**Analízis modell kidolgozása**

41 – maki

Konzulens:

Kovács Richárd

Csapattagok

|  |  |  |
| --- | --- | --- |
| Győrik Marcell | VW5WFI | gyorik@sch.bme.hu |
| Sárosi Dávid | E26KX5 | david.hispan@gmail.com |
| Szonda Katalin | FE17LH | katalin.szonda@gmail.com |
| Tatai Titusz Miklós | IJHLYX | tataititi2@gmail.com |

2019. 02. 26.

# Analízis modell kidolgozása

## Objektum katalógus

### Action

Interfész. A speciális Pandák viselkedését irányítja, és biztosítja ezek szabályos működését.

### Álmos

Speciális Panda. Olyan Panda, ami ha Fotel közelébe kerül, akkor lepihen egy bizonyos időre.

### Breakable

Speciális csempe, mely ha 20-szor rálépnek vagy ráugranak, összetörik. Ha összetört, akkor a rálépő állatok leesnek és meghalnak.

### Control

Vezérlő osztály. Felelőssége a játék indítása, futtatása és bezárása. Hidat képez a felhasználó és a működés között. Felügyeli a futást (betartatja a szabályokat, felügyeli a pandák mozgását).

### Csoki

Csokiautomata, speciális Field. Egy vagy több is lehet a pályán. Tulajdonsága, hogy sípol. Ezt a hangot csak az automata közvetlen közelében lehet hallani. Ennél nagyobb távolságban nincs hatása. Hatása: megijeszt bizonyos pandákat.

### Exit

Kijelölt speciális Field. Jelöli a kijáratot, ahová a játékosoknak az orángutánok segítségével el kell juttatni a pandákat.

### Field

Standard osztály, amely a pálya építőelemeit/csempéit írja le. A benne található Contain nevű attribútumon keresztül kezelhető a csempén tartózkodó karakter és környezete egymásra hatásai. Metódusai csak és kizárólag az egységbe zárhatóság miatt jelennek meg ezen a szinten.

### Fotel

Speciális Field. Ha egy Álmos Panda van a közelében, akkor belefekszik lepihenni.

### Ijedős

Speciális Panda. Ha meghallja a Játékautomata (Jatek) csilingelését megijed és elengedi sorban mögötte álló panda mancsát (továbbá a sorban következő pandákra is elengedik a tartott dolgaikat).

### Játék

Játékautomata. Speciális Field. Hasonlóan működik a csokiautomatához. Szintén csak a közvetlen közelében tartózkodó pandákra képes hatni. De nem minden (csak az Ijedős) panda ijed meg a hangtól (csilingelés).

### Orángután

A játékosok által irányított karakterek. Céljuk az összes (a lehető legtöbb) Panda elfogása. Az orángutánok csak az egyik kezükben tarthatnak pandát, ha többet szeretnének elfogni, akkor az újonnan elfogott példány a kezébe adja a sort, majd megfogja a másik kezét. Ha bármely orángután meghal a játék során, akkor a játék befejeződik.

### Panda

A gép által vezérelt karakterek. Ameddig nem fogja a kezüket semmi szabadon kószálnak a pályán. Különböző pandák különböző helyzetekre reagálhatnak előre meghatározott módon.

### Player

A játékos osztálya. Itt tárolódik a játékos pontszáma, illetve, mely orángutánt irányítja a játékos.

### Szekrény

Speciális Field. Ha egy karakter (Unit) belép a szekrénybe, akkor az azon szekrénnyel (a szekrényen belül tárolt pointer segítségével) összeköttetésben álló szekrényben fog kikötni.

### Ugrálós

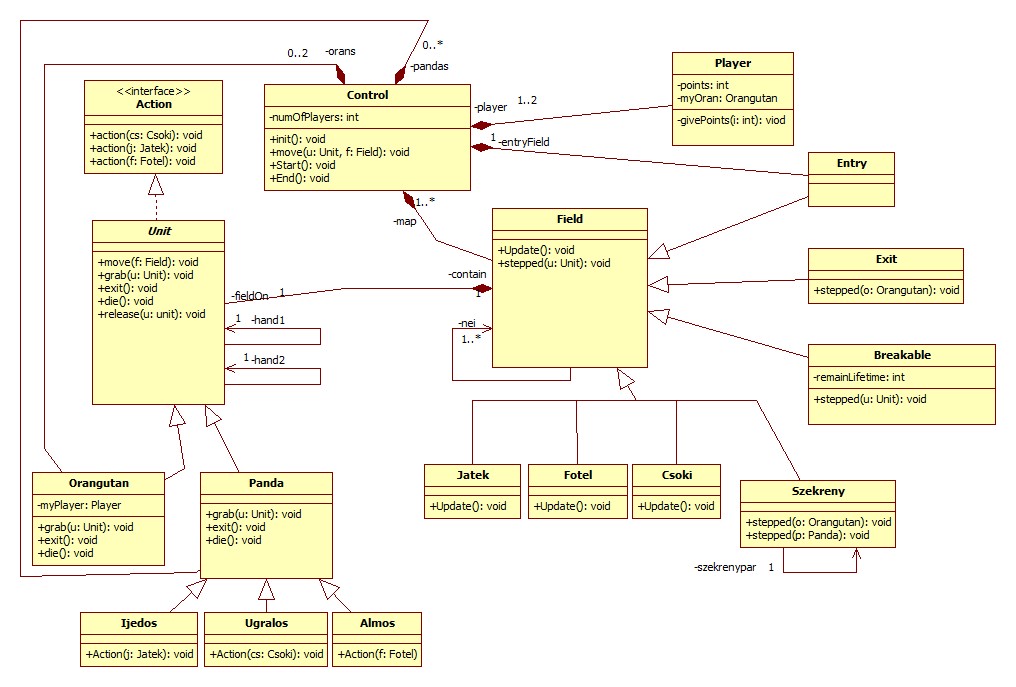
Speciális Panda. Megijed a csokiautomata sípolásától és ugrik egyet, amennyiben törékeny csempén állt, annak élettartama csökken (a rálépés mellett).

### Unit

Karakterek absztrakt ősosztálya. Egységbe zárást hivatott megvalósítani.

## Statikus struktúra diagramok

## Osztályok leírása



### Action

#### Felelősség

Az interface a tárgyak körében játszódó visitor pattern-hez kapcsolódó függvényeket foglalja össze.

#### Ősosztályok

Interface

#### Metódusok

* **void action(Csoki cs)**: csokiautomatától érkező hívást kezel
* **void action(Jatek j)**: játékgéptől érkező hívást kezel
* **void action(Fotel F):** foteltől érkező hívást kezel

### Álmos

#### Felelősség

Az osztály az álmos pandák viselkedéséért felelős

#### Ősosztályok

Object 🡪 Unit 🡪 Panda

#### Interfészek

Az osztály (közvetve) megvalósítja az Action interfészt

#### Attribútumok

**-**

#### Metódusok

* **void action(Fotel f):** Ha az álmos panda fotel közelébe lép, beleül a fotelbe
* a fentieken kívül getter és setter funkciók
* a többi metódusát a pandától örökli

### Breakable

#### Felelősség

Az osztály a törékeny csempék működését valósítja meg

#### Ősosztályok

Object 🡪 Field

#### Interfészek

Nem valósít meg interface-t

#### Attribútumok

* **int lifeTime:** A csempe élettertamát jelöli, azaz, hogy az adott csempére hányszor lehet még rálépni úgy, hogy ne essünk le.

#### Metódusok

* **void stepped(Orángután o):** Azt az setet kezeli, mikor a csempére *o* orángután rálépett
* **void stepped(Panda p):** Azt az setet kezeli, mikor a csempére *p* panda rálépett
* a fentieken kívül getter és setter funkciók
* a többi metódusát a Field osztálytól örökli

### Control

#### Felelősség

Az osztály felel a karakterek mozagtásáért, a játék inicializálásáért, inditásáért, lezárásáért. Hídként szolgál a játék és a játékosok között.

#### Ősosztályok

Object 🡪 Control

#### Interfészek

Az oszály nem valósít meg interfészt.

#### Attribútumok

* **-pandas: Panda**: a játékban szereplő Panda objektumok listája
* **-orans: Orangutan**: a játékban szereplő orángutánok listája
* **-player: Player**: a játékosok listája
* **-entryField: Field:** a bejárat referenciája
* **-map: Field:** a csempék listája

#### Metódusok

* **void move(Unit u, Field f)**: az karaktert a megadott mezőre lépteti
* **viod init()**: inicializálja a játékot
* **void start()**: a játék indítása
* **void end()**: a játék vége
* a fentieken kívül getter és setter funkciók

### Csoki

#### Felelősség

Az osztály leírja annak a csempének a működését, amelyen csokiautomata áll.

#### Ősosztályok

Object 🡪 Field

#### Interfészek

Az oszály nem valósít meg interfészt.

#### Attribútumok

#### Metódusok

* **void trigger():** A csokiautomata sípolásáért felelős függvény
* a többi metódusát a Field osztálytól örökli
* a fentieken kívül getter és setter funkciók

### Exit

#### Felelősség

Az osztály leírja annak a csempének a működését, amelyen a kijárat van.

#### Ősosztályok

Object 🡪 Field

#### Interfészek

Az oszály nem valósít meg interfészt.

#### Attribútumok

#### Metódusok

* **void stepped(o: Orangutan):** a függvény, ami meghívódik, ha egy orángután a csempére lép. Elindítja a pandasor levonulását
* **void stepped(p: Panda):** a függvény, ami meghívódik, ha egy orángután a csempére lép. Ilyenkor nem történik semmi (a panda véletlenül arra kószált).
* a többi metódusát a Field osztálytól örökli
* a fentieken kívül getter és setter funkciók

### Field

#### Felelősség

Az osztály, amely leír egy csempét.

#### Ősosztályok

Object

#### Interfészek

Az oszály nem valósít meg interfészt.

#### Attribútumok

* **-contain: Unit**: Tárolja a rajta található Unit referenciáját.
* **-nei: Field[1..\*]**: A csempe szomszédjait tárolja.

#### Metódusok

* **void trigger():** A csempe akciójáért felelős függvény (nem specializált esetben nem csinál semmit)
* **void stepped(Unit u):** A csempére lépésért felel (nem specializált esetben nem csinál semmit)

### Fotel

#### Felelősség

Az osztály leírja annak a csempének a működését, amelyen fotel áll.

#### Ősosztályok

Object 🡪 Field

#### Interfészek

Az oszály nem valósít meg interfészt.

#### Attribútumok

#### Metódusok

* **void trigger():** A függvény minden kör végén értesíti a szomszédos mezőkön állókat, hogy itt egy fotel
* a többi metódusát a Field osztálytól örökli
* a fentieken kívül getter és setter funkciók

### Ijedos

#### Felelősség

Az osztály az ijedős pandák viselkedéséért felelős

#### Ősosztályok

Object 🡪 Unit 🡪 Panda

#### Interfészek

Az osztály (közvetve) megvalósítja az Action interfészt

#### Attribútumok

**-**

#### Metódusok

* **void action(Jatek j):** Ha az ijedős panda játékautomata csilingelését hallja, megijed, és elengedi a mögötte álló panda mancsát.
* a fentieken kívül getter és setter funkciók
* a többi metódusát a pandától örökli

### Jatek

#### Felelősség

Az osztály leírja annak a csempének a működését, amelyen játékautomata áll.

#### Ősosztályok

Object 🡪 Field

#### Interfészek

Az oszály nem valósít meg interfészt.

#### Attribútumok

#### Metódusok

* **void trigger():** A föggvény a csilingelésért felelős. Mikor csilingel, értesíti a szomszédos mezőket, hogy a játékautomata csilingelt.
* a többi metódusát a Field osztálytól örökli
* a fentieken kívül getter és setter funkciók

### Orangutan

#### Felelősség

Az orángután karakterek mozgása és rá vonatkozó megkötések betartása.

#### Ősosztályok

Object 🡪 Unit

#### Interfészek

Action

#### Attribútumok

* **Hand1: Unit:** A karakter egyik kezében tartott Unitot tárolja.
* **Hand2: Unit:** A karakter másik kezében tartott Unitot tárolja.

A **Hand2** megegyezés alapján kötelezően Pandát tart, a **Hand1** kötelezően null.

#### Metódusok

* **void move(Field f):** mozgatja az orángutánt az *f*-fieldre.

Ezen kívül örökölt metódusai, valamint get-set metódusai léteznek**.**

### Panda

#### Felelősség

Az panda karakterek mozgása és rá vonatkozó megkötések betartása

#### Ősosztályok

Object 🡪 Unit

#### Interfészek

Action

#### Attribútumok

Csak örökölt attribútumai vannak.

#### Metódusok

* **void move(Field f):** mozgatja az pandát az f-fieldre.
* **void grab(Orangutan o):** *o* után beáll a sorba a panda
* **void grab(Panda p):** nem történik semmi (2 panda találkozott)
* a további metódusok örököltek
* a privát adattagok némelyikének getter és setter függvénye is van.

### Player

#### Felelősség

A játékos adatait tartja számon, továbbá biztosítja a játékos-control kommunikációt.

#### Ősosztályok

Object

#### Interfészek

#### Attribútumok

* **points:int:** A játékos pontjai.
* **myOran:** A játékos orángutánja.

#### Metódusok

* **void givePoint(int p):** *p* értékével módosítja a játékos pontjait.

### Szekrény

#### Felelősség

A szekrény típusú speciális mezőt megvalósító osztály.

#### Ősosztályok

Object 🡪 Field

#### Interfészek

Nem valósít meg interfészt.

#### Attribútumok

Csak örökölt attribútumai vannak.

#### Metódusok

* **void stepped(Orangutan o):** Meghívódik, ha Orángután lépett a mezőre.
* **void stepped(Panda p):** Meghívódik, ha Panda lépett a mezőre.

### Ugralos

#### Felelősség

Az osztály az ugrálós pandák viselkedéséért felelős

#### Ősosztályok

Object 🡪 Unit 🡪 Panda

#### Interfészek

Az osztály (közvetve) megvalósítja az Action interfészt

#### Attribútumok

**-**

#### Metódusok

* **void action(Csoki cs):** Ha az ugrálós panda csokiautomata sípolását hallja, megijed, és ugrik egyet (újra rálép arra a mezőre, amin áll).
* a fentieken kívül getter és setter funkciók
* a többi metódusát a pandától örökli

### Unit

#### Felelősség

Az osztály a pályán létező és mozgó/mozgatható karakterek egységbe zárásáért felel.

#### Ősosztályok

Object

#### Interfészek

Action

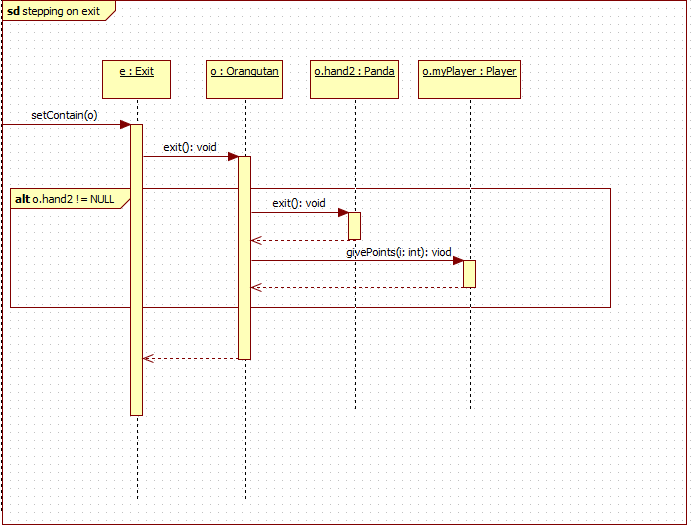
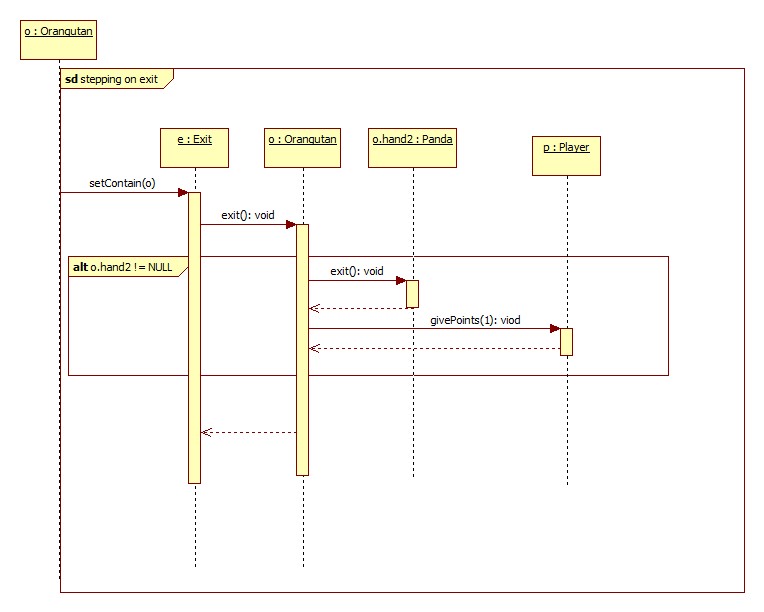
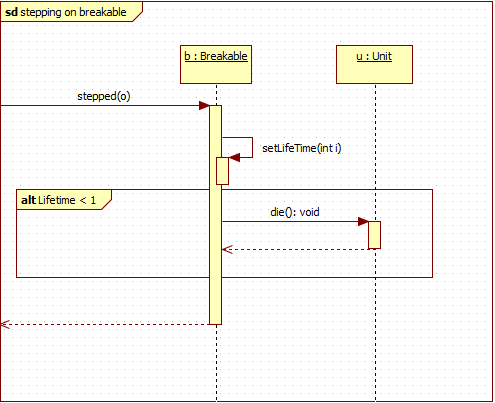
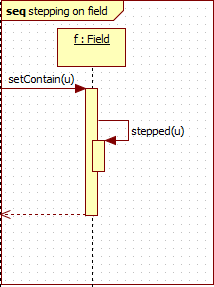
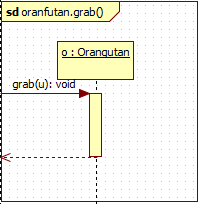
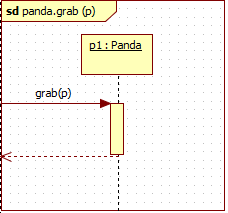
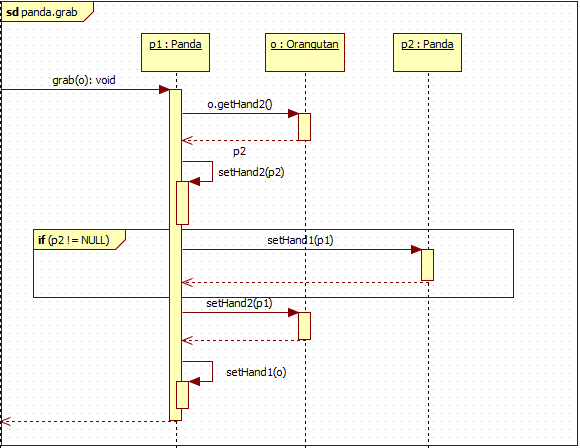
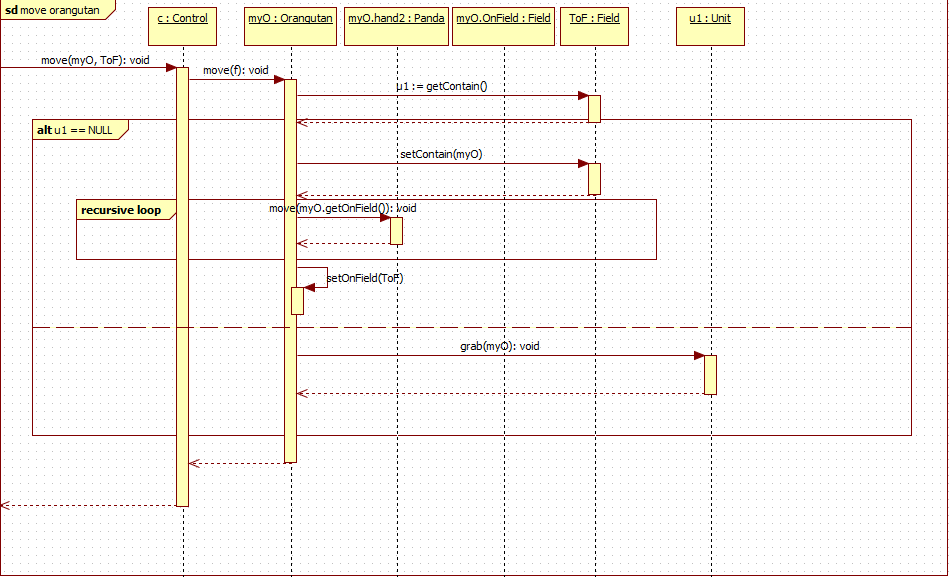
#### Attribútumok

* **Hand1: Unit:** A karakter egyik kezében tartott Unitot tárolja.
* **Hand2: Unit:** A karakter másik kezében tartott Unitot tárolja.

#### Metódusok

* **void move(Field f):** Egy megadott mezőre mozgatja a karaktert, tényleges mozgatás nem itt történik.
* **void grab(Unit u)**: a karakter megfog egy másik karaktert.
* **void exit():** akkor történik, ha egy karakter a kilépési mezőre lép.
* **void die():** a karakter destruktora.
* **release(Unit u):** elengedi a megfelelő kezéből a karaktert
* **void action(Csoki cs):** ez egy sima állat, nem csinál semmit, ha csokiautomata sípol
* **void action(Játék j):** ez egy sima állat, nem csinál semmit, ha a játékgép csilingel
* **void action(Fotel f):** ez egy sima állat, nem csinál semmit, ha a fotel jelez neki
* a fentieken kívül getter és setter funkciók

## Szekvencia diagramok



## State-chartok

## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2019. 02. 24. 21:00 | 6 óra | Győrik  Sárosi  Szonda  Tatai | Osztálydiagram elkészítése, ezzel kapcsolatos problémák felderítése, megoldása |
| 2019. 02. 25 19:00 | 5 óra | Szonda  Tatai | Osztályleírások és szekvenciák elkészítése |
| 2019 02. 26 9:00 | 3 óra | Szonda  Tatai | Egyéb szekvenciák készítése, dokumentum tisztázása |