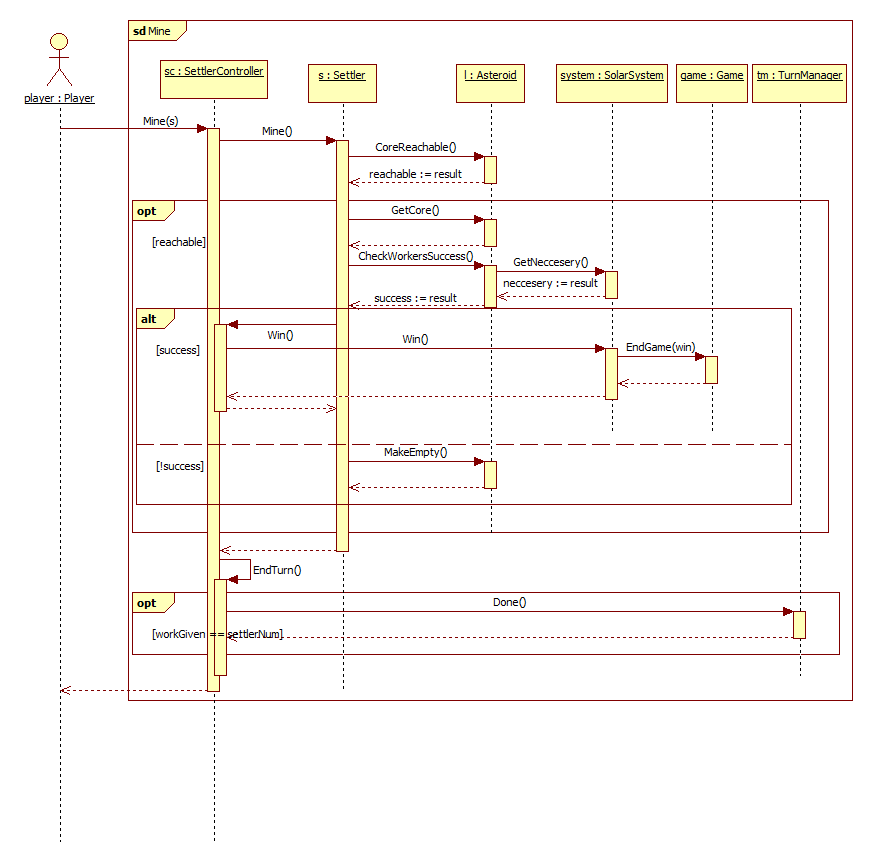
### Asteroid Explosion



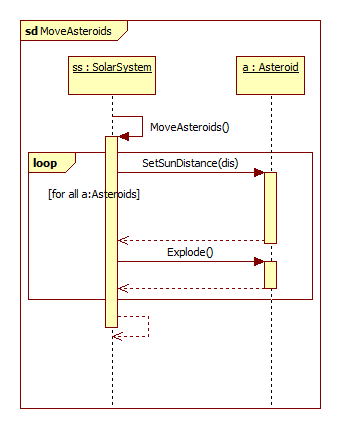
### Make Solar Eruption



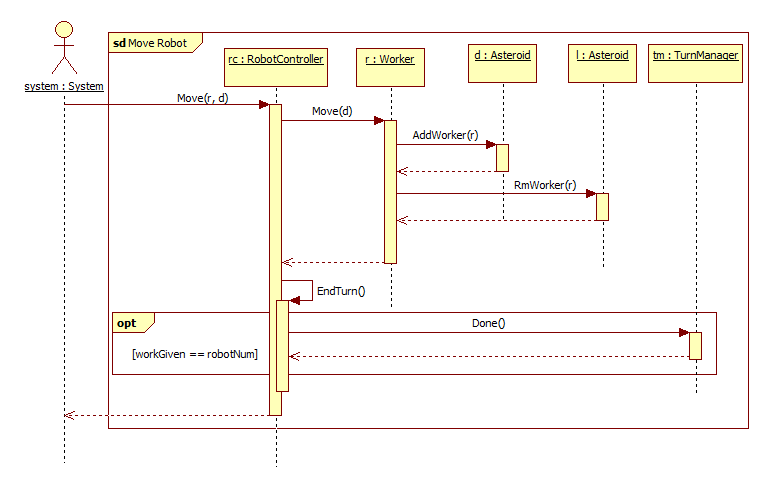
### Mining



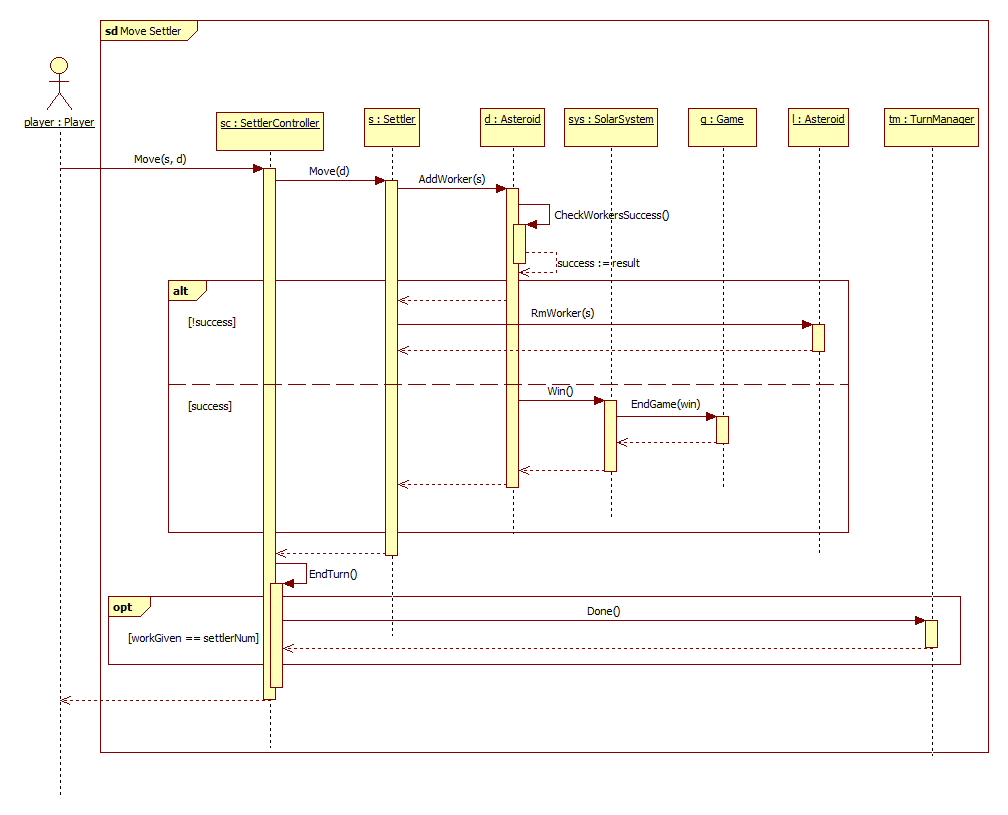
### Move Asteroids



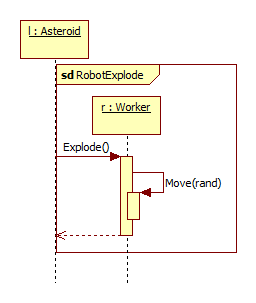
### Move Robot



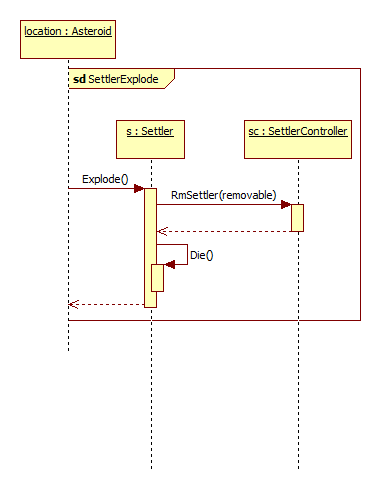
### Move Settler



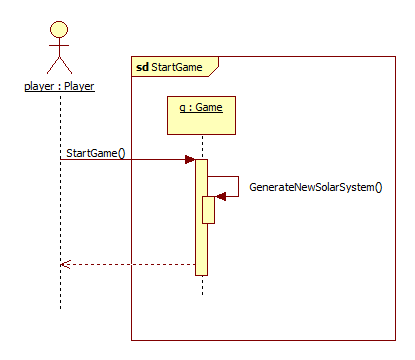
### Robot Explode



### Settler Explode



### Start Game



## Állapotgép diagramok

-

# Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Elvégzett munka** | **Hivatkozások** |
| 2020.11.03. 22:00 | 3 óra | Követelmények listájának elkészítése, use-case-ek megrajzolása és leírásának elkezdése. | 2, 3 |
| 2020.11.04 23:00 | 1 óra | Use-case-ek befejezése. | 3 |
| 2020.11.06 20:00 | 6 óra | Osztálydiagram tervezése. | 4.2 |
| 2020.11.07 22:00 | 6 óra | Szekvencia diagramok rajzolása, javítások az osztálydiagramban. | 5.1-5.6, 4.2 |
| 2020.11.08 12:00 | 4 óra | Szekvencia diagramok rajzolása, javítások az osztálydiagramban. | 5.7-5.13, 4.2 |
| 2020.11.08 18:00 | 4 óra | Az osztálydiagram dokumentálása. | 5.1 |

**Összes elvégzett munka:** 24 óra

**Modellező eszköz:** WhiteStarUML

**Egyéb eszközök:** MS World