**7. Prototípus koncepciója**

48 – modell

Konzulens:

Vörös András

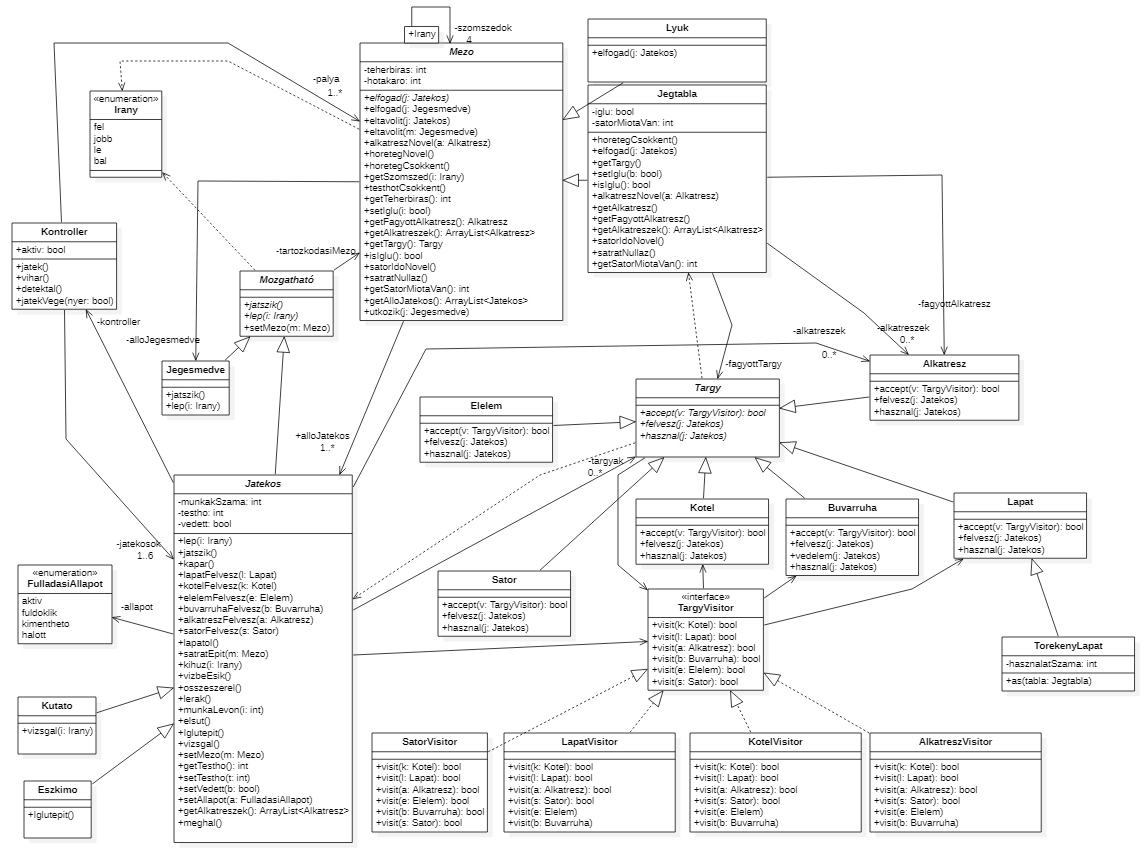
Csapattagok

|  |  |  |
| --- | --- | --- |
| Erős Enikő | PG3XAH | eros\_eniko@yahoo.com |
| Abordán Péter | YI8RGD | abordanpeter@gmail.com |
| Somogyi Bence | Q79IBL | bence.somogyi56@gmail.com |
| Csényi Lívia Ibolya | VFYMT3 | livia.csenyi@gmail.com |
| Erdei Emerencia | Z8HFEC | erdei.emerencia@gmail.com |

2020.04.06.

1. **Prototípus koncepciója**

***7.0 Változás hatása a modellre***

* + 1. **Módosult osztálydiagram**
    2. **Új vagy megváltozó metódusok**
       1. **Kontroller**
       - **Metódusok**
* **void jatek()**: A soron következő játékos jatszik() függvényét meghívja, ami által a játékos max. 4 tevékenységet végezhet. Bizonyos esetekben vihart generál. Ellenőrzéseket végez a detektál függvény segítségével. Lépteti a jegesmedvét.
* **void detektal()**: Ellenőrzi, hogy vannak-e fuldoklók, és a fuldoklásuk fázisának megfelelően beállítja a fulladási állapotuk értékét. Amennyiben a játékos állapota halott, véget vet a játéknak. Minden körben ellenőrzi, hogy semelyik alkatrész sem zuhant-e a vízbe. Ha valamelyik bezuhant, véget ér a játék. Ellenőrzi, hogy van-e kihűlt játékos. Ha igen, véget ér a játék. Megszünteti a sátrakat a mezőkön, ha lejárt az élettartamuk.
* **void vihar():** Meghívja bizonyos mezőknek a horetegNovel() függvényét, ami által a hótakaró egy véletlen számmal megnő az érintett mezőkön. Ezen kívül azon játékosok testhőjét csökkenti, akik olyan mezőn állnak, ahol nincs iglu vagy sátor.
* **void jatekVege(bool):** Ha paramétere false game over, ha true, nyert a csapat.
  + - 1. **Mozgatható**

Absztrakt osztály. Leszármazottai a Jegesmedve és a Játékos.

* + - * **Attribútumok**

**Mezo tartozkodasiMezo:** A mező amin a Mozgatható példány áll.

* + - * **Metódusok**

**abstract void jatszik():** A Mozgatható példányok implementálják.

**abstract void lep(Irany i)**: A mozgatható objektum lép.

* + - 1. **Jegesmedve**
      * **Metódusok**

**void jatszik():** Random számot generál 0 és 3 között és abba az irányba meghívja a jegesmedve lép(Irany i) metódusát.

**void lép(Irany i)**: A paraméterül kapott irányba lépteti a jegesmedvét, majd megnézi, hogy az adott jégtáblán

* + - 1. **Jatekos**
      * **Attribútumok**

**Kontroller kontroller:** A singleton kontroller osztályra referencia.

* + - * **Metódusok**

**satratEpit(m: Mezo):** Ellenőrzi, hogy van-e sátor a játékos tárgy tömbjében és ha van, meghívja a Sátor hasznal(Jatekos: j) metódusát.

**void satorFelvesz(Sator s)**: A paraméterül kapott sátrat beteszi a játékos tárgyak tömbbjébe.

* + - 1. **Mezo**
      * **Metódusok**

**void elfogad(Jegesmedve j):** Befogadja a paraméterül kapott jegesmedvét.

**void eltavolit(Jegesmedve j):** Eltávolítja az adott mezőről a paraméterül kapott jegesmedvét.

**void satorIdoNovel():** Üres metódus, a jégtábla felülírja.

**int getSatorMiotaVan():** Üres metódus, a jégtábla felülírja.

**void satratNullaz():** Üres metódus, a jégtábla felülírja.

**void utkozik(Jegesmedve j)**: Hívja az adott mezőn álló játékos meghal() függvényét.

* + - 1. **Jegtabla**
      * **Attribútumok**

**int SatorMiotaVan:** Értéke alapesetben 0, de ha a jégtáblára valaki sátrat épít értéke 1 lesz, és minden körben nő. Maximális értéke a játékosok száma, ha ezt eléri, újra 0 lesz.

* + - * **Metódusok**

**void satorIdoNovel():** Növeli a SátorMiotaVan attribútum értékét 1-gyel.

**int getSatorMiotaVan():** Visszaadja, hogy mióta áll az adott jégtáblán a sátor.

**void satratNullaz():** Nullára állítja a SatorMiotaVan attribútum értékét.

* + - 1. **Sator**
      * **Metódusok**

**void felvesz(Jatekos j):** Meghívja a paraméterül kapott játékos satorFelvesz(Sator s) függvényét this paraméterrel.

**void hasznal(Jatekos j)**: Az adott mező SatorMiotaVan értékét 1-re állítja.

* + - 1. **TorekenyLapat**
      * **Attribútumok**

**int hasznalatSzama:** 0 és 3 közötti értéket vehet fel..

* + - * **Metódusok**

**void hasznal(Jatekos j):** Ha a hasznalatSzama attribútum értéke kisebb 3-nál az adott mezőnek kétszer meghívja a horetegCsokkent() függvényét, ha a hasznalatSzama attribútum értéke kisebb 3-nál. Majd növeli a hasznalatSzama attribútum értékét 1-gyel.

* + - 1. **SatorVisitor**

A TárgyVisitor interfacet implementálja.

* + - * **Metódusok**

**boolean visit(Sator s):** truet ad vissza.

**boolean visit(Elelem s):** falset ad vissza.

**boolean visit(Buvarruha s):** falset ad vissza.

**boolean visit(Kotel s):** falset ad vissza.

**boolean visit(Lapat):** falset ad vissza.

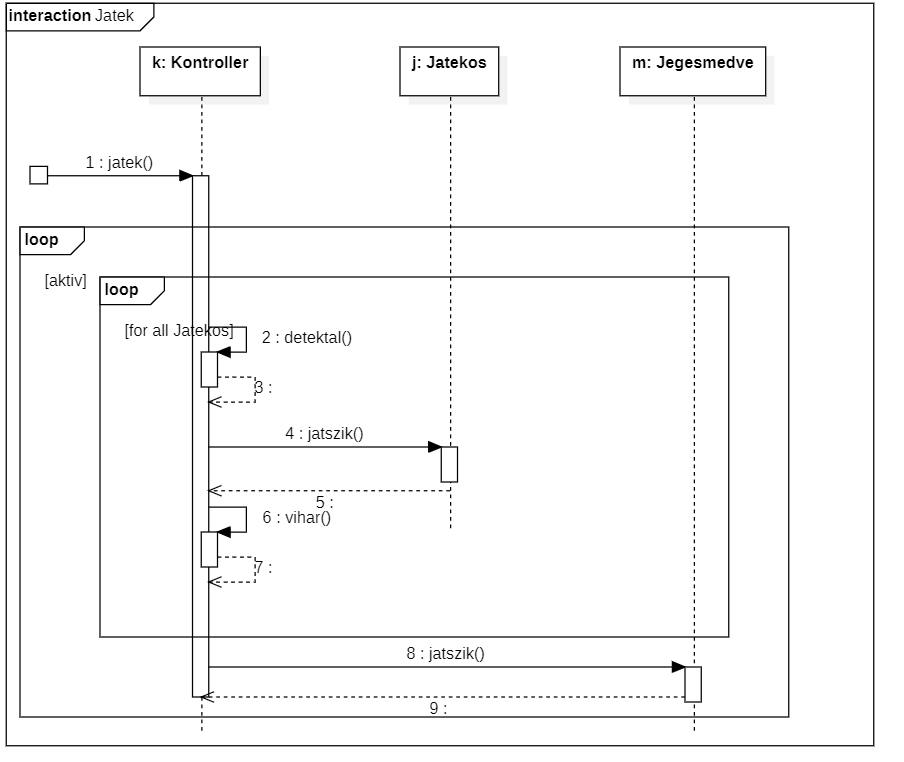
**boolean visit(Alkatresz s):** falset ad vissza.

* + - 1. **Minden Tárgy Visitor osztály leszármazottja**

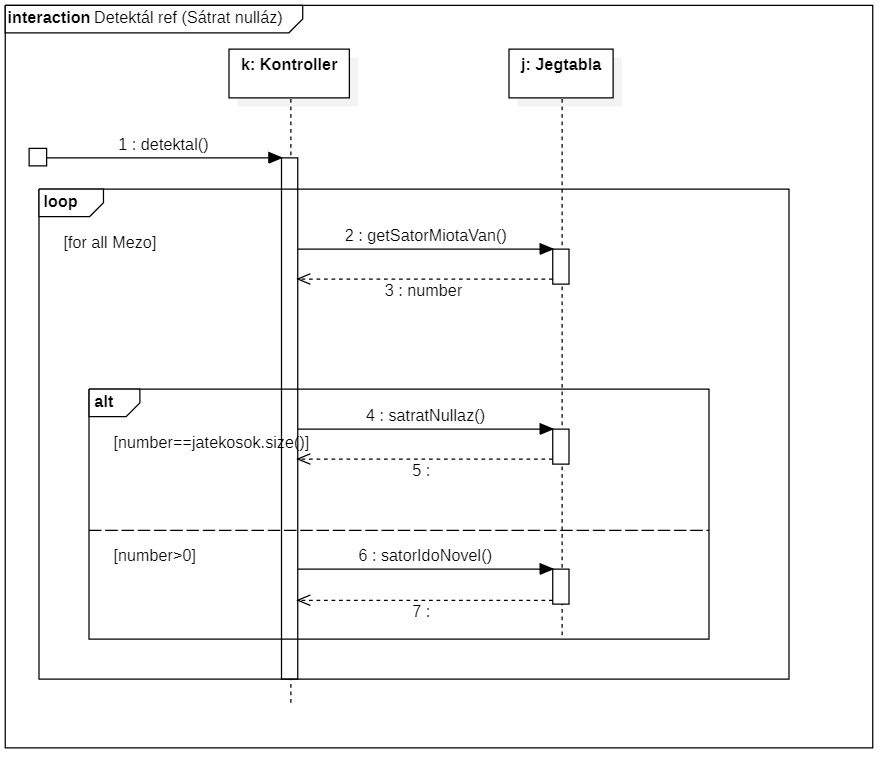
Valamennyi osztály kapott egy **visit(s: Sator): bool** metódust. Ezek közül csak a SatorVisitor osztály ezen metódusa tér vissza **true** értékkel a Visitor Design Pattern, így helyes az integráció.

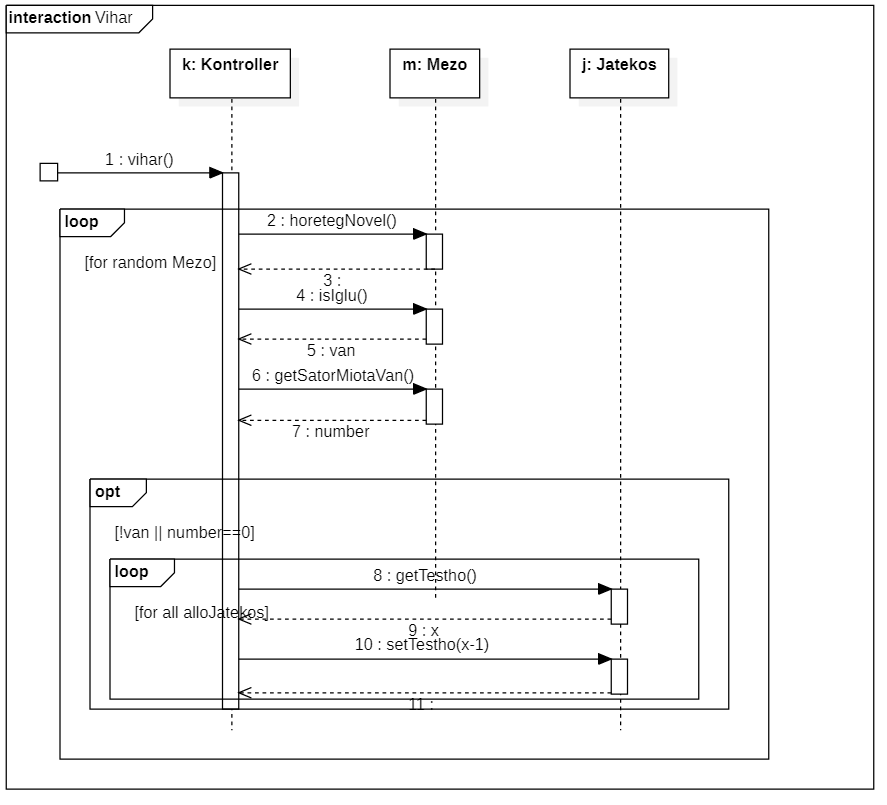
**Szekvencia-diagramok**

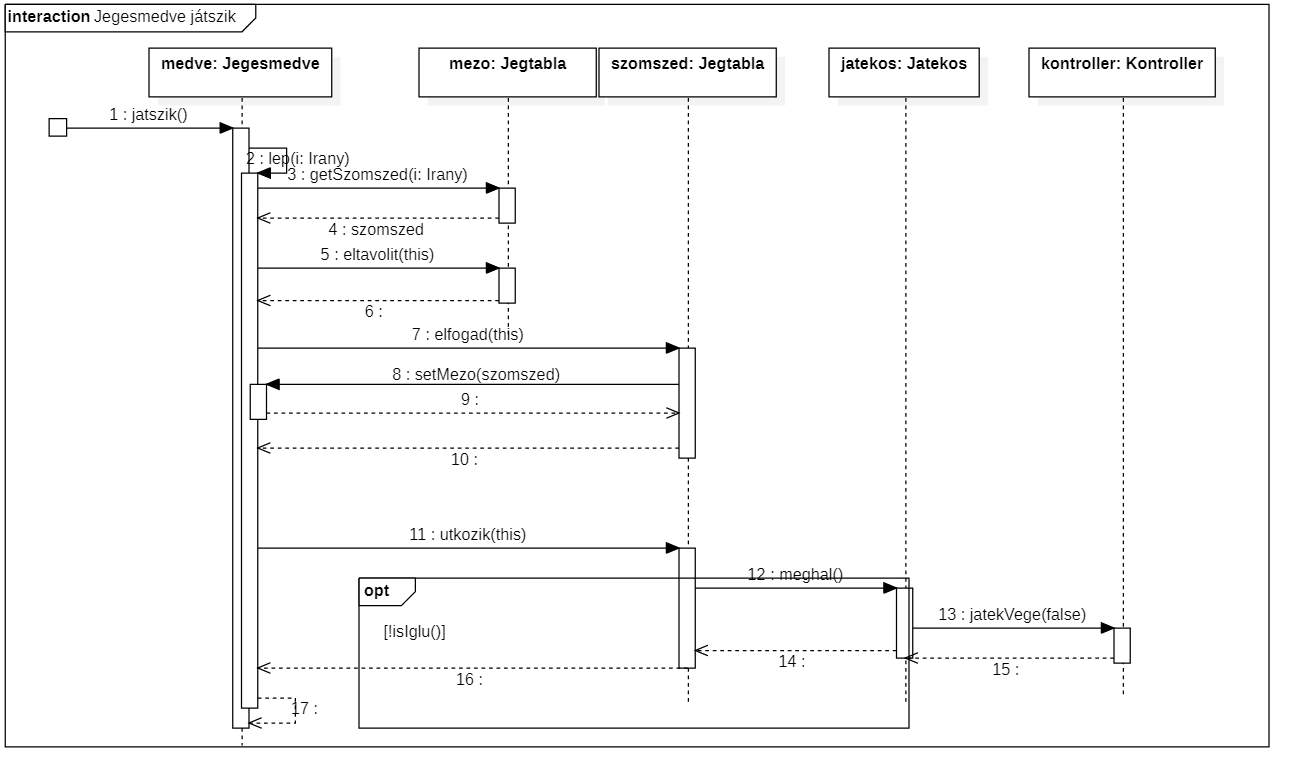
* + - 1. **Játék**

****

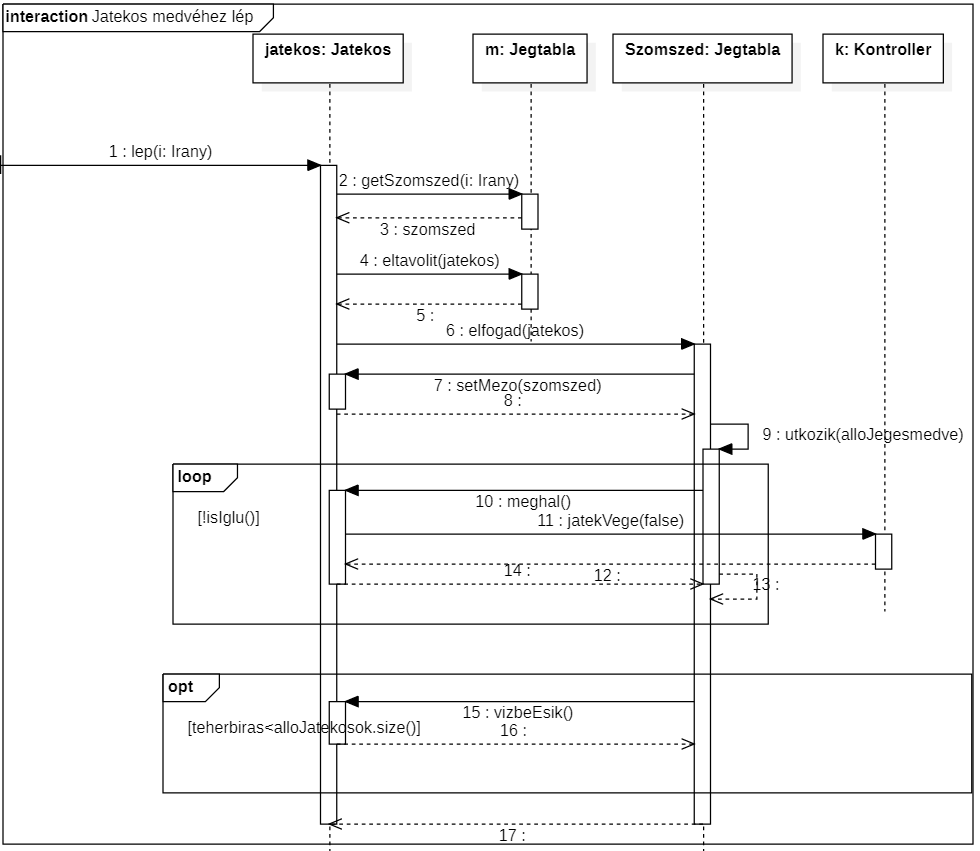
* + - 1. **Detektál (Sátrat nulláz)**

****

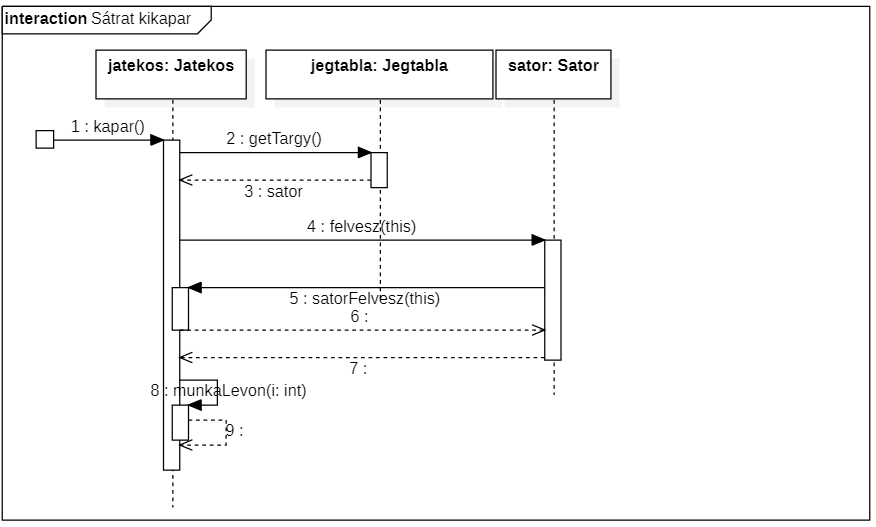
* + - 1. **Vihar**
      2. **A jegesmedve játszik**

****

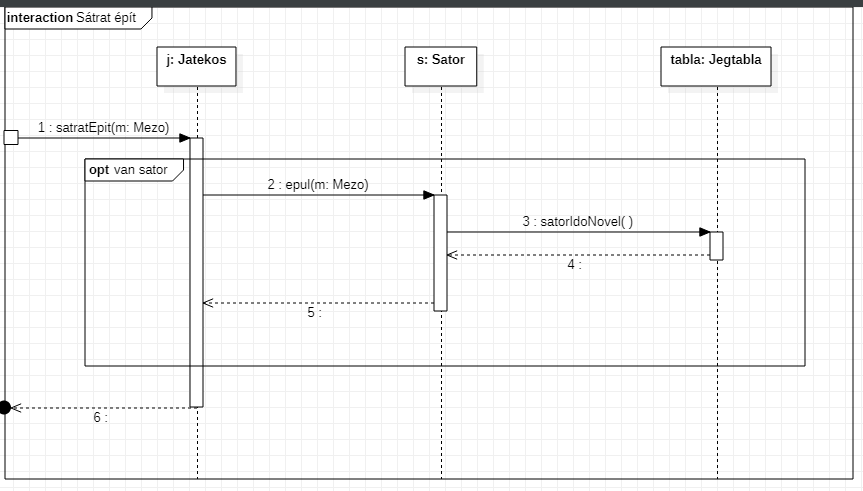
* + - 1. **A játékos lép**

****

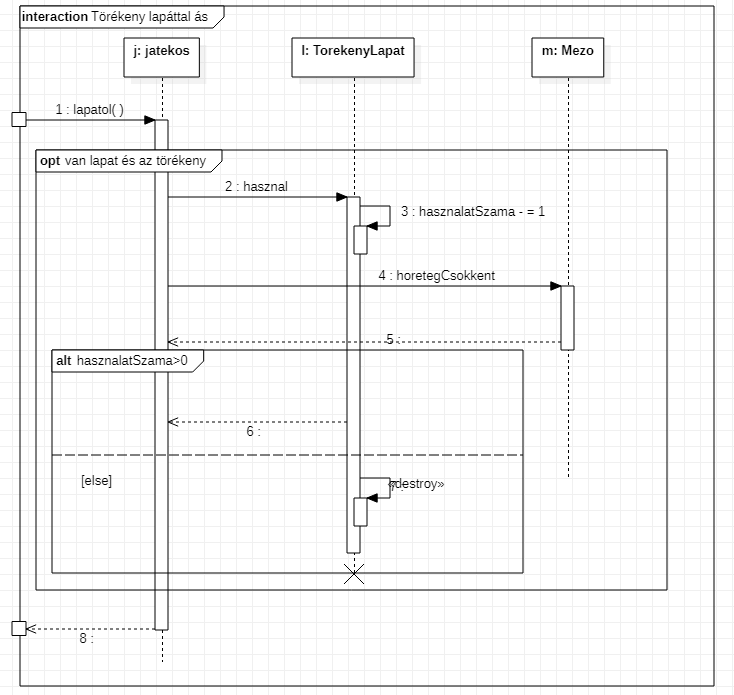
* + - 1. **Sátrat kikapar**

****

* + - 1. **Sátrat épít**

****

* + - 1. **Törékeny lapáttal ás**

****

* 1. ***Prototípus interface-definíciója***
     1. **Az interfész általános leírása**

A prototípus a parancssoron keresztül vezérelhető. A program előre elkészített fájlok alapján is tud működni, onnan olvassa be az inputot a parancssor helyett.  
A pálya mezőinek létrehozását nem támogatja a bemeneti nyelv, mindig lyukkal van körülvéve.

* + 1. **Bemeneti nyelv**

***Palya***

**Leírás:** A paraméterben megadott sor-és oszlopszámú pályát létrehozza. Beáálítja a szoszédságokat.

**Opciók:** palya <sor szám> <oszlop szám>

<m11><m12>...<>

<m21><m22>...<>

…

<mx1><mx2>...<>

pl. palya 5 4

yyyy

yjjy

yjjy

yjjy

yyyy

Az egyes betűk jelentése: y: Lyuk, j: Jegtabla.

A sorok és karakterek között nincs szóköz, a sor végén enter van.

A pálya mezőinek elrendezését a fájlból olvasáskor és konzolra íráskor is egy karakterhalmaz írja le.

***Jatekos***

**Leírás:** Létrehoz egy eszkimót vagy kutatót és adott mezőre helyezi.

**Opciók:** jatekos <név> <típus> <mező>

pl.jatekos j1 kutato m24 (a második sor negyedik oszlopában lévő mezőre teszi).

***Jegesmedve***

**Leírás:** Létrehoz egy jegesmedvét és egy mezőre helyezi.

**Opciók:** jegesmedve <név> <mező>

pl.jegesmedve j1 m24 (a második sor negyedik oszlopában lévő mezőre teszi)

***Lep***

**Leírás:** Lépteti a megadott irányba a játékost

**Opciók:** lep <név> <irány>

pl. lep j1 jobb

Lehetséges irányok: jobb, bal, fel, le.

***Lapat***

**Leírás:** Létrehoz egy lapátot a megadott névvel és leteszi a megadott mezőre.

**Opciók:** lapat <név> <mező>

pl. lapat l1 m33 (3.sor 3.oszlop)

***Kotel***

**Leírás:** Létrehoz egy kötelet a megadott névvel és leteszi a megadott mezőre.

**Opciók:** kotel <név> <mező>

pl. kotel k2 m42

***Elelem***

**Leírás:** Létrehoz egy élelmet a megadott névvel és leteszi a megadott mezőre.

**Opciók:** elelem <név> <mező>

pl.elelem e1m 43

***Buvarruha***

**Leírás:** Létrehoz egy búvárruhát a megadott névvel és leteszi a megadott mezőre.

**Opciók:** buvarruha <név> <mező>

pl.buvarruha b3 m23

***Alkatresz***

**Leírás:** Létrehoz egy alkatrészt a megadott névvel és leteszi a megadott mezőre.

**Opciók:** alkatresz <név> <mező>

pl.alkatresz a1 m41

***Script***

**Leírás:** Előre elkészített szkript. A paraméterben megkapott fájlban lévő utasításokat hajtja végre.

**Opciók:** script <fájl neve>

pl. script forgatokonyv1.txt

***Attrib***

**Leírás:** A paraméterben kapott játékos/tárgy/mező tulajdonságait írja ki.

**Opciók:** attrib <név>

pl. attrib k1, attrib m23

* + 1. **Kimeneti nyelv**

Az attrib parancs kiírja az adott játékelem tulajdonságát.

Formátum: <név>

tul1: érték

tul2: érték …

pl. j1:

mezo: m22

testho: 3

Konkrét esetek:

Jatekos:

mezo: (tartózkodási mező)

tetsho: (testhő értéke)

Jegesmedve:

mezo: (tartózkodási mező)

Jegtabla:

szfel:(a szomszédos mező fel irányban)

szjobb:

szle:

szbal:

horeteg: (hóréteg értéke)

mozgathato: (van e rajta Mozgathato objektum, azaz játékos vagy jegesmedve)

Lyuk:

szfel:(a szomszédos mező fel irányban)

szjobb:

szle:

szbal:

horeteg: (hóréteg értéke)

Lapat:

mezo: (melyik mezon van ha van)

jatekos: (melyik játékosnál van ha felvették)

Kötél:

mezo: (melyik mezon van ha van)

jatekos: (melyik játékosnál van ha felvették)

Búvárruha:

mezo: (melyik mezon van ha van)

jatekos: (melyik játékosnál van ha felvették)

Élelem:

mezo: (melyik mezon van ha van)

jatekos: (melyik játékosnál van ha felvették)

Alkatrész:

mezo: (melyik mezon van ha van)

jatekos: (melyik játékosnál van ha felvették)

* 1. ***Összes részletes use-case***

|  |  |
| --- | --- |
| **Use-case neve** | Palya |
| **Rövid leírás** | A megadott pálya létrehozása |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | 1. Mezők létrehozása 2. A mezők szomszédsági viszonyainak beállítása. |

|  |  |
| --- | --- |
| **Use-case neve** | Jatekos létrehozása |
| **Rövid leírás** | Játékos elhelyezése adott mezőn. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | 1.Játékos létrehozása  2.Játékos elhelyezése a mezőn. |

|  |  |
| --- | --- |
| **Use-case neve** | Lep |
| **Rövid leírás** | Az egyik játékos léptetése. |
| **Aktorok** | Jatekos |
| **Forgatókönyv** | 1.A játékos lép a kiválasztott irányba. |

|  |  |
| --- | --- |
| **Use-case neve** | Lapat létrehozása |
| **Rövid leírás** | Lapát letétele adott mezőre. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | 1.Lapát létrehozása  2.Lapát elhelyezése a mezőn. |

|  |  |
| --- | --- |
| **Use-case neve** | Kotel létrehozása |
| **Rövid leírás** | Kötél letétele adott mezőre. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | 1.Kötél létrehozása  2.Kötél elhelyezése a mezőn. |

|  |  |
| --- | --- |
| **Use-case neve** | Elelem létrehozása |
| **Rövid leírás** | Élelem letétele adott mezőre. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | 1.Élelem létrehozása  2.Élelem elhelyezése a mezőn. |

|  |  |
| --- | --- |
| **Use-case neve** | Buvarruha létrehozása |
| **Rövid leírás** | Búvárruha letétele adott mezőre. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | 1.Búvárruha létrehozása  2.Búvárruha elhelyezése a mezőn. |

|  |  |
| --- | --- |
| **Use-case neve** | Alkatresz létrehozása |
| **Rövid leírás** | Alkatrész letétele adott mezőre. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | 1.Alkatrész létrehozása  2.Alkatrész elhelyezése a mezőn. |

|  |  |
| --- | --- |
| **Use-case neve** | Jegesmedve lép |
| **Rövid leírás** | Jegesmedve létrehozása és a pályára tevése. |
| **Aktorok** | Kontroller |
| **Forgatókönyv** | 1.Jegesmedve létrehozása  2.Jegesmedve elhelyezése a mezőn.  3.A jegesmedve léptetése  3A. Jegesmedve a körülötte lévő mezők valamelyikére lép, ha nincs körülötte játékos.  3B. A játéknak vége, ha ott, ahová lépett, játékos van. |

* 1. ***Tesztelési terv***

|  |  |
| --- | --- |
| **Teszt-eset neve** | 1.Játékos jégtáblára lép |
| **Rövid leírás** | A játékos egy jégtábla mezőre lép adott irányban. |
| **Teszt célja** | A játékos a megfelelő mezőre kerül. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 2.Játékos lyukra lép |
| **Rövid leírás** | A játékos egy lyukra lép és beleesik. |
| **Teszt célja** | A játékos beleesik a lyukba és fuldokolni kezd. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 3. Játékos lyukra lép búvárruhával |
| **Rövid leírás** | A játékos beleesik a lyukba. |
| **Teszt célja** | A játékos beleesik a lyukba, de nem fuldoklik. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 4.Játékos havat lapátol. |
| **Rövid leírás** | A játékos a saját mezőjéről eltakarít havat. |
| **Teszt célja** | Csökken a hóréteg a mezőn. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 5.Tárgyat kikapar a játékos a mezőről. |
| **Rövid leírás** | A játékos kikaparja és felvesz egy tárgyat a mezőről. |
| **Teszt célja** | A tárgy a játékos tárgyai közé kerül. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 6.Egy játékos kihúz egy másikat. |
| **Rövid leírás** | Azt modellezzük, hogy a játékos kihúzza társát a lyukból. |
| **Teszt célja** | A fuldokló játékos kikerül a lyukból és nem fuldoklik. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 7. Játékos megfullad |
| **Rövid leírás** | Lemodellezzük azt a folyamatot, hogy egy játékos megfullad a vízben. |
| **Teszt célja** | Játékos meghal. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 8. Játékos alkatrészt lerak |
| **Rövid leírás** | A játékos lerak a mezőre egy alkatrészt. |
| **Teszt célja** | Az alkatrész a mezőre kerül. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 9.Játékos alkatrészt felvesz |
| **Rövid leírás** | A játékos felveszi a mezőről az alkatrészt. |
| **Teszt célja** | Az alkatrész a játékoshoz kerül. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 10.Eszkimó iglut épít |
| **Rövid leírás** | A játékos épít egy iglut a mezőre. |
| **Teszt célja** | Iglu épül a mezőn. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 11.Jégtábla teherbíró képességének vizsgálata |
| **Rövid leírás** | A kutató megvizsgálja a jégtábla “vastagságát”. |
| **Teszt célja** | Jégtábla vastagságának vizsgálata. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 12.A jelzőpisztoly összeszerelése |
| **Rövid leírás** | Játékos összeszereli a pisztolyt. |
| **Teszt célja** | Pisztoly összeszerelése és elsütése. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 13. Alkatrészek hiányoznak |
| **Rövid leírás** | Azt modellezzük, hogy ha valamelyik alkatrész hiányzik nem lehet összeszerelni a pisztoly, vége a játéknak. |
| **Teszt célja** | Alkatrészek hiányában vége a játéknak. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 14. Jegesmedve lép és nem ütközik senkivel |
| **Rövid leírás** | A jegesmedve lép úgy, hogy körülötte nincs játékos. |
| **Teszt célja** | A jegesmedve a megfelelő mezőre kerül. |

|  |  |
| --- | --- |
| **Teszt-eset neve** | 15. Jegesmedve lép és ütközik valakivel |
| **Rövid leírás** | A jegesmedve lép, hogy mellette minden szomszédos mezőn van egy játékos. |
| **Teszt célja** | A játéknak vége |

* 1. ***Tesztelést támogató segéd- és fordítóprogramok specifikálása***

A teszteket a JUnit keretrendszer segítségével fogjuk elvégezni, a tesztek parancssor segítségével futtathatók. pl: test <teszt függvény neve> <paraméterek>

A tesztek eredménye akár a konzolra akár egy fájlba is kiírható, ezt a tulajdonságot paraméterekkel lehet megadni.

* 1. ***Napló***

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2020.04.02 | 2 óra | Csapat | Bemutatás + Értekezlet |
| 2020.04.02 | 1.5 óra | Erős | Osztálydiagram frissítése |
| 2020.04.03 | 2 óra | Erős | Szekvenciák + új metódusok |
| 2020.04.03 15:00 | 1.5 óra | Csényi | 7.2 |
| 2020.04.04 10:00 | 2.5 óra | Erdei | 7.1, 7.2, 7.3 |
| 2020.04.04 12:00 | 1.5 óra | Abordán | Teszt nyelv kitalálása |
| 2020.04.04. 14:30 | 1.5 óra | Abordán, Somogyi | Teszt nyelv, bemeneti nyelv és kimeneti nyelvek szinkronizációjának elve, helyessége |
| 2020.04.04 16:30 | 1 óra | Somogyi | Osztálydiagram javítása, 7.0.2.17 és 7.0.2.18 |