

Pandapláza

50 – Game of Threads

Konzulens:
Máté Sebestyén

Csapattagok

Hegedüs Amadé
Bódi Dániel
Borsodi Regő
Nyári Tamás

P53ZLW amade.hegedus@gmail.com
KM3B5S bodi.daniels@gmail.com
HH5GMS rego97@gmail.com
X49AI8 tomi.nyari98@gmail.com

2019.05.15.

2. Követelmény, projekt, funkcionalitás

2.1 Bevezetés

2.1.1 Cél

A dokumentum célja a projekt követelményeinek és alapvető funkcionalitásának megfogalmazása, és előrevetíti a projekt végrehajtási tervét.

2.1.2 Szakterület

A létrehozott szoftver egy számítógépen futtatható játék, tehát szórakoztatásra való.

2.1.3 Definíciók, rövidítések

JDK 11	Java Development Kit, java programok létrehozásához szükséges csomag.
Git	Elosztott verziókezelő szoftver.
Github	Online elérhető, ingyenes git szolgáltató.
Eclipse	Fejlesztőkörnyezet, főleg Java nyelvhez.
IntelliJ	Fejlesztőkörnyezet a Java nyelvhez.
HSZK	Hallgatói Számítógép Központ
JRE	Java Runtime Environment, Java programok futtatásához szükséges környezet.
Konzol	Karakteres felület a program ki- és bemenetéhez.
Architekturális kép	Belső szerkezetet mutató ábra.
Funkció	A szoftver működésének egy külön leírható része.
Szkeleton	A program félkész állapota, csak a belső felépítés van megírva.
Prototípus	A program félkész állapota, a belső felépítés és működés van kész, a grafikus felület nincs meg.
Grafikus (átadás)	A program kész (átadható) állapota, a belső működés mellett a grafikus felület is kész van.

2.1.4 Hivatkozások

Feladatkiírás: <https://www.iit.bme.hu/targyak/BMEVIIIB02/feladat>

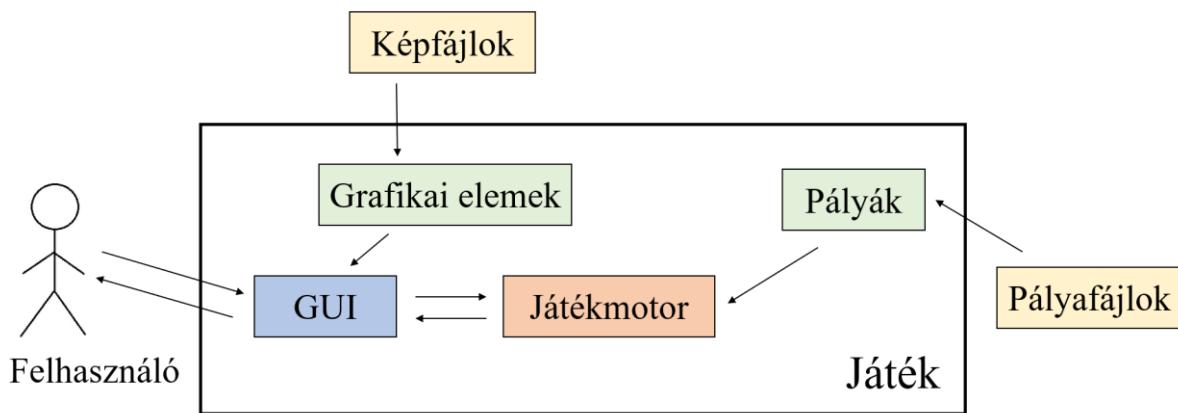
2.1.5 Összefoglalás

A dokumentum tartalmazza:

1. A szoftver áttekintése, az alapvető működésének ismertetése
2. A szoftvertől elvárt követelményeket
3. A lényeges use-case diagramokat
4. A szótárat a szavak értelmezéséhez a projekten belül
5. A projekt tervet
6. A naplót

2.2 Áttekintés

2.2.1 Általános áttekintés



Felhasználó: A játékot irányító személy. Az Ő feladata, hogy meghatározza az orángutánok lépési irányát. A végleges változatban csak a GUI-t látja. Annak segítségével kommunikál és lép interakcióba a rendszerrel. Kiemelendő, hogy a háromlépcsős bemutatás első két fázisában a felhasználó csak a konzolon keresztül tud kommunikálni a játékmotorral, de az architekturális kép elsősorban a végleges állapotot hivatott tükrözni, melyben a felhasználó már nem használja a konzolt.

GUI: Graphical User Interface, grafikus felhasználói felület. Látványos, élvezhető és egyszerű módon jeleníti meg a játékmenetet. Kapcsolatot teremt a játékmotorral és a felhasználóval. A felhasználó a GUI segítségével látja a játék alakulását és ezen keresztül tud az orángutánok számára parancsot kiadni. A harmadik fejlesztési fázis legfontosabb egysége.

Grafikai elemek: Ez az alrendszer felelős a GUI által megjelenített képfájlok beolvasásáért, objektumokhoz illesztéséért. A képeket átadja a GUI-nak megjelenítésre.

Pályák: Ez az alrendszer a pályafájlokat beolvassa, majd a játékmotor számára érhető formájúra fordítja azokat le.

Játékmotor: A játék irányítását végző legfontosabb alrendszer. Az üzleti logikát tartalmazza, a többi alrendszer segítségével kommunikál a felhasználóval, pályákat értelmez és jeleníttet meg. Végrehajtja a játék szereplőinek pozícióváltását, folyamatos monitorozást végez. Ellenőrizni, hogy a játék mikor ér véget. Felügyeli, hogy ne történjen szabálytalanság (pl. ketten lépnek egy mezőre). Szabályozza a köröket, számolja a felhasználó pontjait.

2.2.2 Funkciók

A játék egy plázában játszódik, ahol a pandák elszabadultak. Az okos gorillák feladata az, hogy a pandákat a bevásárlóközpontból kivezzék, hogy az újra alkalmassá váljon eredeti feladatának ellátására. Alapvetően ez egy egyszemélyes játék, ami bizonyos kompromisszumok mellett többszemélyes játékként is játszható. Az általános felállás az, hogy egy játékos irányítja az összes orángutánt, de lehetséges az is, hogy a játékosok a körökben egymásnak átadják az egeret és így minden orángutánnal másik személy léphet. A játék menüjében beállítható az orángutánok száma és neve, így ha valóban több játékos kíván játszani, akkor azok akár saját nevükön is hívhatják orángutánjukat.

Minden orángután épp annyi pontot kap ahány pandát kivezet a plázából. A kivezetés során azonban történhetnek katasztrófák. Ha egy csempé például egy panda alatt leszakad, akkor az a panda meghal. Ilyenkor természetesen már nem tér vissza a halott panda a plázába, azonban az orángután nem fog pontot kapni érte.

Ha a baleset egy olyan pandával történik amelyik a sor közepén van, akkor az öt követő pandák elengedik egymás kezét és a sor azon a részen felbomlik. Abban az esetben, ha egy sor közepén lévő panda elfárad és útközben a saját mezője mellett egy fotelt talál, akkor arra leülhet. A pihenő pandát követő pandák ekkor szintén elengedik egymás mancsát, ami a sor végének a felbomlásához vezet. Ilyenkor a panda egy körön át pihen a fotelben, majd újra járkálni kezd. A játék során találkozhatunk különböző zavaró tényezőkkel, mint a játékgép vagy a csokiautomata. Ezek hangjukkal megijeszthetik a félénk pandákat, így azok elengedik a többi panda mancsát játékgép esetén, vagy ugorhatnak egyet csokiautomata mellett. Az a sor amelyiket pedig nem vezeti egyik orángután sem, felbomlik.

A pláza padlója sokszögekből áll. minden sokszögről az élei mentén lehet átlépni egy másik sokszögre, így egy háromszögről háromféle irányba, egy ötszögről ötféle irányba léphetnek az orángutánok és a pandák. Ebből következik, hogy az irányítás nem billentyűzettel, hanem egérrel történik. Billentyűzet segítségével nehéz lenne egyértelműen kijelölni hatfélé iránynak gombokat.

A játékmenet váltott körökből áll. Az indítást követően minden pandának és minden orángutánnak van egy tetszőleges pozíciója, majd ezt követően a pandák és az orángutánok körei váltják egymást. Az első a pandák köre, mely során minden panda egymás után lép egyet véletlenszerűen. Azért fontos, hogy sorban, egymás után lépjenek, hogy ne alakuljon ki olyan helyzet, hogy egy mezőre többen léptek. Abban az esetben ha egy mezőn már van valami vagy valaki, akkor arra nem léphet rá más. Miután minden panda lépett egyet, az orángutánok köre következik. Ekkor az orángutánok is sorban, egymás után lépnek. Egyjátékos mód esetén egy ember lép az összessel, többjátékos mód esetén pedig az egyes játékosok lépnek saját orángutánjukkal.

A pandák kivezetése úgy történik, hogy az egyik orángután sorba megfogja a pandák kezét, majd azokból egy láncot/sort/kígyót alakít ki és ha a játékos úgy gondolja, hogy már elég sok panda van az orángutánnál, akkor elindul vele a kijárat felé. A kijárathoz vezető úton a lánc bizonyos pontokon megszakadhat, az is lehet, hogy mire az orángután eljut a kijárathoz már egyetlen panda sem lesz nála. Láncszakadás/sorbomlás a fentebb említett okok miatt következhet be. Ha egy orángután kivezette a pandákat a kijáraton, akkor egyből eltűnik onnan és a bejáratnál jelenik meg, ahonnan elindulva újabb pandákat gyűjthet. Egy orángután úgy tudja megfogni egy panda mancsát, hogy a mellette lévő mezőre lép. Ekkor automatikusan csatlakozik a panda a lánchoz.

A játék többféle módon is befejeződhet. Az első lehetőség az, hogy a pandák elfogynak, így az orángutánok nyernek, hiszen sikerült kiüríteniük a bevásárlóközpontot. A másik lehetőség, hogy az összes orángután meghal, mert mindegyik olyan csempére lépett ami gyenge volt és ezért leszakad alatta. Ebben az esetben a pandák nyernek. A harmadik lehetőség, hogy a bejárat és a kijárat között nincs út, hiszen annyi csempe leszakadt már. Ekkor a játék döntetlen, hiszen a bevásárlóközpont nem tudja majd ellátni eredeti feladatát, ha onnan nem lehet kijutni. Ekkor a pláza újjáépítése szükséges.

A játékosoknak lehetősége van arra is, hogy saját pályát készítsen. Ekkor meg kell értenie a pályafájlok struktúráját és szintaktikailag helyes pályát kell készítenie. Ha a pályafájl elkezdett, akkor azt a megfelelő helyre másolva kipróbálhatja, hogy a saját plázájából mennyire könnyen tudja kivezetni a pandákat.

2.2.3 Felhasználók

A játékprogram hat éven felüliek számára ajánlott, ezen kívül semmiféle előzetes tudásra nincs szükség. Könnyen megtanulható a kezelés és a játékmenet egyértelmű, így bárki élvezheti azt.

2.2.4 Korlátozások

A kódolásra vonatkozó korlátozás, hogy csak a standard Java könyvtárak használhatók. Továbbá fontos, hogy csak tiszta szoftvereket használunk.

P12.2.5 Feltételezések, kapcsolatok

Feladat:

<https://www.iit.bme.hu/targyak/BMEVIIIAB02/feladat>

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

2.3 Követelmények

2.3.1 Funkcionális követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Use-case
R00	Egyetlen pálya áll rendelkezésre (emelet)	bemutatás	alapvető	feladatkiírás	view map
R01	A pályát sokszög alakú csempék alkotják	bemutatás	alapvető	feladatkiírás	view map

R02	Egy csempén egyszerre csak egy állat állhat	bemutatás	fontos	feladatkiírás	view map, control orangutan, control panda
R03	Egy csempéről a szomszédos csempére csak az élek mentén lehet átmenni	bemutatás	fontos	csapat	control orangutan, control panda
R04	Egyes csempék eltörnek, ha több mint 20-szor lépnek rájuk	bemutatás	alapvető	feladatkiírás	view map
R05	A pályán Orángutánok, Pandák és tárgyak vannak	bemutatás	fontos	feladatkiírás	view map
R06	Ha egy állat egy törött csempére lép az meghal	bemutatás	fontos	feladatkiírás	view map, control orangutan, control panda
R07	A tárgyakat nem lehet a helyükről elmozdítani	bemutatás	alapvető	feladatkiírás	view map, control furniture
R08	Egy tárgy lehet szekrény, fotel, játékgép, csokiautomata	bemutatás	alapvető	feladatkiírás	view map, control furniture
R09	játékgép néha csilingel, a csokiautomata néha sípol	bemutatás	alapvető	feladatkiírás	view map, control furniture

R10	A csilingelés és a sípolás hangja csak a szomszédos csempéken hallható	bemutatás	alapvető	feladatkiírás	view map
R11	A szekrénybe bele lehet lépni és egy másik szekrényből lépünk ki	bemutatás	alapvető	feladatkiírás	view map, control orangutan, control panda, control furniture
R12	A játékos az Orángutánokat irányítja	bemutatás	fontos	feladatkiírás	control orangutan
R13	A játékos feladata a pandák elfogása	bemutatás	alapvető	feladatkiírás	view map, control orangutan
R14	A pályán van egy bejárat és egy kijárat	bemutatás	fontos	feladatkiírás	view map
R15	Ha egy Orángután a kijáraton egy csoport pandát kivezet azért pontot kap	bemutatás	fontos	feladatkiírás	view map, control orangutan, control panda, record score
R16	A kivezetés után az Orángután a bejáraton jön vissza	bemutatás	fontos	feladatkiírás	view map
R17	Minden pandának van egy különleges tulajdonsága	bemutatás	fontos	feladatkiírás	view map, control panda

R18	Van olyan panda, amelyik a játékgép csilingelésétől megijed, és elengedi a mögötte álló panda mancsát	bemutatás	fontos	feladatkiírás	view map, control panda
R19	Van, amelyik a csokiautomata sípolásától ugrik egyet, amitől a törékeny csempe élettartama csökken.	bemutatás	fontos	feladatkiírás	view map, control panda
R20	Van, amelyik a fotel mellett elmenve elfárad, és beül a fotelbe.	bemutatás	fontos	feladatkiírás	view map, control panda

2.3.2 Erőforrásokkal kapcsolatos követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
R31	A program elkészítéséhez használt nyelv: Java	bemutatás	alapvető	Tantárgyfelelős	
R32	A programkód fordítható JDK 11-en	bemutatás	fontos	Tárgyfelelős	
R33	Git, Github	nincs	fontos	csapat	verziókezelés

R34	Eclipse, IntelliJ	nincs	opcionális	csapat	Java fejlesztő környezet
R35	Számítógép és a hozzá tartozó alapvető felszerelések	nincs	fontos	csapat	Játék futtatása és az irányítás

2.3.3 Átadással kapcsolatos követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
R41	Szkeleton átadás	bemutatás	fontos	Tárgyfelelős	márc. 27
R42	Protó átadás	bemutatás	fontos	Tárgyfelelős	ápr. 24
R43	Grafikus átadás	bemutatás	fontos	Tárgyfelelős	máj. 15
R44	A program telepíthető és futtatható a HSZK gépein	bemutatás	alapvető	Tárgyfelelős	
R45	JRE 11-el rendelkező Számítógép	bemutatás	alapvető	csapat	

2.4 Lényeges use-case-ek

2.4.1 Use-case leírások

Use-case neve	Control Orangutans
Rövid leírás	A játékos irányítja egy orangután mozgását a pályán.
Aktorok	Player
Forgatókönyv	<ol style="list-style-type: none">1. A játékos kiválasztja az orangutánt.2. A játékos kiválasztja a mezőt, ahova lépjen.3.A.1. Ha a lépés szabályos az orangután lép.3.B.1. Ha a lépés szabálytalan másik mezőt kell választani..4. Az orangután mezőt vált.5. Ha van panda a mezőn, akkor az kapcsolódik a láncba.

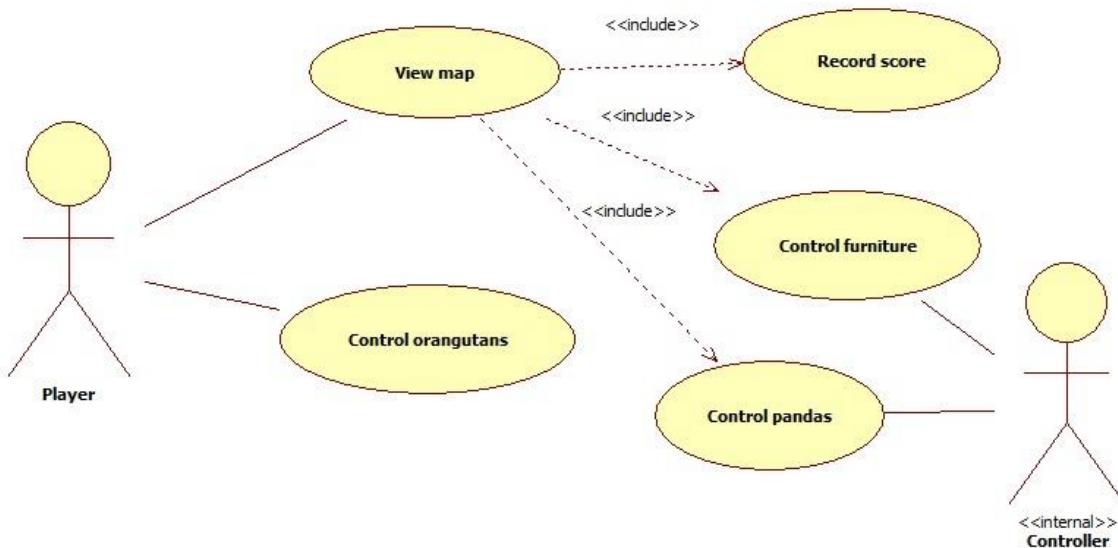
Use-case neve	View map
Rövid leírás	A játékos megtekinti a pályát.
Aktorok	Player
Forgatókönyv	<ol style="list-style-type: none">1. A rendszer kirajzolja a pálya állapotát, és a játékos pontszámát.2. A játékos megtekinti azt.

Use-case neve	Record score
Rövid leírás	A játékos pontokat szerez.
Aktorok	Player
Forgatókönyv	<ol style="list-style-type: none">1. A játékos kivezet egy orangutánt a kijáraton.2. A távozó pandák után pontot kap.

Use-case neve	Control furniture
Rövid leírás	A tárgyak időnként aktiválódnak.
Aktorok	Controller
Forgatókönyv	1.A.1. A játékgép időnként csilingel. 1.A.2. A szomszédos pandák megijednek, ha ez a képességük 1.B.1. A csokiautomata időnként sípol. 1.B.2. A szomszédos pandák megijednek, ha ez a képességük

Use-case neve	Control pandas
Rövid leírás	A pandák véletlenszerűen kószálnak.
Aktorok	Controller
Forgatókönyv	1. A játékos lép egy orangutánnal. 2. A befogott pandák követik az orangutánt. 3. A Controller lép a pályán lévő összes még nem befogott pandával véletlenszerűen. 3.A.1. Ha egy elfáradó panda fotel mellé kerül, beleül abba.

2.4.2 Use-case diagram



2.5 Szótár

A szótár a projekten belül használt szavakat definiálja az adott értelmezésben.

aktiválódás	Játékgép esetén csilingelés, Csokiautomata esetén sípolás.
állat	panda vagy orangután
beül a fotelbe	Ha a pandának az a képessége, akkor a fotel mellett elmenve elfárad és beleül. Ha láncban van, akkor elengedi a mögötte levőnek a kezét és megszakad a lánc. A következő körben feláll és megy tovább
csempe	Egy sokszög, ezekből áll a pálya
csilingelés	A játékgép néha csilingel, ami a szomszédos csempéken hallatszik.
csoport pandát kivezetni	Ha egy orangután kimegy a kijáraton úgy, hogy vezet egy lánc pandát, akkor a pandák is kimennek vele és az orangután pontot kap
él	Két csempe közös széle
elengedi a mögötte álló panda mancsát	Ha egy panda megijed vagy beül a fotelba, és láncban van, akkor a mögötte állónak elengedi a kezét és megszakad a lánc. A lánc hátsó része is felborulik.
élettartam	Egy szám, ami a gyenge csempékhez van rendelve. Azt mondja meg, hányszor lehet még rálépni. Ha 0, és valaki rálép, akkor beszakad, és meghal aki rálép
elfogni	Ha egy orangután egy panda mellé lép, akkor elfogja, a panda a lánc elejére kerül
elmozdítani	A tárgyat átvinni egy csempéről egy másikra
eltörök	Ha egy gyenge csempének elfogy az élettartama, akkor eltörök és meghal aki rálép
irányítja	Minden orangutánt egy játékos irányít. Egy játékos több orangutánt is irányíthat.
játékos	Ember, aki a játékkal játszik.
lánc	Befogott pandák sorozata, akik az orangután minden lépésére a láncban előttük álló helyére lépnek.
lép	Minden állatnak minden körben van egy lépése. Egy szomszédos mezőre léphet, ha ott nem áll senki és nincs rajta tárgy.
meghal	Ha egy állat alatt eltörök a csempe, akkor lezuhan. Nem tér vissza a játékba és kiesik.
pálya	A csempékből álló játéktér, ahol az állatok léphetnek, illetve a tárgyat vannak
szabályos (lépés)	Orangután esetén üres mező, vagy olyan mező, amin szekrény, vagy be nem fogott panda áll. Panda esetén üres mező, szekrény vagy fotel(feltéve ha a panda elfáradt).

szabálytalan (lépés)	Minden lépés ami nem szabályos.
szomszédos csempe	Ha két csempének van közös éle, akkor szomszédosak
tárgy	Fotel, játékgép, csokiautomata vagy szekrény
törékeny csempe	Bizonyos csempék törékenyek, csak 20-szor lehet rájuk lépni. Ha valaki ezután rálép, akkor meghal.
tulajdonság	Minden pandának van egy tulajdonsága. Vagy megijed a sípolástól, vagy a csilingeléstől, és ekkor ugrik egyet, vagy ha elfárad, ekkor beül a fotelbe.

2.6 Projekt terv

2.6.1 Feladatok megoldásáért felelős személyek

Csapatunk négy főből áll. A feladatokat igyekszünk közösen átbeszélni, majd egyenletesen négy felé szétosztani úgy, hogy minden csapattag olyan részfeladatot csinálhasson amiben a legtehetségesebb és amivel szívesen foglalkozik. Mivel az egyik legfőbb célunk a fejlődés, ezért az egyes feladatrészket több csapattag és átnézi, javítja, bírálja és így alakul ki a végső eredmény. A dokumentálás alapvetően közös feladat, hiszen úgy gondoljuk, hogy mindenki a saját munkáját tudja a legjobban és a legnagyobb részletességgel leírni. Az egyes feladatok elvégzéséért felelős személyek nevét egy táblázatba foglaltuk össze, de ettől bizonyos esetekben eltérünk és segítünk egymásnak.

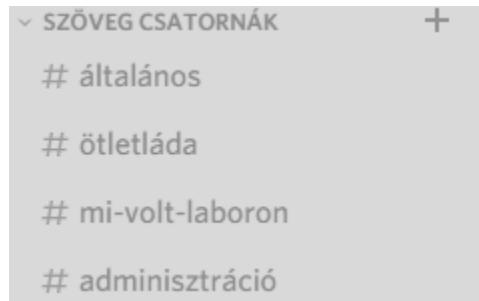
Feladatok	Felelős személyek
<ul style="list-style-type: none"> A feladatok egyenlő arányú szétosztásának ellenőrzése Értekezletek megszervezése Folyamatos kapcsolattartás a csapattagokkal A határidők betartásának ellenőrzése Egyes UML diagramok bizonyos részleteinek kidolgozása Kódolás 	Hegedüs
<ul style="list-style-type: none"> Logikai problémák részletes kidolgozása és megoldása Matematikai modellek alkalmazása a problémák megoldásához UML diagramok készítése Kódolás 	Borsodi
<ul style="list-style-type: none"> Csapatmunkát támogató eszközök folyamatos karbantartása UML diagramok készítése Kódolás Logikai problémák részletes kidolgozása és megoldása Grafikai elemek tervezése 	Bódi

<ul style="list-style-type: none"> A játék grafikus felületének elkészítése 	
<ul style="list-style-type: none"> A dokumentáció ellenőrzése, formázása A minőségi munka biztosítása Kódolás Grafikai elemek tervezése Grafikai elemek készítése A játék grafikus felületének elkészítése 	Nyári

2.6.1 Csoportmunkát támogató eszközök

2.6.1.1 Discord

“Kommunikáció strukturáltan.” Az alapvetően gamerek számára készült kommunikációs platformot használjuk az írott szövegek megosztására és a konferenciabeszélgetések lebonyolítására. A beszélgetésekhez különféle szobákat használunk, hogy bizonyos információk későbbi visszakeresése egyszerűbb legyen.



2.6.1.2 Google Docs

“Dokumentálás közösen.” Mivel mindenki a saját munkáját tudja a legjobban dokumentálni, ezért fontos volt egy olyan eszköz alkalmazása, melynek segítségével egy dokumentumon egyidőben közösen dolgozhatunk. A Google Dokumentumok az az eszköz, ahol megjegyzéseket és építő jellegű kritikákat fűzünk egymás munkájához. Valósidőben látjuk, ha egy másik személy épp dolgozik a dokumentum egyes részén, így azt is figyelemmel kísérhetjük.

2.6.1.3 Git és GitHub

“Pull és Push.” Mivel a forráskódon alapvetően minden csapattag dolgozik, ezért fontos, hogy mindenki elérje a legfrissebb projektet. A Git-et, mint verziókövető rendszert a GitHub-al, mint Remote Repository-val használjuk annak érdekében, hogy mindenki a legfrissebb verziójú forráskódon dolgozhasson.

2.6.1.4 Google Drive

“Fájlmegosztás egyszerűen”: A beadandó dokumentumok, vázlatok, tervek tárolásához egy minden csapattag által elérhető felhő alapú szolgáltatást használunk, így minden fájlt egy helyen megtalálunk.

2.6.2 Fejlesztőkörnyezet

A csoporton belül a kódoláshoz az **Eclipse 4.8.0** verzióját és az **EGit** bővítményt használjuk. Az említett plugin a Git egyszerű, gyors, grafikus felületű használatát teszi lehetővé.

2.6.3 Határidők

Határidő	Feladat	Beadás/bemutatás
2019. február 18.	Követelmény, projekt, funkcionálitás	beadás
2019. február 25.	Analízis modell kidolgozása 1.	beadás
2019. március 4.	Analízis modell kidolgozása 2.	beadás
2019. március 11.	Szkeleton terv	beadás
2019. március 25.	Szkeleton	beadás
2019. március 27.	Szkeleton	bemutatás
2019. április 1.	Prototípus koncepciója	beadás
2019. április 8.	Részletes tervezet	beadás
2019. április 23.	Prototípus	beadás
2019. április 24.	Prototípus	bemutató
2019. április 29.	Grafikus felület specifikációja	beadás
2019. május 13.	Grafikus változat	beadás
2019. május 15.	Grafikus változat	bemutató

2.6.4 A bemutatókra készített változatok funkcionálitása

2.6.4.1 Szkeleton

Az üzleti logika “csontváza”, mely elsősorban az OO dekompozíciót és az osztályok közti kapcsolatokat jelenti meg. Alkalmasnak kell lennie a use-case forgatókönyvek tesztelésére. Az osztályok metódusainak többsége nem teljesen definiált, azok főként csak a konzolra írnak ki egy mondatot és a megjelenített üzenetek sorrendjéből lehet következtetést levonni a működést illetően. Az interfések és osztályok többségét a szkeleton már tartalmazza, azonban azoknak nem a végleges változatát. Fontos, hogy itt még csak karakteres felületen léphet interakcióba a felhasználó a programmal, hiszen a grafikus felület létrehozása a harmadik, végső fázisban kerül csak elő.

2.6.4.2 Prototípus

Az üzleti logika végső változata, mely a végleges interféseket és osztályokat tartalmazza. Az összes metódus (elvileg) már definiálva van, így ez a változat már alkalmas a program helyes működésének a tesztelésére. Az üzleti logika megtervezésre került, azonban a felhasználó itt is csak a karakteres felületen keresztül léphet interakcióba a programmal.

2.6.4.3 Grafikus változat

A program végső változata, mely logikai szempontból megegyezik a prototípussal, maximum kisebb javításokat tartalmaz. Az előzőleg bemutatott változathoz képest a legnagyobb különbség a grafikus

felhasználói felület létezése, mely a program használatát élvezetessé teszi. Ez a változat már a végső leadásra készített, teljesértékű program.

2.7 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2019. február 13. 20:00	50 perc	Borsodi, Bódi, Hegedüs, Nyári	<p>Értekezlet témája: A feladat értelmezése, a dokumentum pontjainak elosztása.</p> <p>Döntés:</p> <ul style="list-style-type: none"> • 2.1, 2.5: Nyári (hi.: 02. 17. 20:00) • 2.2, 2.6: Hegedüs (hi.: 02. 17. 8:00) • 2.3: Bódi (02. 17. 08:00) • 2.4: Borsodi (02.17. 18:00)
2019. február 15. 14:30	1 óra	Bódi	2.3 szakasz elkészítése
2019. február 16. 09:00	1 óra	Borsodi	2.4 szakasz elkészítése
2019. február 16. 16:00	1 óra	Hegedüs	2.6 szakasz elkészítése
2019. február 17. 10:00	30 perc	Borsodi, Bódi, Hegedüs, Nyári	<p>Értekezlet témája: A feladat leírásában nem szereplő részletek átbeszélése.</p> <p>Döntés:</p> <ul style="list-style-type: none"> • A pályákat fájlból töltjük be • Egy játékos több orángutánt irányít • Panda és orángután is meghalhat • Ha egy sorban lévő Panda leül, akkor tőle kezdve a sor fel bomlik • Ha a szekrénybe úgy megy be egy panda vagy orángután, hogy semelyik másik szekrényből nem tud kijönni, akkor az eredeti szekrény előtt találja magát újból
2019. február 17. 12:00	1 óra	Hegedüs	2.2 szakasz elkészítése
2019. február 17. 14:30	1,5 óra	Nyári	2.1, 2.5 szakasz elkészítése
2019. február 17. 20:00	30 perc	Bódi	Dokumentum formázása, hibák javítása és kiegészítés

3. Analízis modell kidolgozása

3.1 *Objektum katalógus*

3.1.1 Panda

Egy pandát reprezentál. minden körben tud egyet lépni és a gép irányítja. Mindegyiknek van egy képessége. Háromféle képesség lehetséges. Ha a szomszédban a csokiautomata sípol ugrik egyet, vagy ha játékgép csipog, akkor elengedi a láncban mellette állók mancsát, vagy fotel mellé érve beleülhet abba. minden panda tárolja a mezőt, amin áll, és ha megfogták, az öt elfogó orángutánt és a mögötte lévő pandákat is tárolja.

3.1.2 Orangutan

Egy orángutánt valósít meg a játékban. minden körben egyet tud lépni és egy külső felhasználó irányítja. Az orángután tudja, hogy melyik mezőn áll, illetve az elfogott pandák közül az utolsót, mert az van legelöl a sorban. Ha egy Panda mellé lép, akkor azt elfogja, és a Panda a sor elejére kerül. Nyilvántartja a szerzett pontjainak számát, amit a pandák kivezetéséért kap.

3.1.3 Field

Egy csempét valósít meg. minden más objektum lehet rajta, de csak egy dolg egy időben. Lehet törékeny csempe is, ami tárolja a hátralevő lépések számát is. Ha az elfogyott, akkor betörik, és aki rálép, illetve rajta áll meghal. Feladata a szomszédos csempékkal történő kommunikáció is.

3.1.4 Entrance

A bejáratot valósítja meg, egy darab van belőle, és az orángutánok itt lépnek be a pályára miután a kijáraton kiléptek.

3.1.5 Exit

A kijáratot valósítja meg, az orángutánok ezen keresztül viszik ki a megfogott pandákat, továbbá számon tartja a bejáratot és továbbítja neki a kilépő orángutánt.

3.1.6 Sofa

Egy fotelt valósít meg. Ha egy pandának ez a képessége, és elhalad egy szomszédos mezőn, akkor beül egy körre a fotelbe, és ha már megvolt fogva, akkor a mögötte lévő kezét elengedi. Tárolja azt a mezőt, amin áll (nem tud mozogni) és ha ült benne panda, akkor azt is.

3.1.7 Chocolate Machine

Csokiautomatát valósít meg, tárolja saját mezőjét. Néha sípol és ez a szomszédos csempéken hallatszik. Ha ott olyan panda áll amely ijedős, a csokiautomata sípolásától ugrik egyet.

3.1.8 Gambling Machine

Játékgépet valósít meg, tárolja saját mezőjét, néha csilingel, ekkor értesíti a szomszédos mezőket és ha azokon áll ijedős panda, akkor az elengedi a mögötte álló kezét.

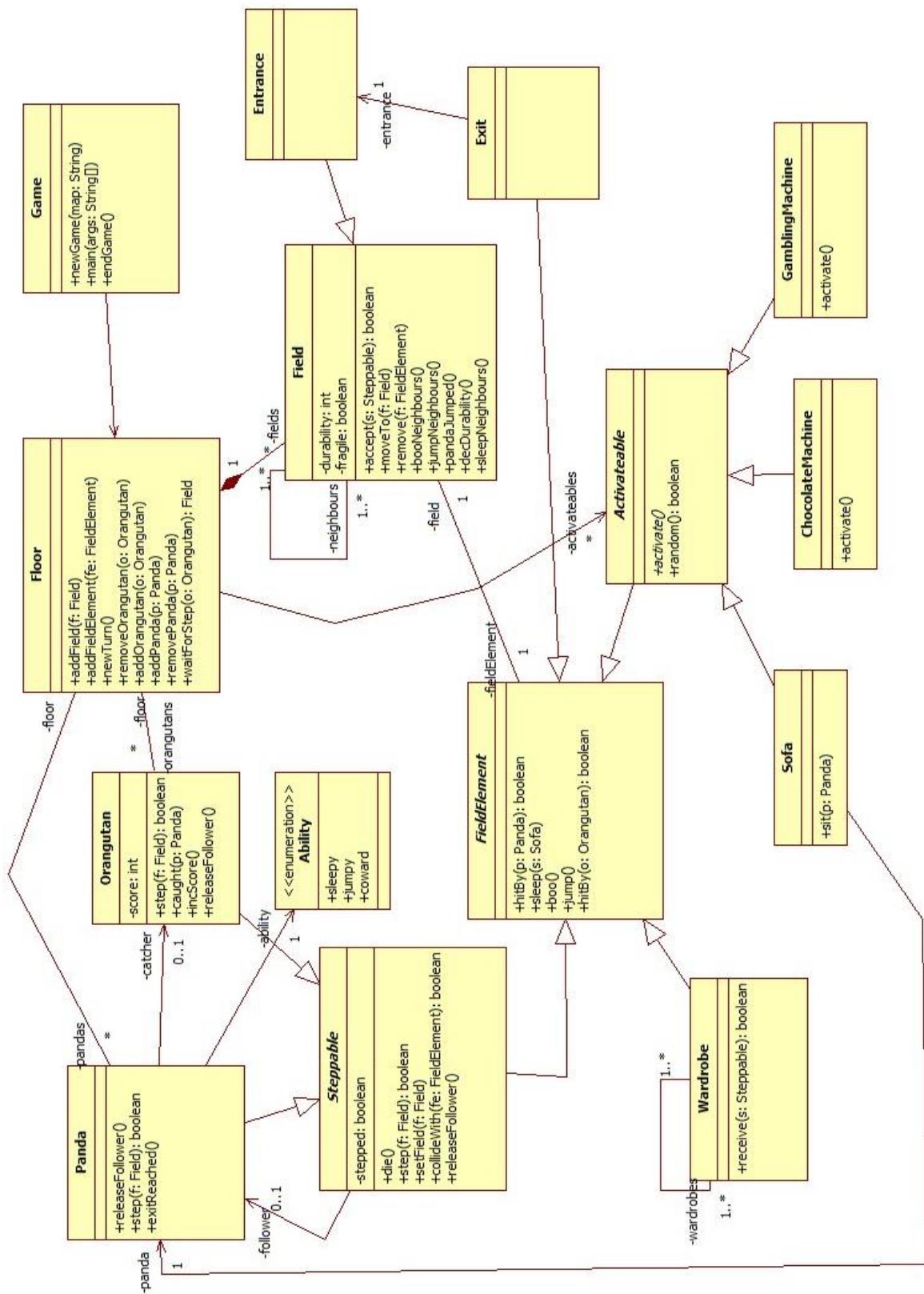
3.1.9 Wardrobe

Egy szekrényt valósít meg, tárolja saját mezőjét. Ha egy orángután belelép egy szekrénybe, akkor véletlenszerűen jön ki egy másik szekrényből. Ezáltal kapcsolatot tart a többi szekrénnel. Ha az orángután a szekrénybe pandákat vezet be, akkor a pandákkal együtt jön ki egy másik szekrényből.

3.1.10 Floor

Az emeletet (padlót) valósítja meg, tárolja a mezőket, az inicializálás során felépíti a pályát, elhelyezi a pandákat és orángutánokat, továbbá számon tartja őket. Új kör esetén értesíti a számottartott elemeket.

3.2 Statikus struktúra diagramok



3.3 Osztályok leírása

3.3.1 Ability

- Felelősség

Egy enumeráció, ami a pandák különböző képességeit tárolja. Ez lehet fáradékony, és ijedős csokiautomatától vagy játékgéptől.

- Attribútumok

- **sleepy**: a fáradékony panda képessége.
- **jmpy**: a csokiautomata hangja megijeszti az ilyen pandát, ami ugrik egyet, és csökken a csempe élettartama.
- **coward**: a játékgép hangja megijeszti az ilyen pandát, ami elengedi a mögötte álló kezét emiatt.

3.3.2 Activateable

- Felelősség

Absztrakt (ős)osztály, amely biztosítja, hogy a belőle származó osztályok minden körben aktiválhatók. Az aktiválás bizonyos speciális képességek előhívását jelenti.

- Ősosztályok

FieldElement → Activateable

- Metódusok

- **void activate()**: A speciális képesség aktiválását biztosító absztrakt függvény. minden leszármazott osztály saját maga határozza meg (definiálja), hogy az activate során milyen tevékenységet végez.
- **boolean random()**: Véletlenszerűen true-val vagy false-al visszatérő virtuális függvény. minden leszármazott saját maga dönt róla, hogy elfogadja a függvény default implementációját (50% esély a true-ra és 50% esély a false-ra), vagy felüldefiniálja.
- **boolean hitBy(Panda p)**: A függvény paraméterként kap egy Panda objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void sleep(Sofa s)**: A pihenést szimbolizáló virtuális függvény. Default implementációja üres, hiszen a pihenés egy speciális képesség, amivel csak bizonyos pandák rendelkeznek.
- **void boo()**: A megijedést szimbolizáló virtuális függvény. Default implementáció szerint nem történik semmi, ha megijed egy FieldElement. Az megijedés egy képesség, amivel csak bizonyos Pandák rendelkeznek.
- **void jump()**: Az ugrást szimbolizáló virtuális függvény. Default implementációja üres, hiszen az ugrás egy speciális képesség, amivel csak bizonyos Pandák rendelkeznek.
- **boolean hitBy(Orangutan o)**: A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.

3.3.3 ChocolateMachine

- Felelősség

Csokiautomát reprezentáló osztály, amely bizonyos speciális képességű Panda objektumokat ugraszt meg azzal, hogy néha aktiválódik és sípol.

- Ősosztályok

FieldElement → Activateable → ChocolateMachine

- Metódusok

- **void activate():** A speciális képesség aktiválását biztosító függvény. A csokiautomata a függvény meghívásának hatására sípol.
- **boolean random():** True-val vagy false-al visszatérő függvény, mely abban segít, hogy az activate függvény törzse egy előre definiált valószínűsséggel lefusson.
- **boolean hitBy(Panda p):** A függvény paraméterként kap egy Panda objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void sleep(Sofa s):** A pihenést szimbolizáló függvény. Ebben az esetben üres, hiszen a csokiautomata nem képes leülni a paraméterként kapott Sofa objektumra.
- **void boo():** A megijedést szimbolizáló függvény. Ebben az esetben üres, hiszen a csokiautomata nem képes megijedni.
- **void jump():** Az ugrást szimbolizáló függvény. Ebben az esetben üres, hiszen a csokiautomata nem képes ugrani.
- **boolean hitBy(Orangutan o):** A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.

3.3.4 Entrance

- Felelősség

Bejáratot reprezentáló osztály. Az orangutánok az Entrance osztály egy objektumán tudnak visszajönni a pályára miután a kijáraton kimentek.

- Ősosztályok

Field → Entrance

- Attribútumok

- **int durability:** Tartósság, pozitív egész számmal kifejezve. Megadja, hogy hányszor lehet rálépní a mezőre, ha a fragile attribútum értéke true.
- **boolean fragile:** A mező törékenységét jelző attribútum. Ha értéke true, akkor a mező törékeny, különben pedig nem az.
- **Field[] neighbours:** Egy mező szomszédos mezőit tartalmazó attribútum.
- **FieldElement fieldElement:** A mezőn lévő tárgyat vagy élőlényt tároló attribútum.

- Metódusok

- **boolean accept(Steppable s):** Függvény, mely a paraméterként kapott Steppable objektum sikeres fogadását jelzi. True értékkel tér vissza, ha a paraméterként kapott Steppable objektumot elhelyezte magán, egyébként pedig a visszatérési értéke false.
- **void moveTo(Field f):** A függvény a rajta álló Steppable másik mezőre való áthelyezésében segít. A sikeres lépés után "takarít".
- **void remove(FieldElement f):** A függvény a paraméterként kapott FieldElement-et letörli a mezőről ha az rajta van.
- **void booNeighbours():** A függvény a mező szomszédain lévő mezők FieldElement-jein meghívja a boo() függvényt.
- **void jumpNeighbours():** A függvény a mező szomszédain lévő mezők FieldElement-jein meghívja a jump() függvényt.
- **void pandaJumped():** Egy panda ugrásakor ez a függvény csökkenti a durability attribútum értékét egyelőre úgy, hogy a decDurability() függvényt meghívja.
- **void decDurability():** A függvény a durability attribútum értékét csökkenti egyelőre.

- **void sleepNeighbours()**: A függvény a mező szomszédain lévő mezők FieldElement-jein meghívja a sleep() függvényt.

3.3.5 Exit

- Felelősség

Kijáratot reprezentáló osztály. Az orángutánok és a pandák ezen keresztül tudják elhagyni a játékteret, majd ezt követően a bejáraton tudnak visszatérni.

- Ősosztályok

FieldElement → Exit

- Attribútumok

- **Entrance entrance**: A bejáratot tároló attribútum.

- Metódusok

- **boolean hitBy(Panda p)**: A függvény paraméterként kap egy Panda objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void sleep(Sofa s)**: A pihenést szimbolizáló függvény. Ebben az esetben üres, hiszen a kijárat nem képes leülni a paraméterként kapott Sofa objektumra.
- **void boo()**: A megijedést szimbolizáló függvény. Ebben az esetben üres, hiszen a kijárat nem képes megijedni.
- **void jump()**: Az ugrást szimbolizáló függvény. Ebben az esetben üres, hiszen a kijárat nem képes ugrani.
- **boolean hitBy(Orangutan o)**: A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.

3.3.6 Field

- Felelősség

Egy mezőt reprezentáló osztály. FieldElement-ek helyezkednek el rajta. Ezekből a mezőkből áll össze a pálya.

- Attribútumok

- **int durability**: Tartósság, pozitív egész számmal kifejezve. Megadja, hogy hányszor lehet rálépni a mezőre, ha a fragile attribútum értéke true.
- **boolean fragile**: A mező törékenységét jelző attribútum. Ha értéke true, akkor a mező törékeny, különben pedig nem az.
- **Field[] neighbours**: Egy mező szomszédos mezőit tartalmazó attribútum.
- **FieldElement fieldElement**: A mezőn lévő tárgyat vagy élőlényt tároló attribútum.

- Metódusok

- **boolean accept(Steppable s)**: Függvény, mely a paraméterként kapott Steppable objektum sikeres fogadását jelzi. True értékkel tér vissza, ha a paraméterként kapott Steppable objektumot elhelyezte magán, egyébként pedig a visszatérési értéke false.
- **void moveTo(Field f)**: A függvény a rajta álló Steppable másik mezőre való áthelyezésében segít. A sikeres lépés után "takarít".
- **void remove(FieldElement f)**: A függvény a paraméterként kapott FieldElement-et letörli a mezőről ha az rajta van.

- **void booNeighbours()**: A függvény a mező szomszédain lévő mezők FieldElement-jein meghívja a boo() függvényt.
- **void jumpNeighbours()**: A függvény a mező szomszédain lévő mezők FieldElement-jein meghívja a jump() függvényt.
- **void pandaJumped()**: Egy panda ugrásakor ez a függvény csökkenti a durability attribútum értékét egyelőre úgy, hogy a decDurability() függvényt meghívja.
- **void decDurability()**: A függvény a durability attribútum értékét csökkenti egyelőre.
- **void sleepNeighbours()**: A függvény a mező szomszédain lévő mezők FieldElement-jein meghívja a sleep() függvényt.

3.3.7 FieldElement

- Felelősség

A mezőn lévő elemek absztrakt ősosztálya. Biztosítja, hogy minden mezőre helyezhető elem rendelkezzen a benne lévő metódusokkal.

- Attribútumok

Field field: Attribútum, ami azt a mezőt tárolja, melyen az adott FieldElement elhelyezkedik.

- Metódusok

- **void activate()**: A speciális képesség aktiválását biztosító **absztrakt** függvény. minden leszármazott osztály saját maga határozza meg, hogy az activate függvényében mit kell csinálnia.
- **boolean random()**: Véletlenszerűen true-val vagy false-al visszatérő **virtuális** függvény. minden leszármazott saját maga dönt róla, hogy elfogadja a függvény default implementációját (50% esély a true-ra és 50% esély a false-ra), vagy felüldefiniálja.
- **boolean hitBy(Panda p)**: A függvény paraméterként kap egy Panda objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void sleep(Sofa s)**: A pihenést szimbolizáló **virtuális** függvény. Default implementációja üres, hiszen a pihenés egy speciális képesség, amivel csak bizonyos pandák rendelkeznek.
- **void boo()**: A megijedést szimbolizáló **virtuális** függvény. Default implementáció szerint nem történik semmi, ha megijed egy FieldElement. Az megijedés egy képesség, amivel csak bizonyos Pandák rendelkeznek.
- **void jump()**: Az ugrást szimbolizáló **virtuális** függvény. Default implementációja üres, hiszen az ugrás egy speciális képesség, amivel csak bizonyos Pandák rendelkeznek.
- **boolean hitBy(Orangutan o)**: A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.

3.3.8 Floor

- Felelősség

A játékmény vezérlését végző osztály. Tárolja a mezőket, tárgyakat, az orangutánokat és a pandákat. Új kört indít, törlő, feldolgozza a felhasználó parancsát.

- Attribútumok

- **Orangutan[] orangutans**: Az orangutánokat tároló attribútum.
- **Panda[] pandas**: A pandákat tároló attribútum.
- **Activateable[] activateables**: Az aktiválható tárgyakat tároló attribútum.
- **Field[] fields**: A mezőket tároló attribútum.

- **Metódusok**

- **void addField(Field f):** A függvény segítségével egy mező adható hozzá a padló Field objektumokból áll.
- **void addFieldElement(FieldElement fe):** A függvény egy FieldElement-et ad hozzá véletlenszerűen valamelyik mezőhöz, az inicializálás során van rá szükség.
- **void newTurn():** A függvény segítségével új kör indítható, mely következtében a játék sorban lépet minden léptethető objektumot sorban.
- **void removeOrangutan(Orangutan o):** A függvény a paraméterként kapott Orangutan objektumot törli az orangutans-ból.
- **void addOrangutan(Orangutan o):** A függvénynel egy új Orangutan objektum adható az orangutans-hoz.
- **void removePanda(Panda p):** A függvény a paraméterként kapott Panda objektumot törli a pandas-ból.
- **void addPanda(Panda p):** A függvénynel egy új Panda objektum adható az pandas-hoz..
- **Field waitForStep(Orangutan o):** A függvény a paraméterként kapott orangután lépéseknek célmezőjére vár, melyet a felhasználó felület fog megadni, visszatér ezzel a mezővel.

3.3.9 GamblingMachine

- **Felelősség**

Játékgépet reprezentáló osztály, amely bizonyos speciális képességű Panda objektumokat ijeszt meg azzal, hogy néha aktiválódik és csilingel.

- **Ősosztályok**

FieldElement → Activateable → GamblingMachine

- **Metódusok**

- **void activate():** A speciális képesség aktiválását biztosító függvény. A játékgép a függvény meghívásának hatására csilingel.
- **boolean random():** True-val vagy false-al visszatérő függvény, mely abban segít, hogy az activate függvény törzse egy előre definiált valószínűsséggel lefusson.
- **boolean hitBy(Panda p):** A függvény paraméterként kap egy Panda objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void sleep(Sofa s):** A pihenést szimbolizáló függvény. Ebben az esetben üres, hiszen a játékgép nem képes letölni a paraméterként kapott Sofa objektumra.
- **void boo():** A megijedést szimbolizáló függvény. Ebben az esetben üres, hiszen a játékgép nem képes megijedni.
- **void jump():** Az ugrást szimbolizáló függvény. Ebben az esetben üres, hiszen a játékgép nem képes ugrani.
- **boolean hitBy(Orangutan o):** A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.

3.3.10 Game

- **Felelősség**

A játékot irányító osztály, felelős a játék elindításáért és befejezéséért.

- **Metódusok**

- **newGame(String map):** A paraméterben kapott pályán elkezd egy új játékot.
- **main(String[] args):** Ez a metódus fog először meghívódni a játék során, ez felelős a szoftver indulásáért.

- **endGame():** Ezzel a metódussal befejezi a játékot.

3.3.11 Orangutan

- Felelősség

Egy orángutánnak feleltethető meg, annak megvalósításáért felel.

- Ősosztályok

FieldElement → Steppable → Orangutan

- Attribútumok

- **int score:** az orángután pontjait tárolja
- **boolean stepped:** true, ha már lépett a körben az orángután

- Metódusok

- **boolean step(Field f): Lépést indít az f Field felé.**
- **void caught(Panda p):** Elkappa a p pandát, és megfogja annak kezét, illetve áthelyezesi saját magát és pandát is.
- **void incScore():** növeli a pontjainak a számát 1-el.
- **void releaseFollower():** Elengedi a mögötte álló panda kezét.
- **void die():** Az orángután meghal, mert beszakadt alatta a csempé.
- **void setField(Field f):** Átállítja az aktuális mezőt a paraméterben kapotttra, ha sikeres volt a lépés.
- **boolean collideWith(FieldElement fe):** Ez a metódus hívódik meg, ha ütközik az fe FieldElementtel. Visszatér az ütközés kimenetelével.
- **boolean hitBy(Panda p):** A függvény paraméterként kap egy Panda objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void sleep(Sofa s):** Örökolt függvény, nem csinál semmit, mert csak a fáradékony pandák ülnek le.
- **void boo():** Örökolt függvény, nem csinál semmit, mert csak az ijedős pandát ijednek meg.
- **void jump():** Örökolt függvény, nem csinál semmit, mert csak az ijedős pandát ijednek meg.
- **boolean hitBy(Orangutan o):** A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.

3.3.12 Panda

- Felelősség

Egy pandát reprezentáló osztály.

- Ősosztályok

FieldElement → Steppable → Panda

- Attribútumok

- **boolean stepped:** True, ha már lépett a körben a panda.

- Metódusok

- **void exitReached():** Ha a pandát vezető orángután elérte a kijáratot, akkor az meghívja a következő panda exitReached függvényét, ami meghívja a lánc következő tagját, így véigmegy az egész láncon. Kitörli magát a panda, és pontot ad az orángutánnak.
- **void releaseFollower():** Elengedi a mögötte álló panda kezét.

- **void die():** Az orangután meghal, mert beszakadt alatta a csempe.
- **void setField(Field f):** Átállítja az aktuális mezőt a paraméterben kapotttra, ha sikeres volt a lépés.
- **boolean collideWith(FieldElement fe):** Ez a metódus hívódik meg, ha ütközik az fe FieldElementtel. Visszatér az ütközés kimenetelével.
- **boolean step(Field f):** Lépést indít az f Field felé. Sikeres lépés esetén true-val tér vissza, egyébként pedig false-al.
- **boolean hitBy(Panda p):** A függvény paraméterként kap egy Panda objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void sleep(Sofa s):** Ha a panda fáradékony, és meghívódik rajta ez a metódus, akkor a paraméterben kapott fotelben fog aludni egy körig.
- **void boo():** Ha a panda ijedős a játékgép hangjától, akkor elengedi a mögötte álló kezét.
- **void jump():** Ha a szomszédos mezőn aktiválódik a csokiautomata, akkor hívódik meg a metódus. Ha a panda ijedős a csokiautomata hangjától, akkor ugrik egyet, és csökken a csempe élettartama egyel.
- **boolean hitBy(Orangutan o):** A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.

3.3.13 Sofa

- Felelősség

Egy fotel működését valósítja meg. Ha egy fáradékony panda elhalad a szomszédos mezőn, akkor beleül egy körre.

- Ősosztályok

FieldElement → Activateable → Sofa

- Attribútumok

- **Field field:** A mező, amin áll a fotel
- **Panda panda:** Ha ül panda a fotelben, akkor azt itt tárolja, mert a panda nem állhat a mezőn, hisz ott van a fotel.

- Metódusok

- **void activate():** Aktiválja a fotelt minden körben, ha ül benne panda, akkor az feláll és elmegy, ha nem ül benne, akkor viszont a szomszédos csempéken állóknak meghívja a sleep() függvényét.
- **boolean random():** Örökolt függvény, nincs jelentősége, mert minden körben aktiválódik a fotel.
- **boolean hitBy(Panda p):** A függvény paraméterként kap egy Panda objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void sleep(Sofa s):** Örökolt függvény, de nem csinál semmit, mert csak bizonyos pandákra van hatással.
- **void boo():** Örökolt függvény, de nem csinál semmit, mert csak bizonyos pandákra van hatással.
- **void jump():** Örökolt függvény, de nem csinál semmit, mert csak bizonyos pandákra van hatással.
- **boolean hitBy(Orangutan o):** A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.

3.3.14 Steppable

- Felelősség

Egy absztrakt osztály, ami minden olyan tárgynak az ōsosztálya, ami mozgatható a pályán.

- Ősosztályok

Field → FieldElement

- Attribútumok

- **Field field:** A mező, amin éppen áll az objektum
- **boolean Stepped:** Tárolja, hogy lépett e már a körben az objektum.
- **Panda follower:** Ez a mögötte levő panda, akinek fogja a kezét. Ha nincs mögötte panda, akkor üres.

- Metódusok

- **boolean hitBy(Panda p):** Akkor hívódik meg, ha egy panda akar lépni arra a helyre, ahol áll.
- **void sleep(Sofa s):** A pihenést szimbolizáló virtuális függvény. Default implementációja üres, hiszen a pihenés egy speciális képesség, amivel csak bizonyos pandák rendelkeznek.
- **void boo():** A megijedést szimbolizáló virtuális függvény. Default implementáció szerint nem történik semmi, ha megijed egy FieldElement. Az megijedés egy képesség, amivel csak bizonyos Pandák rendelkeznek.
- **void jump():** Az ugrást szimbolizáló virtuális függvény. Default implementációja üres, hiszen az ugrás egy speciális képesség, amivel csak bizonyos Pandák rendelkeznek.
- **boolean hitBy(Orangutan o):** A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void die():** Meghalást szimbolizáló függvény, ha meghívódik, az objektum kiesik a játékból.
- **boolean step(Field f):** Lépést kezdeményez a paraméterben kapott mező felé. Ha sikeres akkor true, ha sikertelen, akkor false a visszatérési érték.
- **void setField(Field f):** Átállítja az aktuális mezőt, amin áll, ha helyváltoztatás van.
- **boolean collideWith(FieldElement fe):** Ez a metódus hívódik meg, ha ütközik az fe FieldElementtel. Visszatér az ütközés kimenetelével.
- **void releaseFollower():** Valamiért megbomlik a lánc, és ezért elengedi a mögötte álló kezét, a lánc hátralevő része felbomlik.

3.3.15 Wardrobe

- Felelősség

Egy szekrényt valósít meg. Ha egy Steppable belelép egybe, akkor egy másik szekrényből bukkan elő, ez véletlenszerűen történik.

- Ősosztályok

FieldElement → Wardrobe

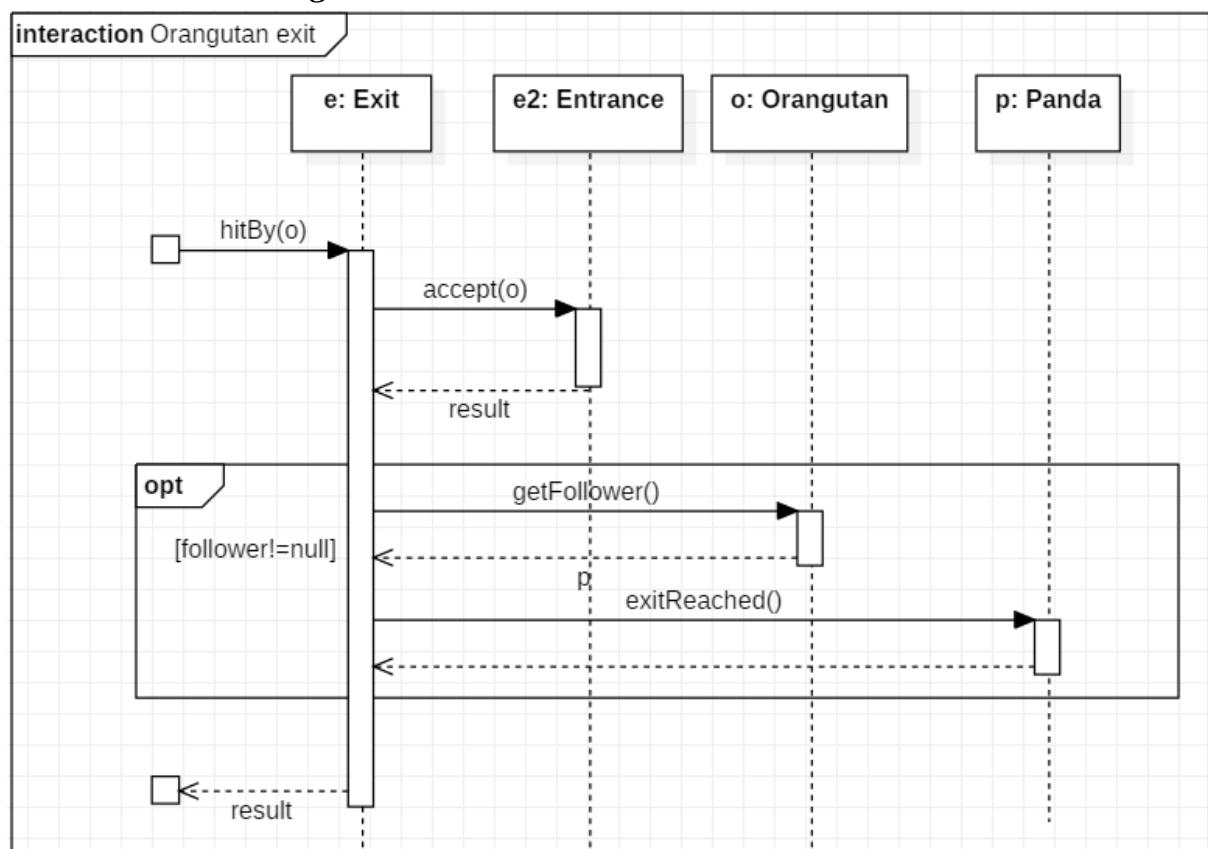
- Attribútumok

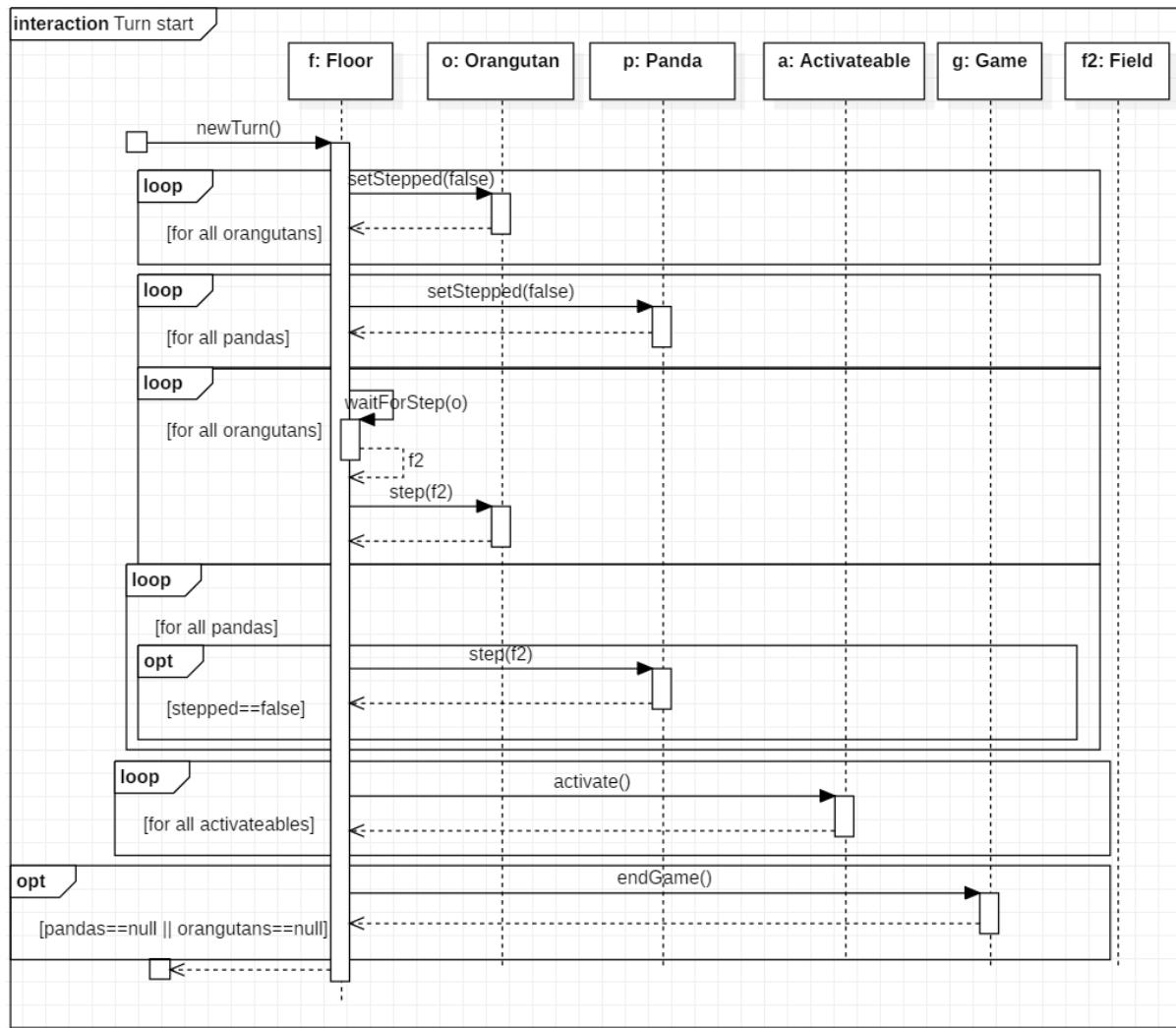
- **Field field:** A mező, ahol a szekrény van.
- **Wardrobe[] wardrobes:** A többi szekrény a pályán, ezek valamelyikéből bukkan elő, ha valami belelép. Nem lehet üres, legalább 2 szekrénynek kell lennie a pályán.

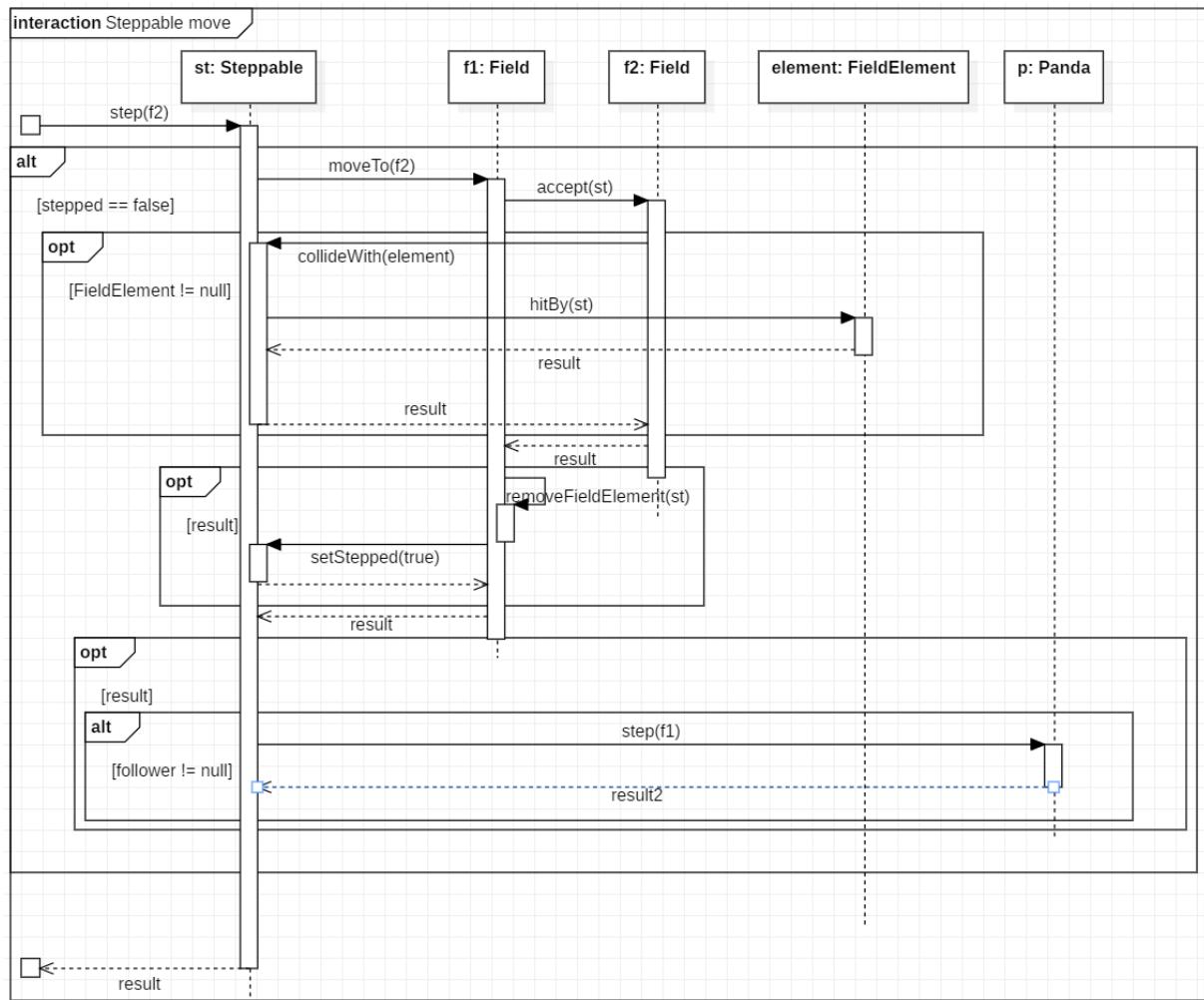
- Metódusok

- **boolean hitBy(Panda p):** A függvény paraméterként kap egy Panda objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **boolean hitBy(Orangutan o):** A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void sleep(Sofa s):** A pihenést szimbolizáló virtuális függvény. Default implementációja üres, hiszen a pihenés egy speciális képesség, amivel csak bizonyos pandák rendelkeznek.
- **void boo():** A megijedést szimbolizáló virtuális függvény. Default implementáció szerint nem történik semmi, ha megijed egy FieldElement. Az megijedés egy képesség, amivel csak bizonyos Pandák rendelkeznek.
- **void jump():** Az ugrást szimbolizáló virtuális függvény. Default implementációja üres, hiszen az ugrás egy speciális képesség, amivel csak bizonyos Pandák rendelkeznek.
- **boolean receive(Steppable s):** Ha s belelőp a szekrénybe, akkor ez meghívódik, és s át fog kerülni egy másik szekrénybe.

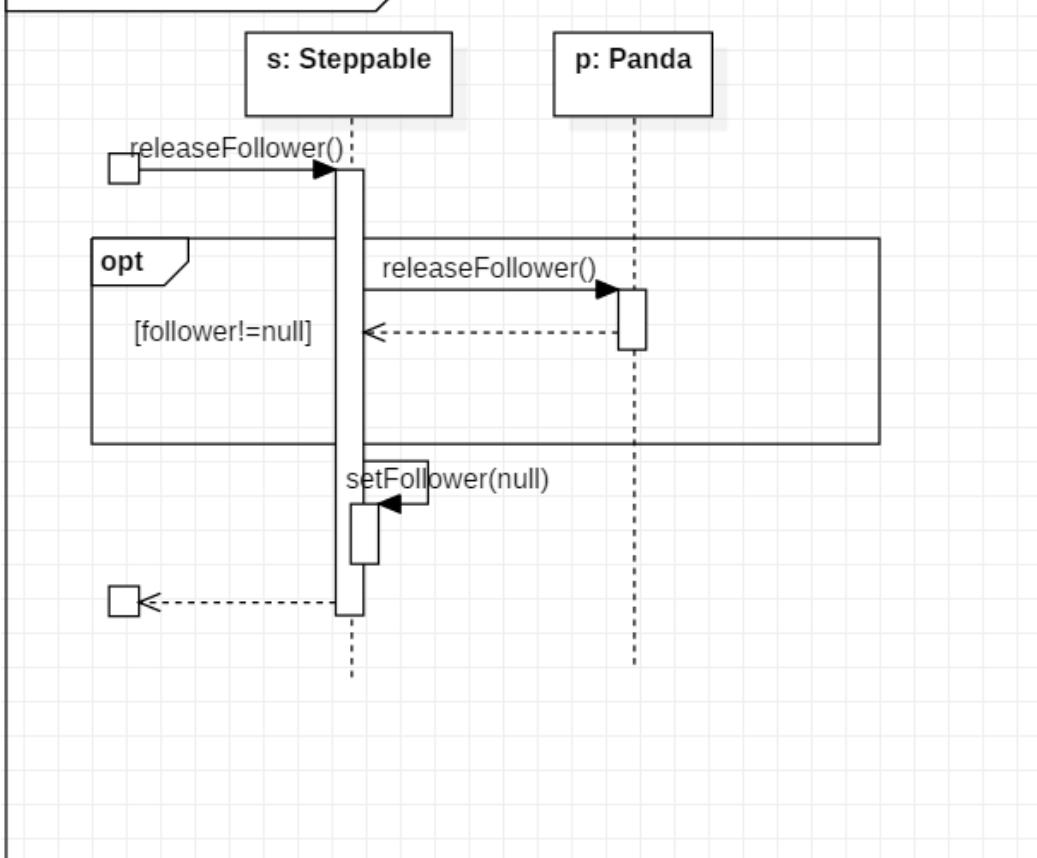
3.4 Szekvencia diagramok



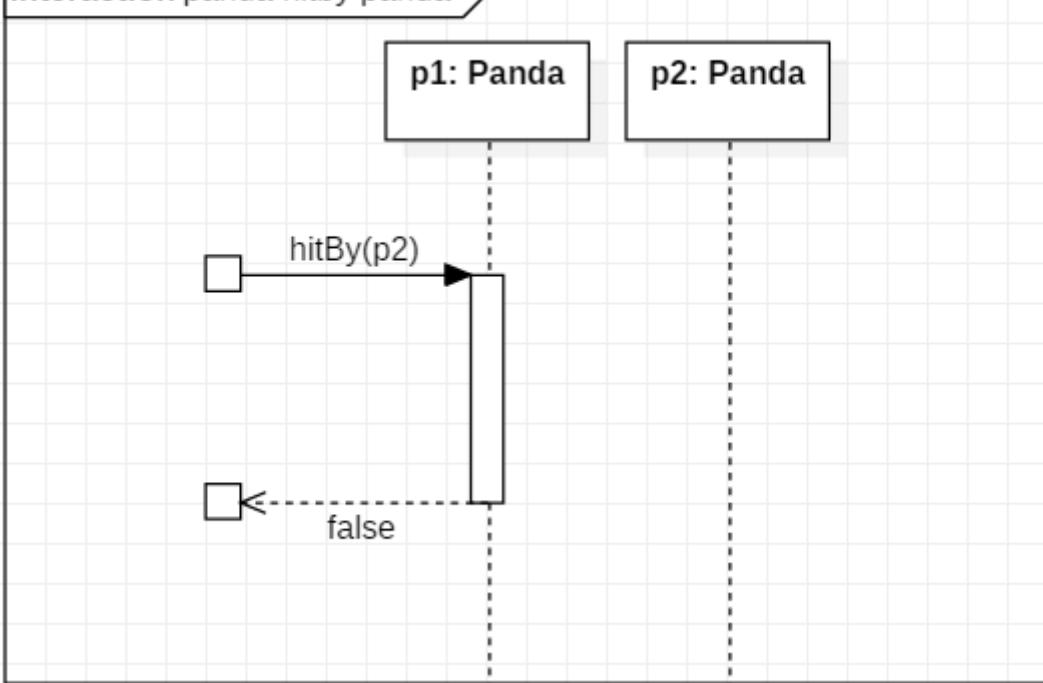


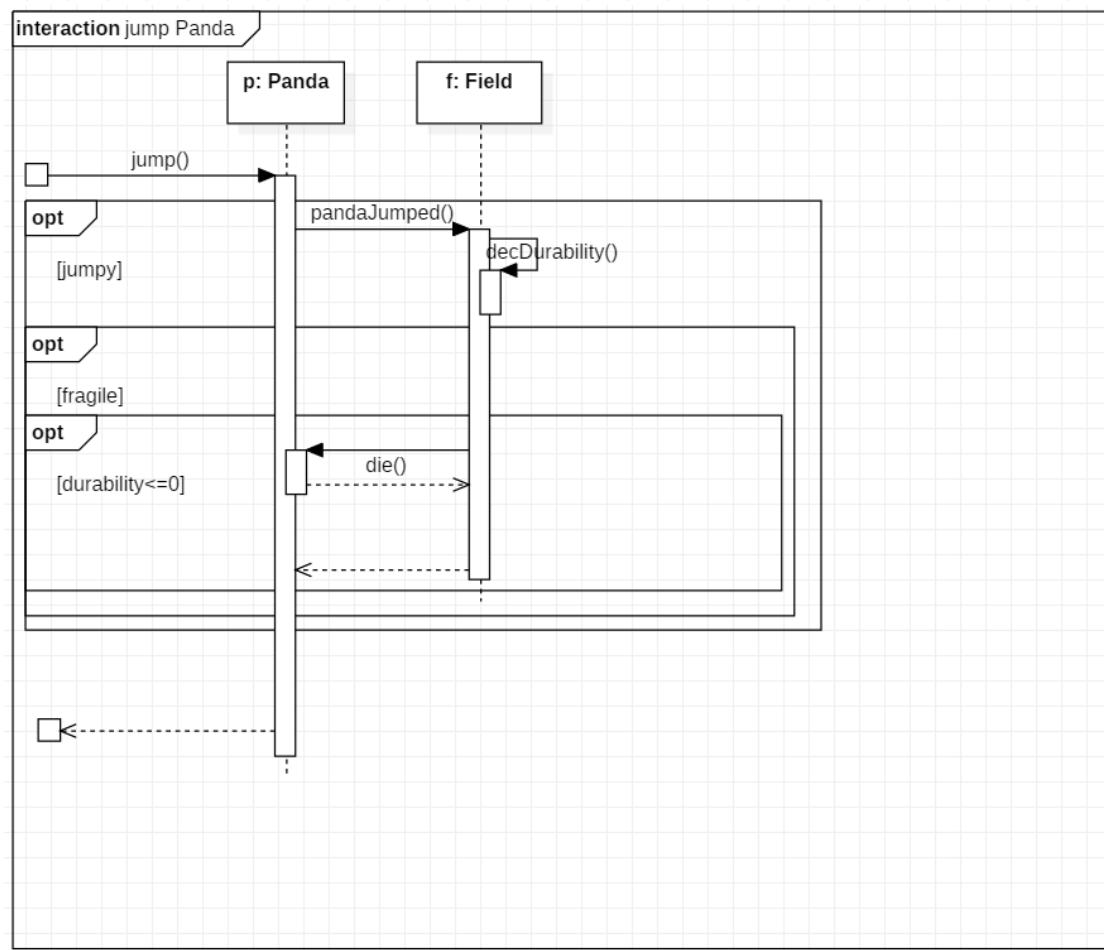
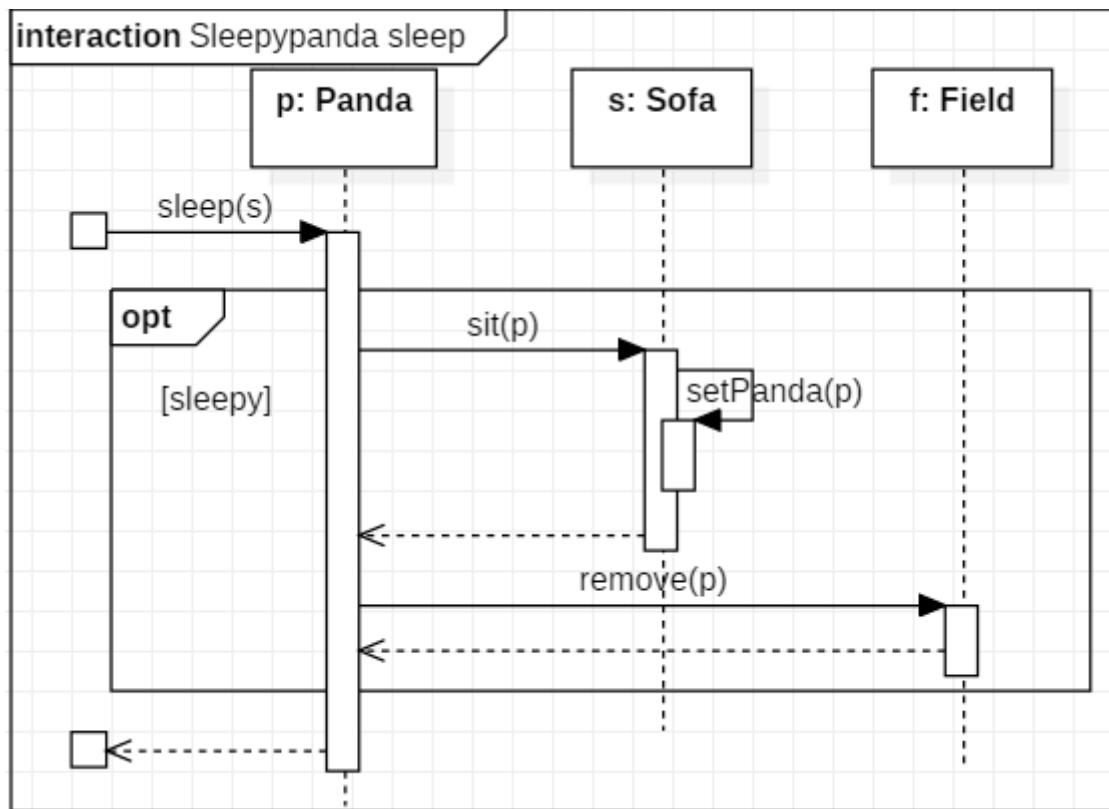


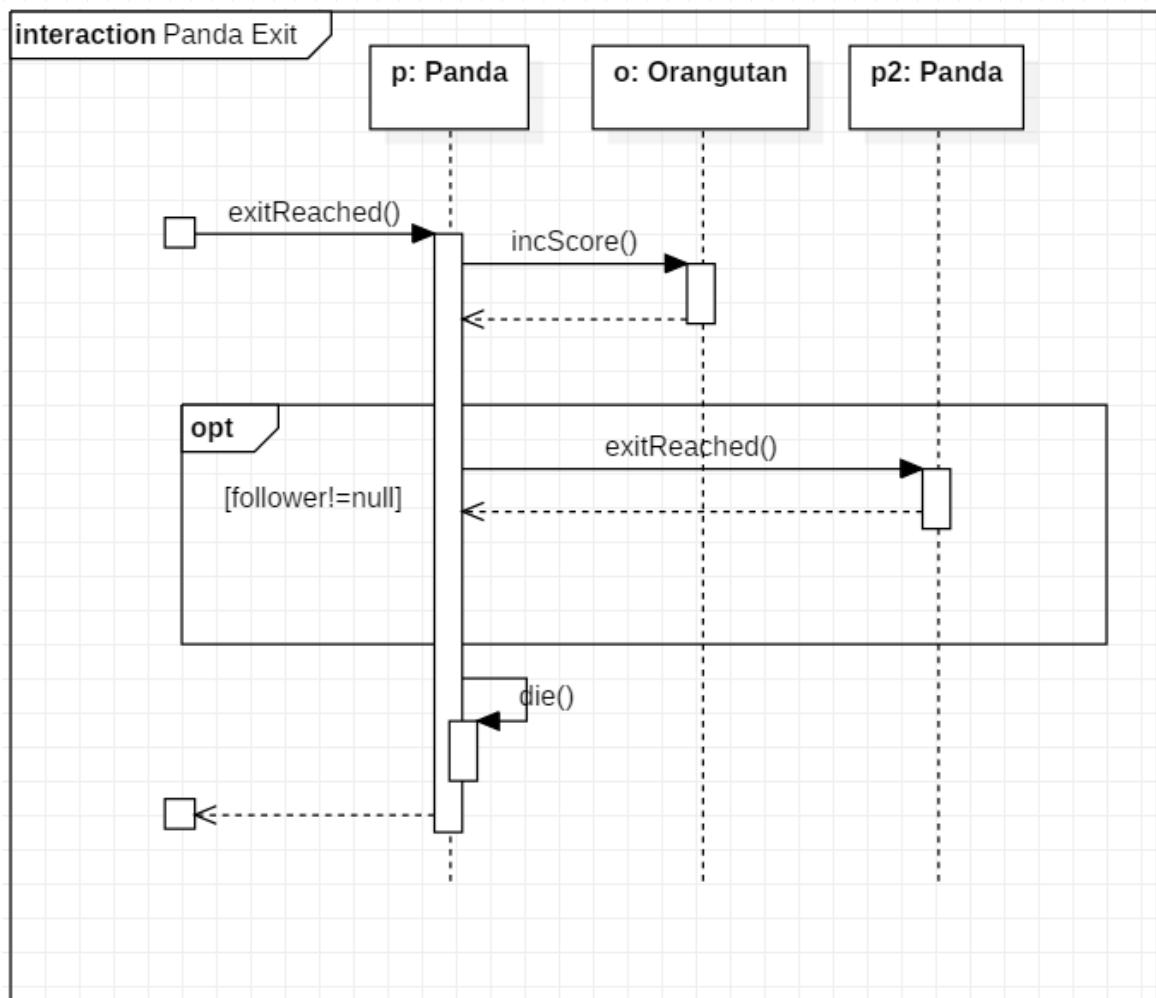
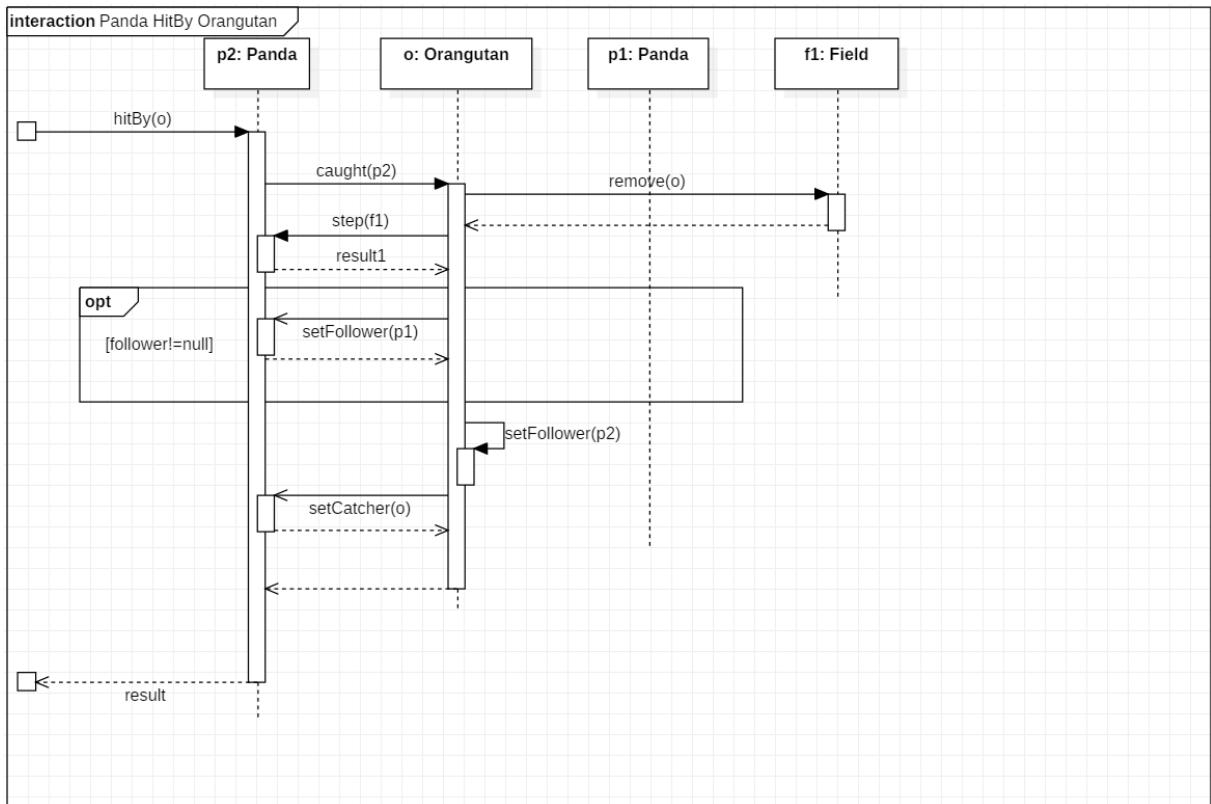
interaction ReleaseFollower

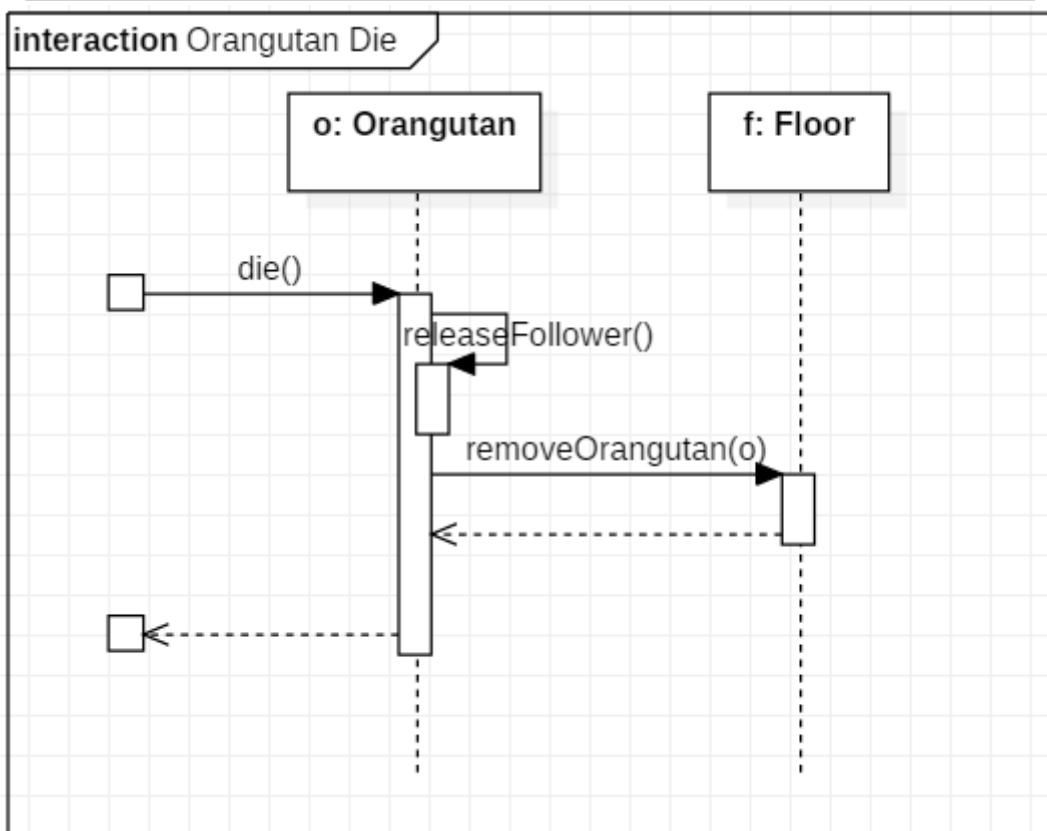
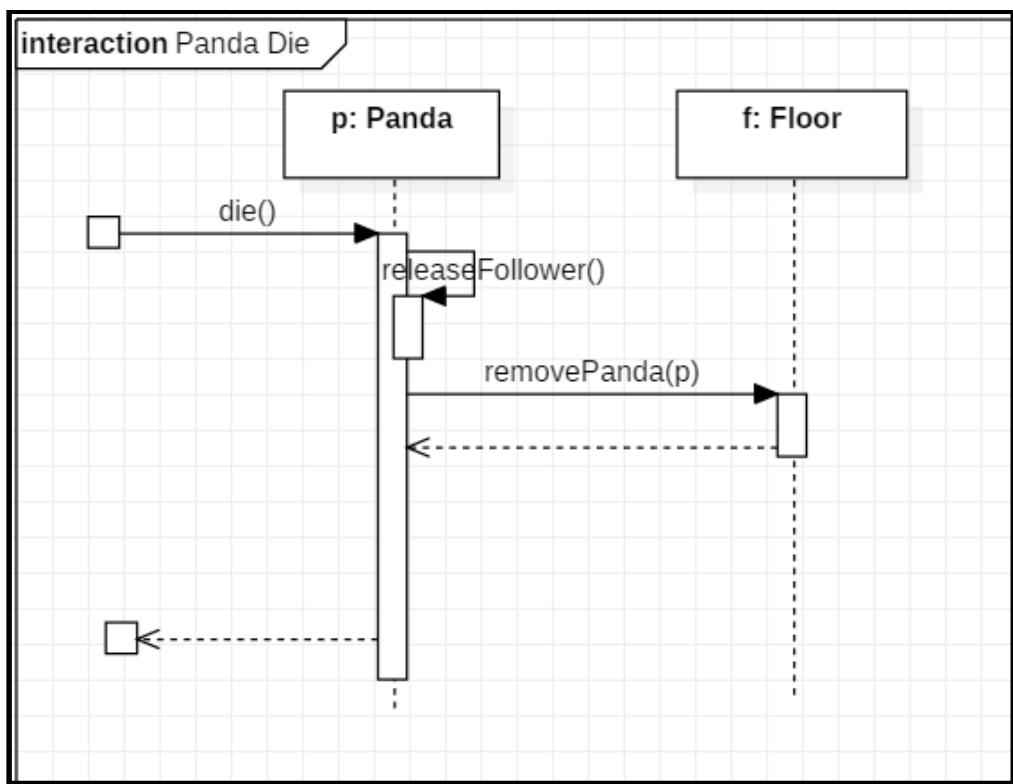


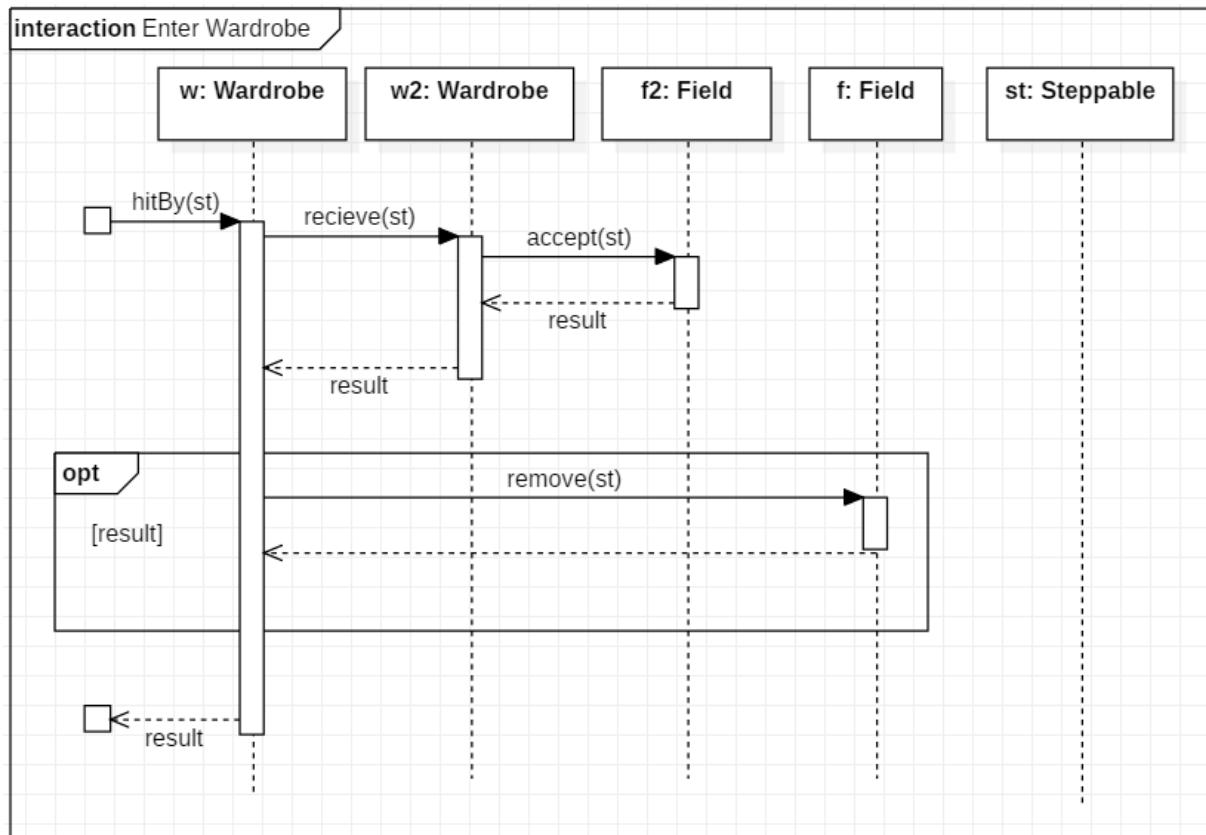
interaction panda hitby panda

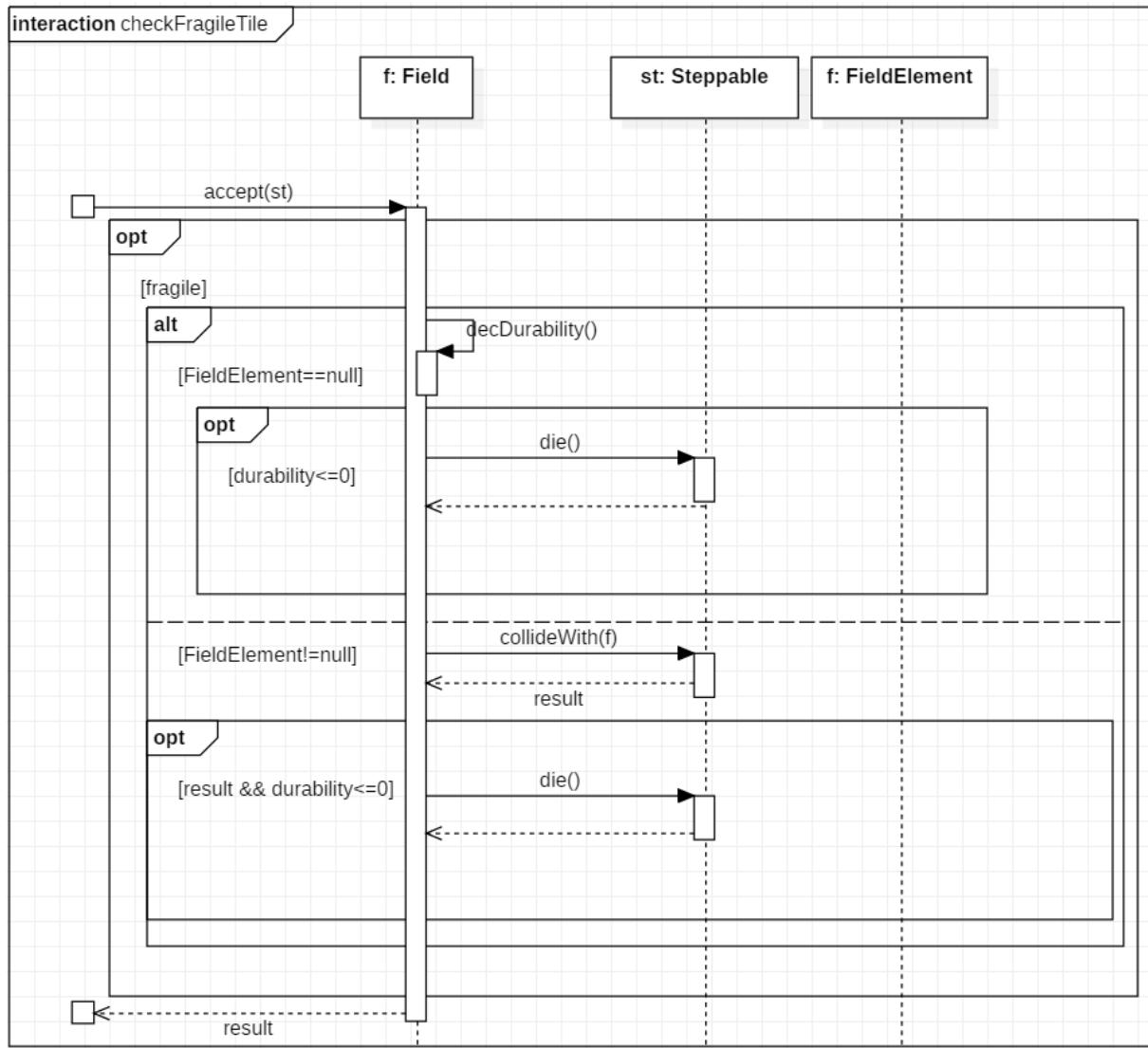


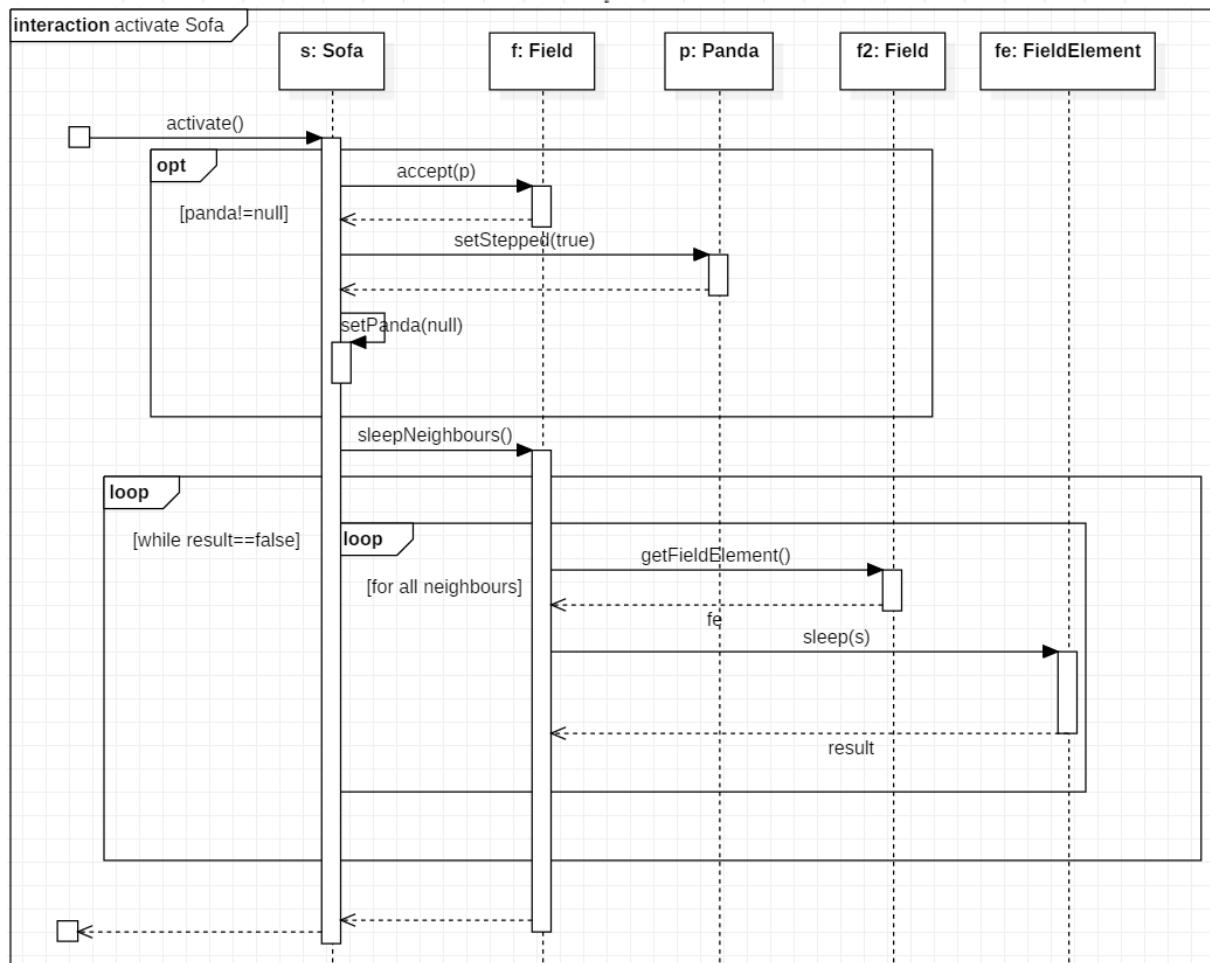
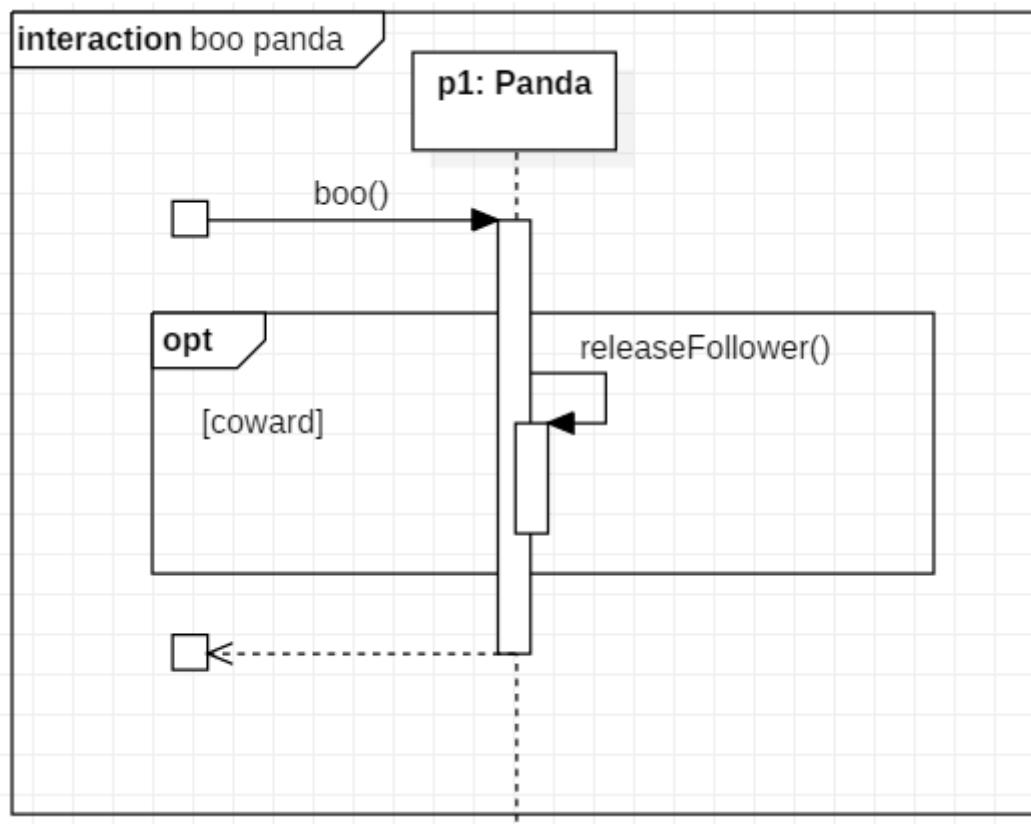


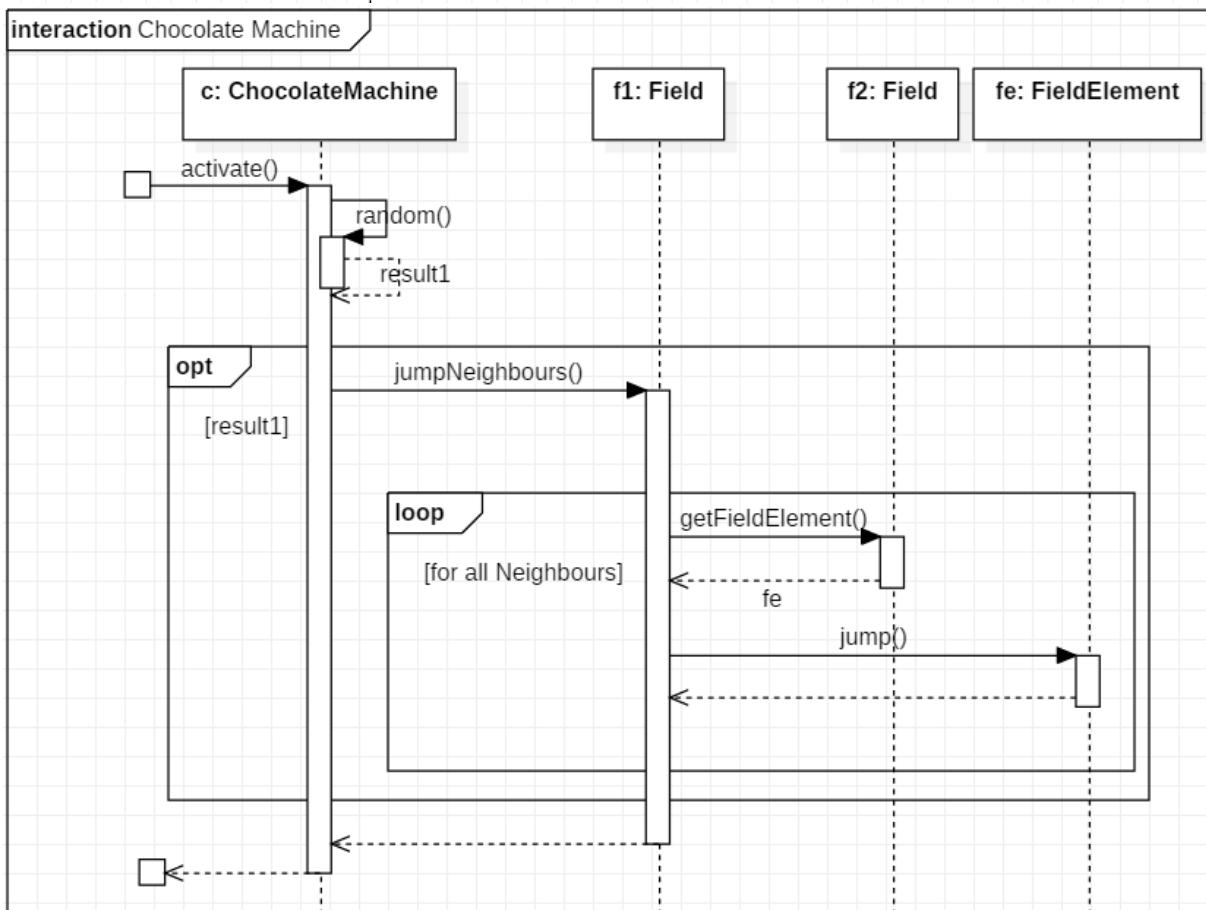
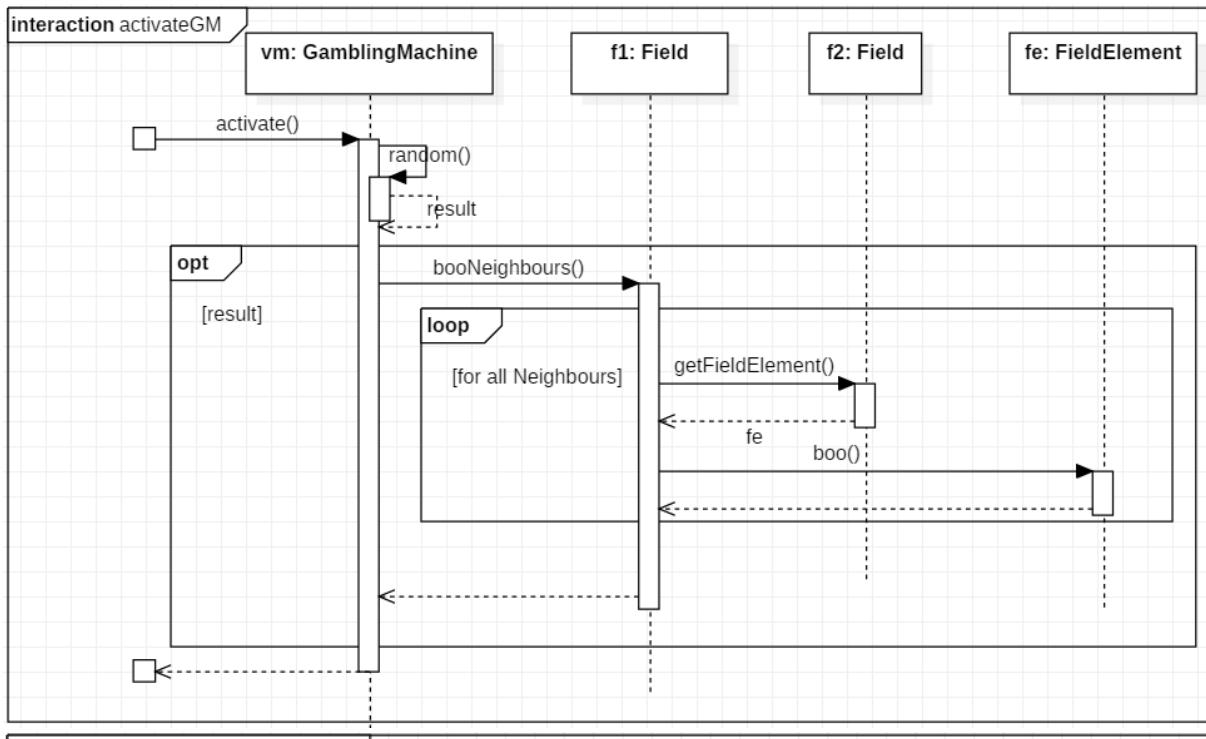






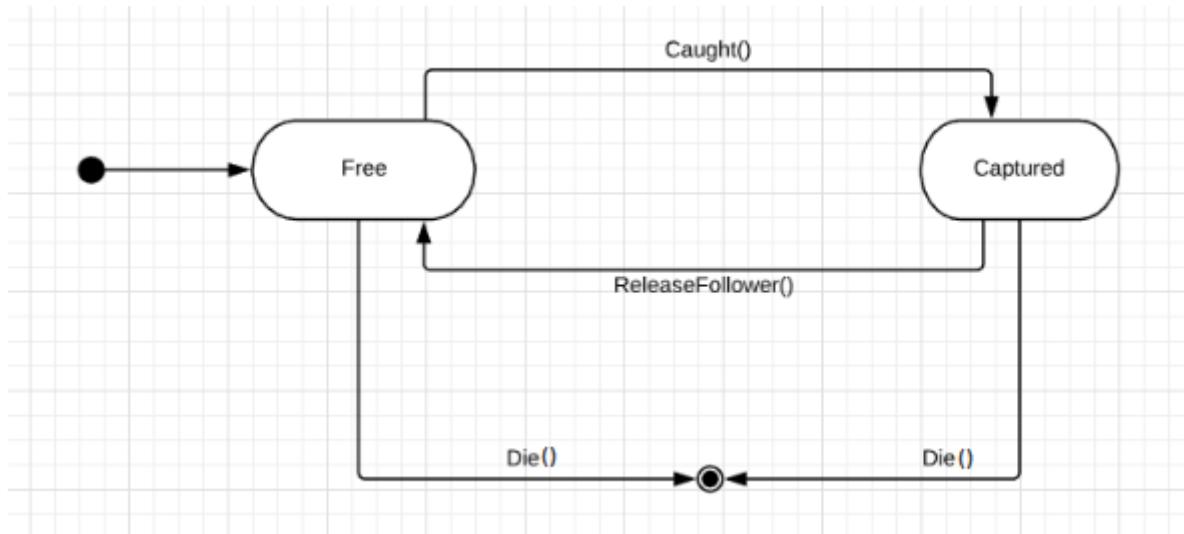






3.5 State-chartok

3.5.1 States of Panda



3.6 Napló

Kezdet	Időtartam	Résznevők	Leírás
2019.02.21. 19:00	1 óra	Nyári	Objektumkatalógus elkészítése
2019.02.22. 16:00	2 óra	Borsodi	Osztálydiagram vázlat elkészítése
2019.02.22. 20:00	1 óra	Bódi	Borsodi által készített osztálydiagram vázlat szerkesztése
2019.02.23. 10:00	2 óra	Bódi	Szekvencia diagramok tervezése
2019.02.23. 17:00	3 óra	Borsodi	Bódi által készített szekvencia diagramok átnézése és a hibák javítása, illetve újak tervezése
2019.02.23. 21:00	1,5 óra	Bódi	Szekvencia diagramok javítása
2019.02.23. 21:00	1 óra	Borsodi	Osztálydiagram javítása
2019.02.23. 20:00	2 óra	Hegedüs	Osztályok leírásának elkészítése
2019.02.24. 22:00	1,5 óra	Nyári	Osztályok leírásának kiegészítése, a state-chart elkészítése, a dokumentum bizonyos pontjainak formázása.
2019.02.24. 00:00	2 óra	Hegedüs	Osztályok leírásának javítása, kiegészítése, továbbá a dokumentum végleges megformázása, előkészítése a nyomtatásra.

4. Analízis modell kidolgozása

4.1 Objektum katalógus

4.1.1 Panda

Egy pandát reprezentál. minden körben tud egyet lépni és a gép irányítja. Mindegyiknek van egy képessége. Háromféle képesség lehetséges. Ha a szomszédban a csokiautomata sípol ugrik egyet, vagy ha játékgép csipog, akkor elengedi a láncban mellette állók mancsát, vagy fotel mellé érve beleülhet abba. minden panda tárolja a mezőt, amin áll, és ha megfogták, az öt elfogó orángutánt és a mögötte lévő pandákat is tárolja.

4.1.2 Orangutan

Egy orángutánt valósít meg a játékban. minden körben egyet tud lépni és egy külső felhasználó irányítja. Az orángután tudja, hogy melyik mezőn áll, illetve az elfogott pandák közül az utolsót, mert az van legelöl a sorban. Ha egy Panda mellé lép, akkor azt elfogja, és a Panda a sor elejére kerül. Nyilvántartja a szerzett pontjainak számát, amit a pandák kivezetéséért kap.

4.1.3 Field

Egy csempét valósít meg. minden más objektum lehet rajta, de csak egy dolog egy időben. Lehet törékeny csempe is, ami tárolja a hátralevő lépések számát is. Ha az elfogyott, akkor betörik, és aki rálép, illetve rajta áll meghal. Feladata a szomszédos csempékkal történő kommunikáció is.

4.1.4 Entrance

A bejáratot valósítja meg, egy darab van belőle, és az orángutánok itt lépnek be a pályára miután a kijáraton kiléptek.

4.1.5 Exit

A kijáratot valósítja meg, az orángutánok ezen keresztül viszik ki a megfogott pandákat, továbbá számíthatnak a bejáratot és továbbítja neki a kilépő orángutánt.

4.1.6 Sofa

Egy fotelt valósít meg. Ha egy pandának ez a képessége, és elhalad egy szomszédos mezőn, akkor beül egy körre a fotelbe, és ha már megvolt fogva, akkor a mögötte lévő kezét elengedi. Tárolja azt a mezőt, amin áll (nem tud mozogni) és ha ült benne panda, akkor azt is.

4.1.7 Chocolate Machine

Csokiautomatát valósít meg, tárolja saját mezőjét. Néha sípol és ez a szomszédos csempéken hallatszik. Ha ott olyan panda áll, amely ijedős, a csokiautomata sípolásától ugrik egyet.

4.1.8 Gambling Machine

Játékgépet valósít meg, tárolja saját mezőjét, néha csilingel, ekkor értesíti a szomszédos mezőket és ha azokon áll ijedős panda, akkor az elengedi a mögötte álló kezét.

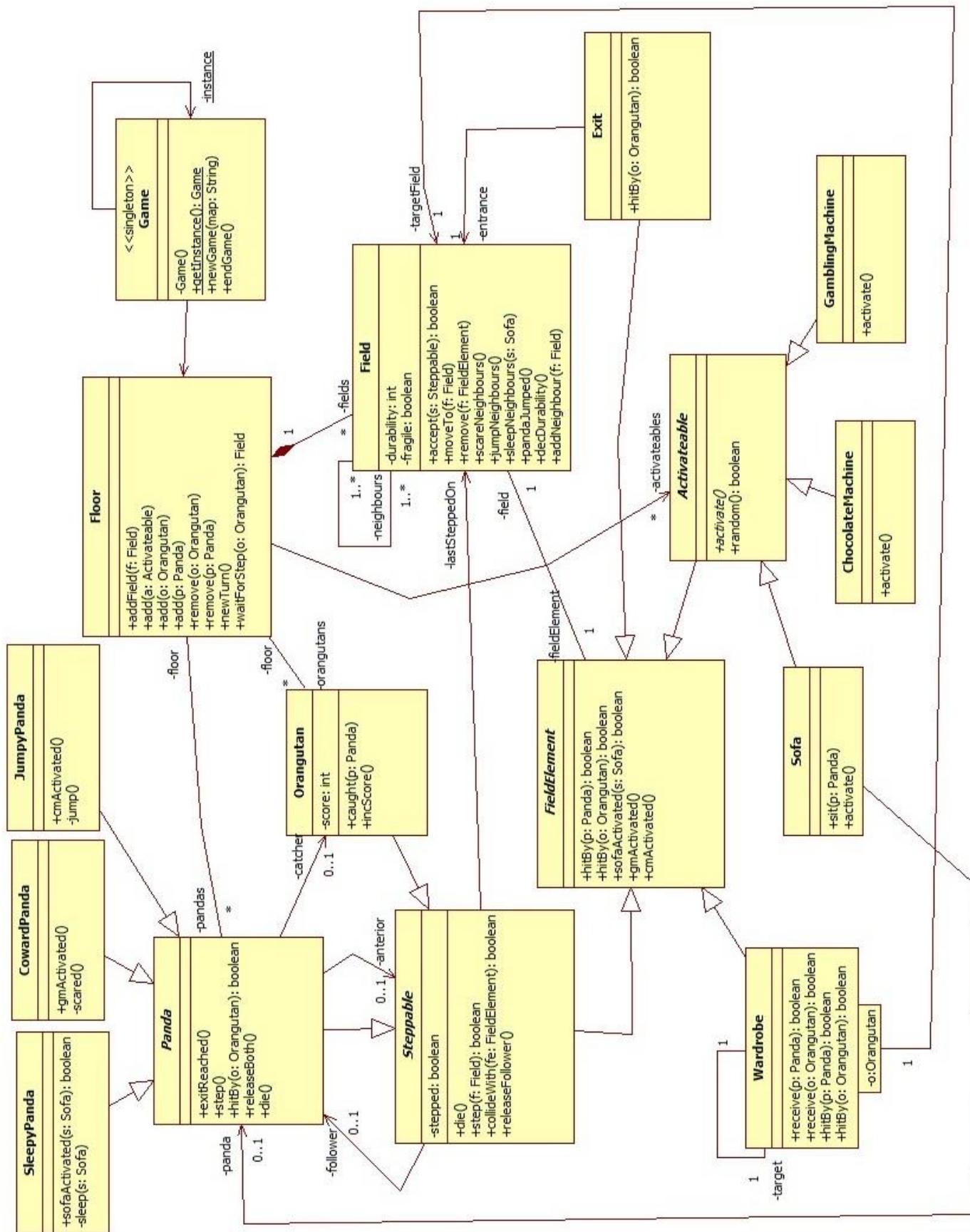
4.1.9 Wardrobe

Egy szekrényt valósít meg, tárolja saját mezőjét. Ha egy orángután belelép egy szekrénybe, akkor véletlenszerűen jön ki egy másik szekrényből. Ezáltal kapcsolatot tart a többi szekrénnel. Ha az orángután a szekrénybe pandákat vezet be, akkor a pandákkal együtt jön ki egy másik szekrényből.

4.1.10 Floor

Az emeletet (padlót) valósítja meg, tárolja a mezőket, az inicializálás során felépíti a pályát, elhelyezi a pandákat és orángutánokat, továbbá számoltartja őket. Új kör esetén értesíti a számoltartott elemeket.

4.2 Statikus struktúra diagramok



4.3 Osztályok leírása

4.3.1 Activateable

- Felelősség

Absztrakt (ős)osztály, mely biztosítja, hogy a belőle származó osztályok aktiválhatók minden körben. Az aktiválás során minden játékban lévő objektum a saját speciális képességének megfelelő műveletet hajtja végre.

- Ősosztályok

FieldElement → Activateable

- Metódusok

- **void activate()**: A speciális képesség aktiválását biztosító absztrakt függvény. minden leszármazott osztály saját maga határozza meg (definiálja), hogy az aktiválás során milyen speciális tevékenységet végez.
- **boolean random()**: Egy virtuális függvény, mely default implementáció szerint pontosan azonos eséllyel tér vissza false vagy true értékkel. minden leszármazott saját maga dönt róla, hogy a függvény default implementációját használja, vagy felüldefiniálja azt.

4.3.2 ChocolateMachine

- Felelősség

Csokiautomát reprezentáló osztály, amely bizonyos speciális képességű Panda objektumokat „ugraszt meg” azzal, hogy időnként aktiválódik és sípol.

- Ősosztályok

FieldElement → Activateable → ChocolateMachine

- Metódusok

- **void activate()**: A speciális képesség (sípolás) aktiválását biztosító függvény. Létezését az Activateable osztály azonos nevű absztrakt függvénye biztosítja.

4.3.3 Exit

- Felelősség

A kijáratot megvalósító osztály. Ha egy orángután rálép, akkor kiviszi a hozzá tartozó sort, és megkapja a pontokat értük.

- Ősosztályok

FieldElement → Exit

- Attribútumok

- **Field entrance**: Tárol egy mezőt, amely egyben a bejárata is a plázának. A kijáraton kilépő pandák itt jönnek vissza.

- Metódusok
 - **boolean hitBy(Orangutan o):** Ha egy orangután rálép, akkor kivezette a mögötte levő sort, és a bejáraton fog bejönni a következő körben.

4.3.4 Field

- Felelősség
Egy mezőt valósít meg. Ezeken állhatnak az állatok, illetve a tárgyak.
- Attribútumok
 - **int durability:** Egyes mezők törékenyek, és van élettartamuk. Ha ennél többször lépnek rájuk, akkor összetörnek. A hátralevő lépések számát tárolja ez az attribútum.
 - **boolean fragile:** Igaz, ha törékeny a csempe, különben hamis.
 - **FieldElement fieldElement:** A csempén álló dolgot tárolja.
 - **Field[] neighbours:** A szomszédos csempéket tárolja.
- Metódusok
 - **boolean accept(Steppable s):** A paraméterként kapott Steppable lépését vezényli. Ha sikeres volt a lépés igaz értéket ad, ha sikertelen, hamis.
 - **void moveTo(Field f):** A mezőn álló elemet mozgatja a paraméterben megadott csempére.
 - **void remove(FieldElement f):** A paraméterben kapott FieldElement-et eltávolítja, ha rajta van az adott mezőn. Ha nem a paraméterben kapott FieldElement van a mezőn, vagy nincs rajta semmi, akkor a függvény nem csinál semmit.
 - **void scareNeighbours():** A játékgép hangját valósítja meg, értesíti a szomszédos mezők FieldElementjeit a játékgép aktiválódásáról.
 - **void jumpNeighbours():** A csokiautomata hangját valósítja meg, értesíti a szomszédos mezők FieldElementjeit a csokiautomata aktiválódásáról.
 - **void sleepNeighbours(Sofa s):** Szól a szomszédos mezőkön álló FieldElementeknek hogy a mezőn a paraméterként átadott Sofa található.
 - **void pandaJumped():** Ha egy panda ugrik egyet a mezőn, akkor meghívja ezt a metódust. Ha törékeny a csempe, akkor csökken az élettartama.
 - **void decDurability():** Csökkenti a csempe élettartamát eggyel.
 - **void addNeighbour(Field f):** A paraméterben adott mezőt hozzáadja a szomszédjaihoz

4.3.5 FieldElement

- Felelősség
A mezőn lévő elemek absztrakt (ős)osztálya. Biztosítja a játékban lévő elemek közös viselkedését.
- Attribútumok
 - **Field field:** Azt a mezőt tárolja, melyen az adott FieldElement elhelyezkedik.

- Metódusok
 - **boolean hitBy(Panda p):** A függvény paraméterként kap egy Panda objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza. Default implementáció szerint nem történt ütközés, így false-al tér vissza.
 - **boolean hitBy(Orangutan o):** A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza. Default implementáció szerint nem történt ütközés, így false-al tér vissza.
 - **boolean sofaActivated(Sofa s):** A függvény a Sofa aktiválódása által kiváltott viselkedést tartalmazza. Default implementáció szerint üres, csak azok a leszármazottak definiálják felül, melyeknek reagálniuk kell a Sofa aktiválódására. Ha egy FieldElement reagál a Sofa aktiválódására (leül a Sofa-ra), akkor a függvény true-val tér vissza, különben pedig false-al.
 - **void gmActivated():** A függvény a GamblingMachine aktiválódása által kiváltott viselkedést tartalmazza. Default implementáció szerint üres, csak azok a leszármazottak definiálják felül, melyeknek reagálniuk kell a GamblingMachine aktiválódására.
 - **void cmActivated():** A függvény a ChocolateMachine aktiválódása által kiváltott viselkedést tartalmazza. Default implementáció szerint üres, csak azok a leszármazottak definiálják felül, melyeknek reagálniuk kell a ChocolateMachine aktiválódására.

4.3.6 Floor

- Felelősség

Egy emeletet valósít meg, tárolja a csempéket, az aktiválható objektumokat, az orángutánokat és a pandákat.
- Attribútumok
 - **Field[] fields:** A csempéket tartalmazó tömb
 - **Orangutan[] orangutans:** Az orángutánokat tartalmazó tömb
 - **Panda[] pandas:** A pandákat tároló tömb
 - **Activateable[] activateables:** Az aktiválható objektumokat tároló tömb
- Metódusok
 - **void addField(Field f):** A paraméterben kapott mezőt hozzáadja az emelethez
 - **void add(Activateable a):** A paraméterben kapott aktiválható objektumot hozzáadja az emelethez
 - **void add(Orangutan o):** Egy orángutánt ad hozzá az emelethez
 - **void add(Panda p):** Egy pandát ad hozzá az emelethez
 - **void remove(Orangutan o):** Egy orángutánt eltávolít az emeletről
 - **void remove(Panda p):** Egy pandát eltávolít az emeletről
 - **void newTurn():** Elindít egy új kört a játékban
 - **Field waitForStep(Orangutan o):** A paraméterben megadott orángután válaszára vár. Az orángután a visszaadott mezőre fog lépni.

4.3.7 GamblingMachine

- Felelősség

Játékgépet reprezentáló osztály, amely bizonyos speciális képességű Panda objektumokat ijeszt meg azzal, hogy néha aktiválódik és csilingel.

- Ősosztályok

FieldElement → Activateable → GamblingMachine

- Metódusok
 - **void activate():** A speciális képesség aktiválását biztosító függvény. A játékgép a függvény meghívásának hatására véletlenszerűen dönt, hogy csilingeljen-e.

4.3.8 Game

- Felelősség

A játékot irányító singleton osztály, felelős a játék elindításáért és befejezéséért.
- Attribútumok
 - **Game instance:** Tárolja a Game osztály egyetlen példányát.

- Metódusok
 - **Game():** Privát konstruktur, létrehozza a Game osztály egyetlen példányát.
 - **Game getInstance():** Visszatér a Game osztály egyetlen létező példányával.
 - **void newGame(String map):** A paraméterben kapott pályán elkezd egy új játékot.
 - **void endGame():** Ezzel a metódussal befejezi a játékot.

4.3.9 JumpyPanda

- Felelősség

A Panda leszármazottja, egy olyan pandát valósít meg, ami megijed a csokiautomata sípolásától.
- Metódusok
 - **void cmActivated():** Ha egy csokiautomata sípol a szomszédos mezőn, akkor ez a metódus meghívódik és reagál a sípolásra(a panda ugrik egyet).
 - **void jump():** A panda ugrik egyet, és ez csökkenti annak a csempének az élettartamát, amin áll a panda.

4.3.10 Orangutan

- Felelősség

Egy orangutánt valósít meg az osztály. A Steppable leszármazottja és feladata, hogy megfogja a pandákat. Egy emberi szereplő irányítja.
- Ősosztályok

FieldElement → Steppable → Orangutan
- Attribútumok
 - **int score:** Az orangután pontszáma.
 - **Floor floor:** Az emelet, ahol az orangután van.
- Metódusok
 - **void caught(Panda p):** Az orangután elkapja a paraméterben átadott pandát. Ekkor a panda bekerül az orangután mögé a sorba.
 - **void incScore():** Növeli az orangután pontszámát eggyel.

4.3.11 Panda

- Felelősség

Egy pandát valósít meg a játékban. A gép irányítja, célja, hogy az orangutánok ne kapják el.

- Ősosztályok

FieldElement → Steppable → Panda

- Attribútumok

- **Orangutan catcher:** Ha a panda el van kapva, akkor ez az attribútum tárolja az elkapó orangutánt.
- **Steppable anterior:** A panda előtt levő Steppable a sorban, ha a panda el van kapva.
- **Floor floor:** Az emelet, ahol a panda van.

- Metódusok

- **void exitReached():** Ha a pandát vezető orangután elérte a kijáratot, akkor az meghívja a következő panda exitReached függvényét, ami meghívja a lánc következő tagját, így végigmegy az egész láncon. Kitörli magát a panda, és pontot ad az orangutánnak.
- **void step():** A panda ennek hatására lépni fog véletlenszerű szomszédos mezőre.
- **boolean hitBy(Orangutan o):** A függvény paraméterként kap egy Orangutan objektumot, majd eldönti, hogy ezzel az objektummal történt-e ütközés. Ha igen, akkor true-val, különben false-al tér vissza.
- **void releaseBoth():** A sor felbomlik, ezért a panda elengedi mindenkinél a kezét, akinek fogta.
- **void die():** A panda meghal, szól az emeletnek hogy távolítsa el.

4.3.12 SleepyPanda

- Felelősség

Az aluszékony pandát valósítja meg. Ha elmegy egy fotel mellett, akkor abba beleülhet egy körre.

- Ősosztályok

FieldElement → Steppable → Panda → SleepyPanda

- Metódusok

- **boolean sofaActivated(Sofa s):** Értesíti a pandát, hogy a paraméterként átadott fotel a szomszédos mezőn van. Ha úgy dönt, beleülhet egy körre, ekkor a függvény true-val tér vissza, különben pedig false-al.
- **void sleep(Sofa s):** A panda elalszik a paraméterben megadott fotelben. Ha el volt fogva, akkor a sor felbomlik mögötte.

4.3.13 Sofa

- Felelősség

Egy fotel működését valósítja meg. Ha egy fáradékony panda elhalad a szomszédos mezőn, akkor beleülhet egy körre.

- Ősosztályok

FieldElement → Activateable → Sofa

- Attribútumok
 - **Panda panda:** Ha ül panda a fotelben, akkor azt itt tárolja, mert a panda nem állhat a mezőn, hisz ott van a fotel.
- Metódusok
 - **void activate():** Aktiválja a fotelt minden körben, ha ül benne panda, akkor az feláll és elmegy, feltéve, hogy van szabad szomszédos mező. Ha nem ül benne panda, akkor a szomszédos csempéken állóknak meghívja a sofaActivated() függvényét.
 - **void sit(Panda p):** A paraméterben megadott panda elalszik egy körre a fotelben

4.3.14 Steppable

- Felelősség
Egy absztrakt osztály, ami minden olyan tárgynak az ōsosztálya, ami mozgatható a pályán.
- Ősosztályok
FieldElement → Steppable
- Attribútumok
 - **boolean stepped:** Tárolja, hogy lépett e már a körben az objektum.
 - **Panda follower:** A sorban mögötte levő panda, aki fogja a kezét. Ha nincs mögötte panda, akkor üres.
 - **Field lastSteppedOn:** Az utolsó mező, ahova a Steppable lépett.
- Metódusok
 - **void die():** Meghalást reprezentáló függvény, ha meghívódik, az objektum kiesik a játékból.
 - **boolean step(Field f):** Lépést kezdeményez a paraméterben kapott mező felé. Ha sikeres akkor true, ha sikertelen, akkor false a visszatérési érték.
 - **boolean collideWith(FieldElement fe):** Ez a metódus hívódik meg, ha ütközik az fe FieldElementtel. Visszatér az ütközés kimenetével.
 - **void releaseFollower():** Valamiért megbomlik a lánc, és ezért elengedi a mögötte álló kezét, a lánc hátralevő része felbomlik.

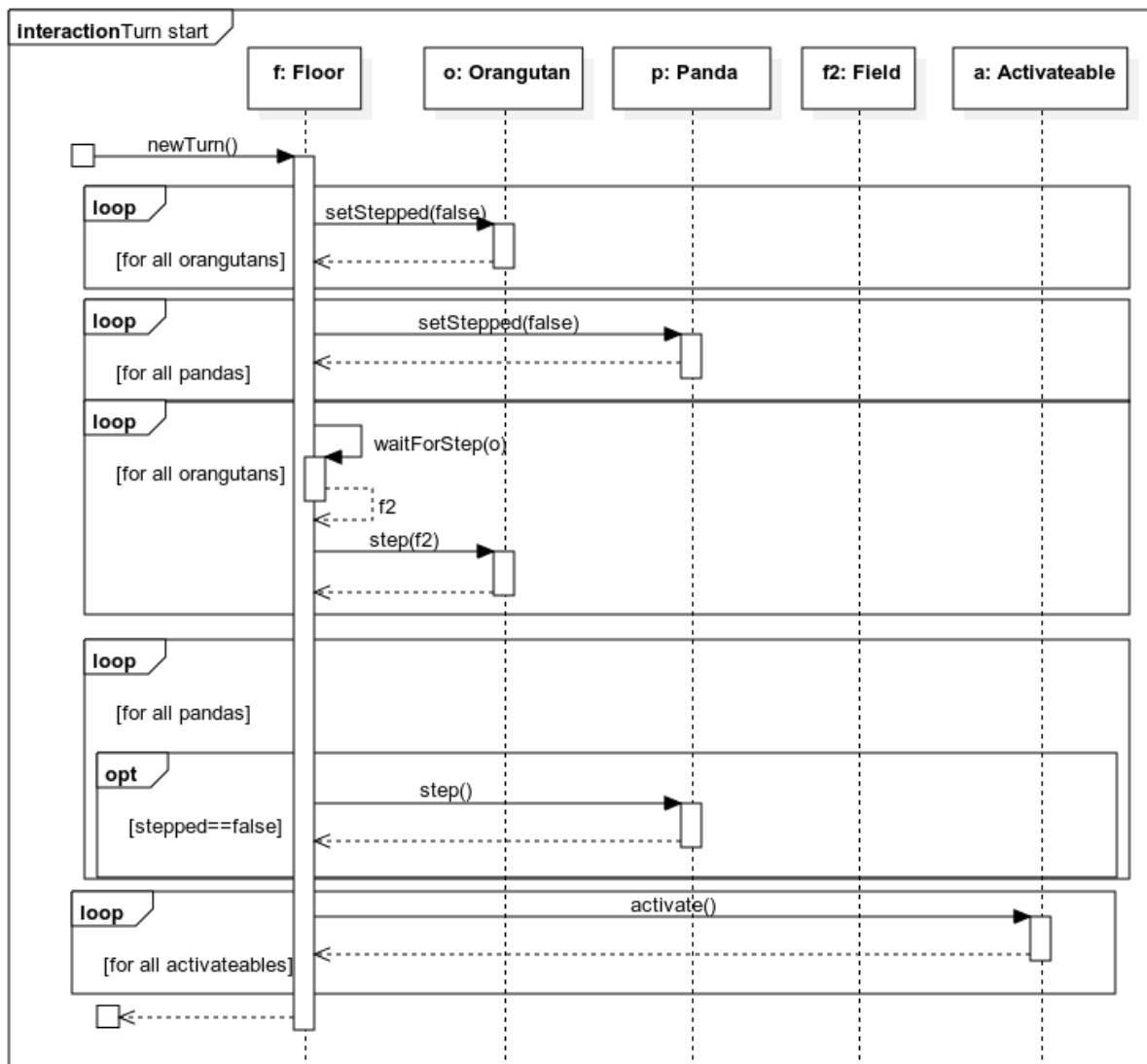
4.3.15 Wardrobe

- Felelősség
Egy szekrényt valósít meg. Ha egy Steppable belelép egybe, akkor egy másik szekrényből bukkan elő.
- Ősosztályok
FieldElement → Wardrobe
- Attribútumok
 - **Wardrobe target:** Az a szekrény, ahova küldi majd a következő belépőt.
 - **Field[Orangutan] targetField:** Az orangutánoknál tárolt mező, ahová az őt követő pandákat továbbítani kell.
- Metódusok
 - **boolean receive(Panda p):** A paraméterként kapott pandát továbbítja egy szomszédos mezőre, ha elfogott pandáról van szó, akkor egy konkrét mezőre, ha szabad panda, akkor véletlenszerűen választott szomszédos mezőre. Visszatér a továbbítás sikerességével.

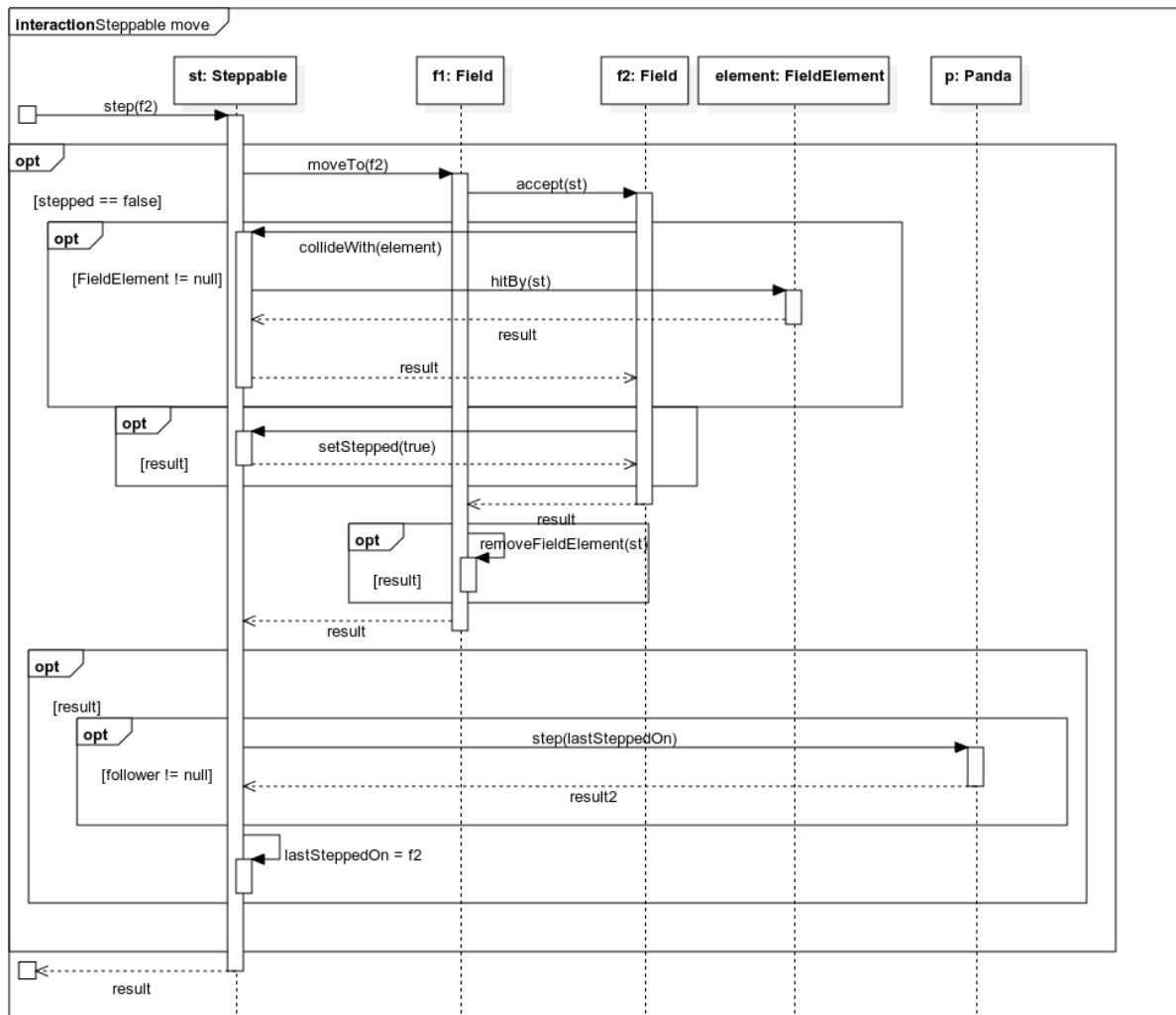
- **boolean receive(Orangutan o):** A paraméterként kapott orangutánt továbbítja egy szomszédos mezőre. Visszatér a továbbítás sikerességével.
- **boolean hitBy(Panda p):** A függvény paraméterként kap egy Panda objektumot, majd megpróbálja továbbítani a célszekrénynek, ha sikerül true-t ad, különben false-al tér vissza.
- **boolean hitBy(Orangutan o):** A függvény paraméterként kap egy Orangutan objektumot, majd megpróbálja továbbítani a célszekrénynek, ha sikerül true-t ad, különben false-al tér vissza.

4.4 Szekvencia diagramok

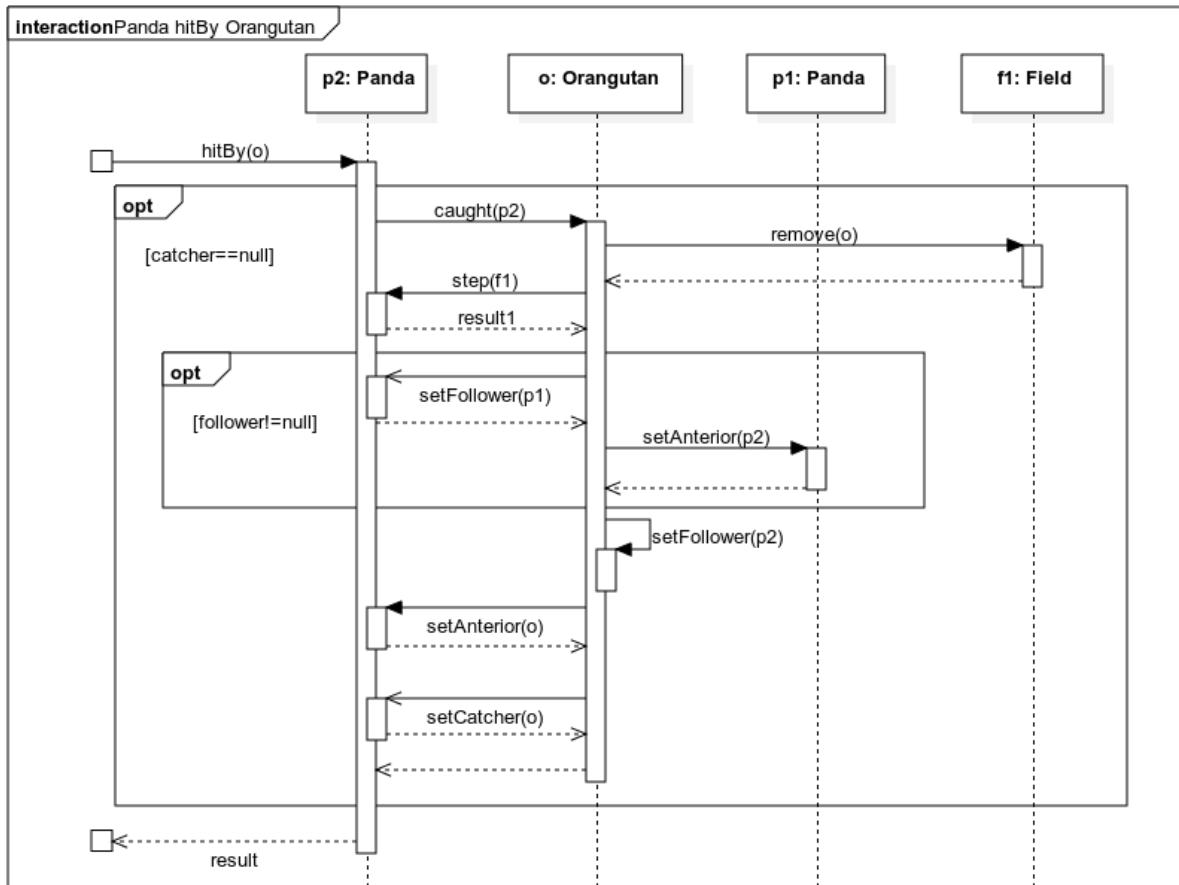
4.4.1 Turn start



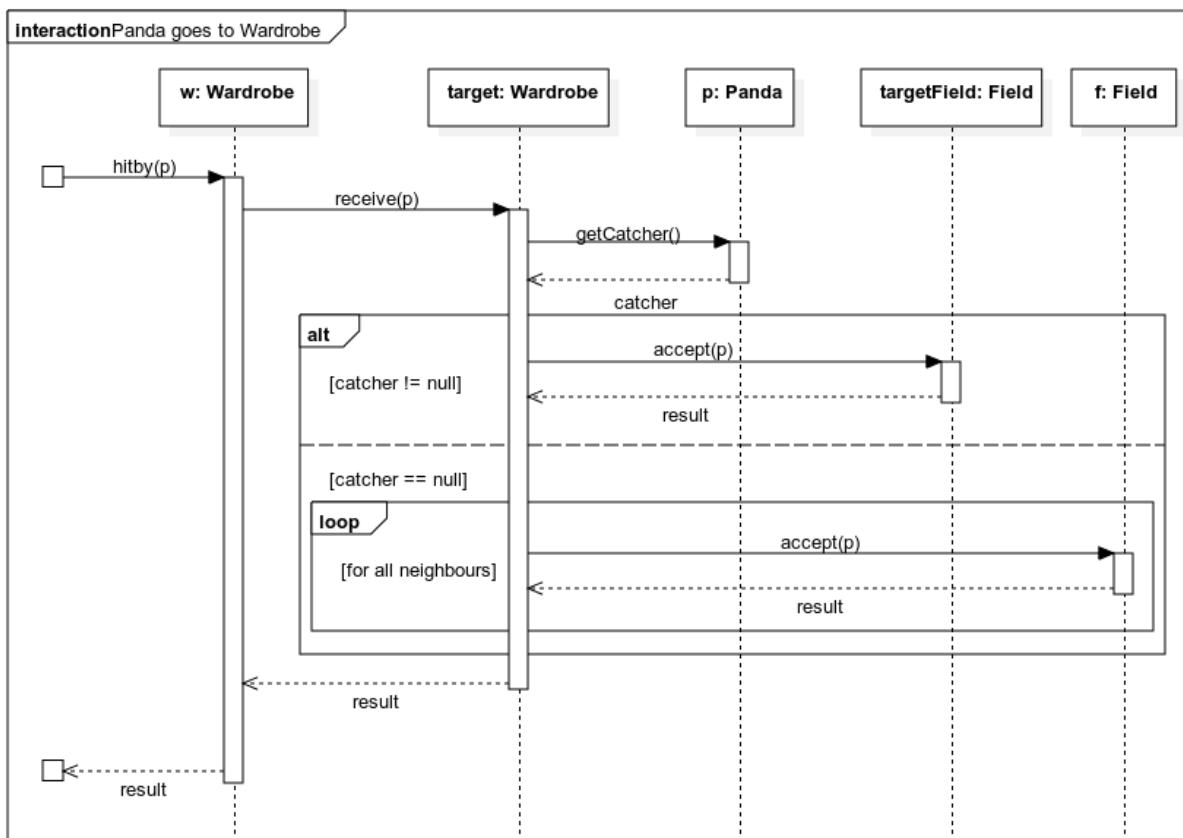
4.4.2 Steppable move



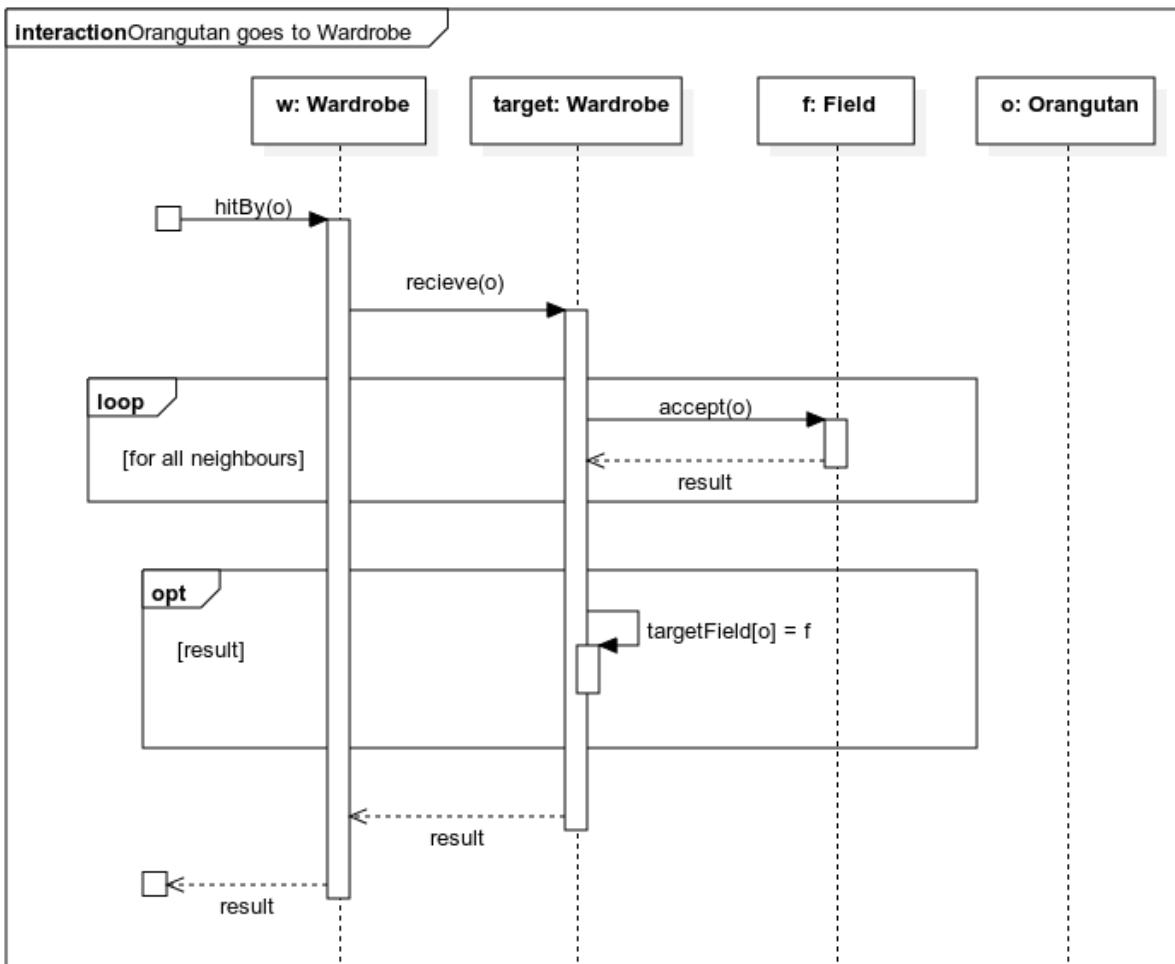
4.4.3 Panda hitBy Orangutan



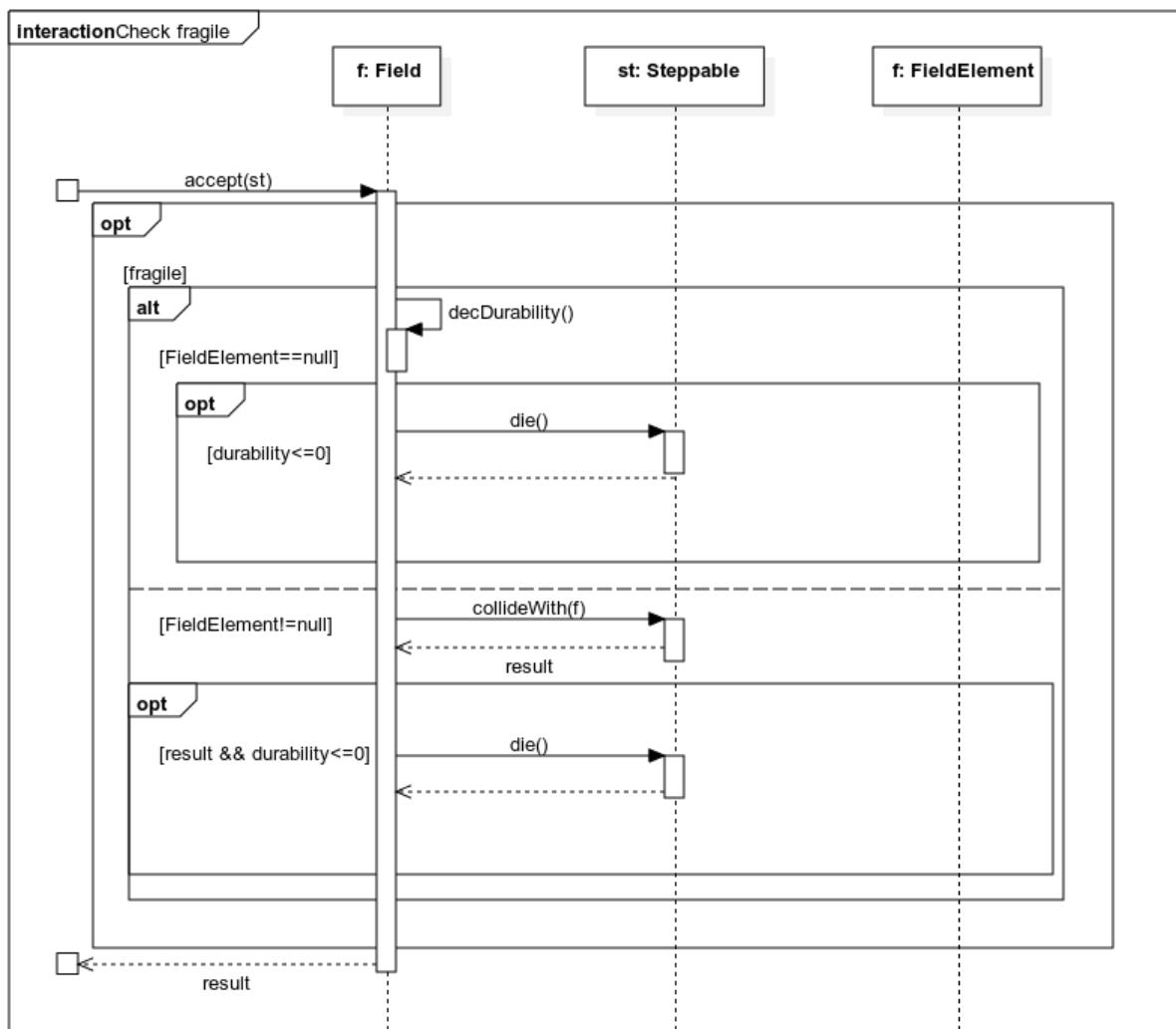
4.4.4 Panda goes to Wardrobe



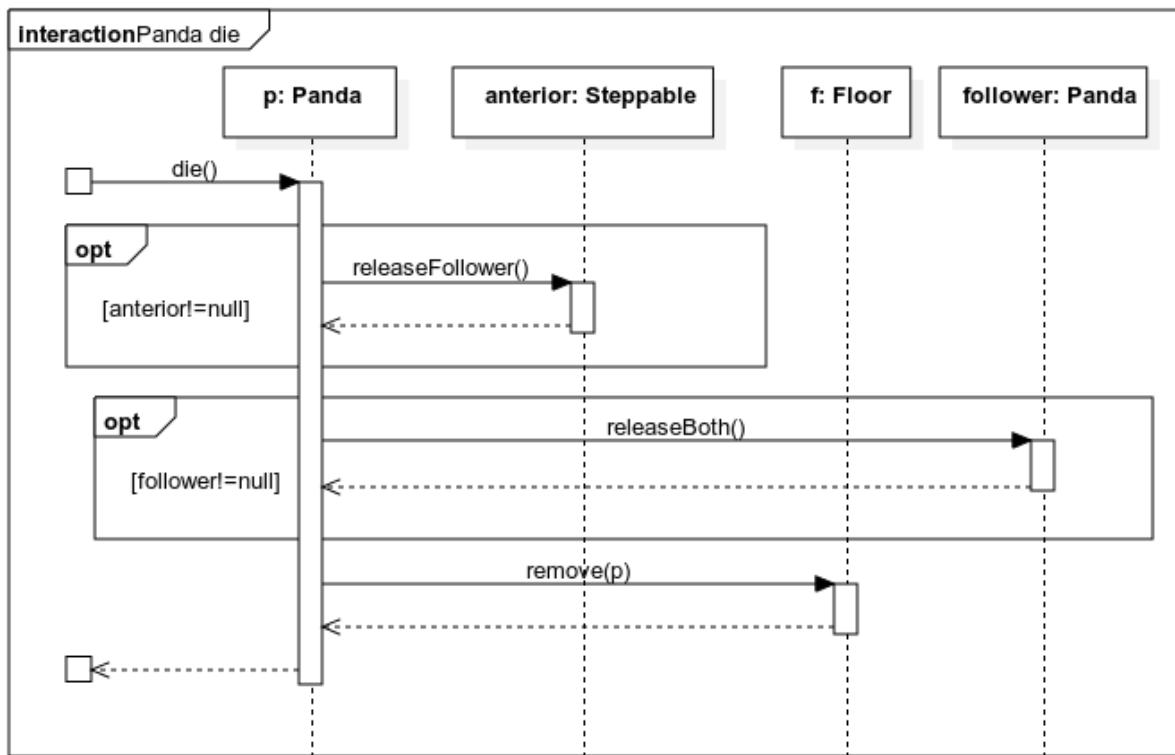
4.4.5 Orangutan goes to Wardrobe



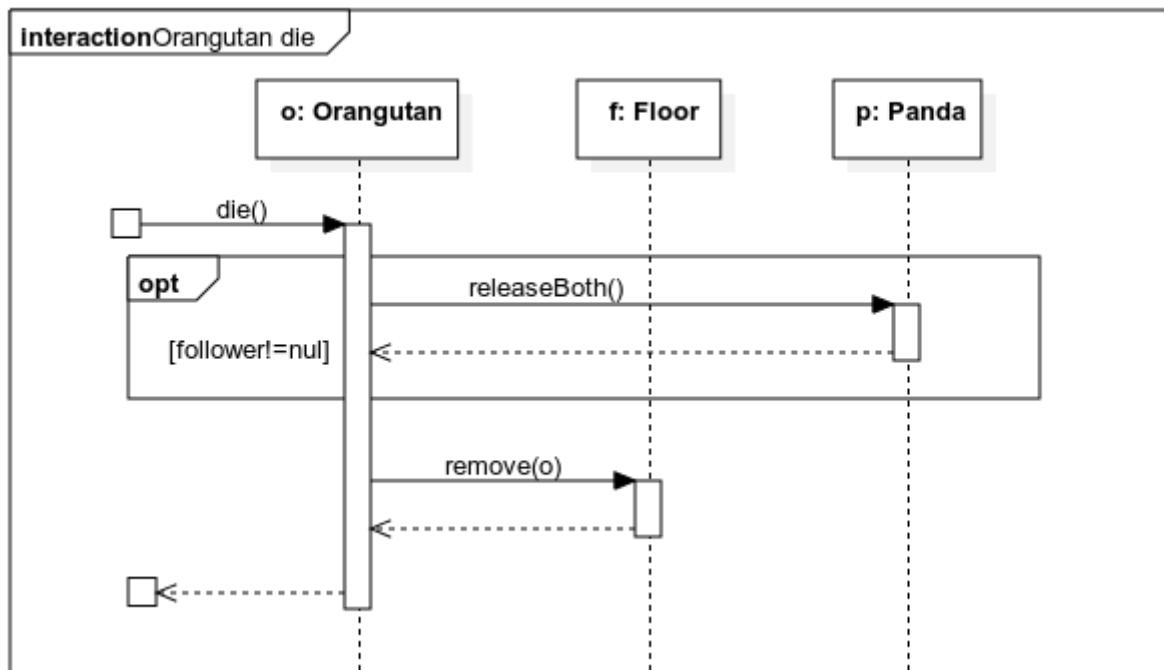
4.4.6 Check fragile



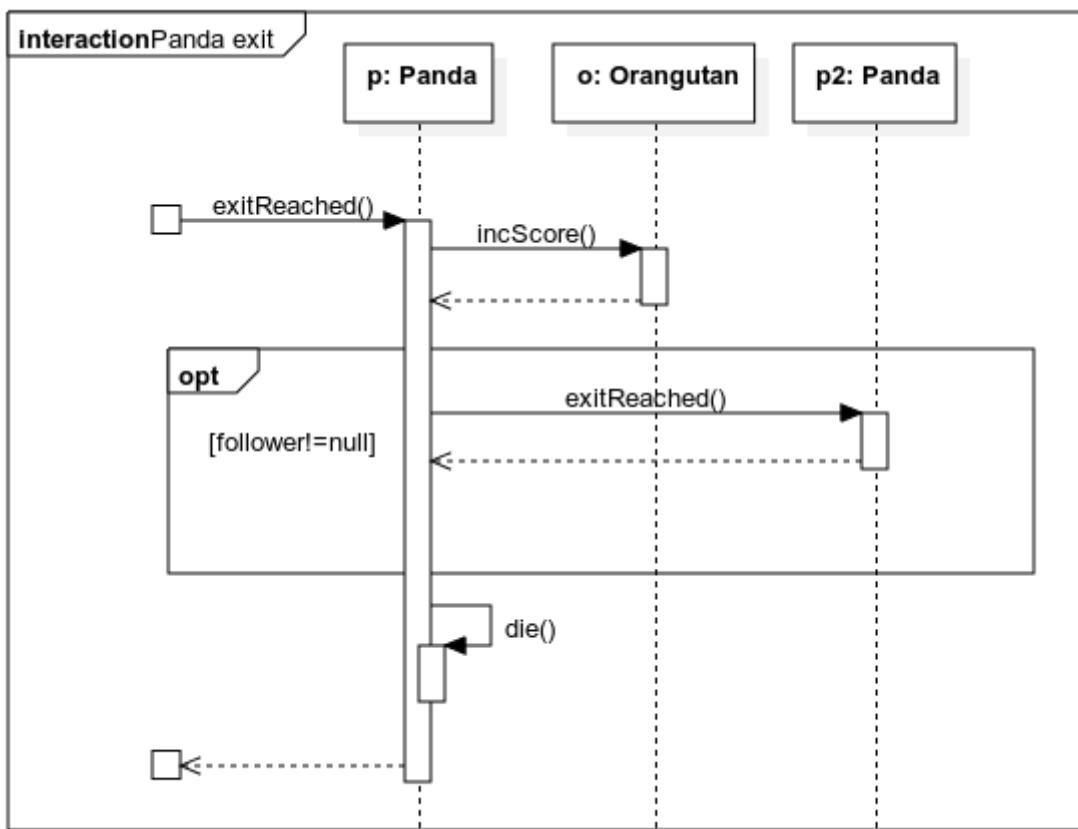
4.4.7 Panda die



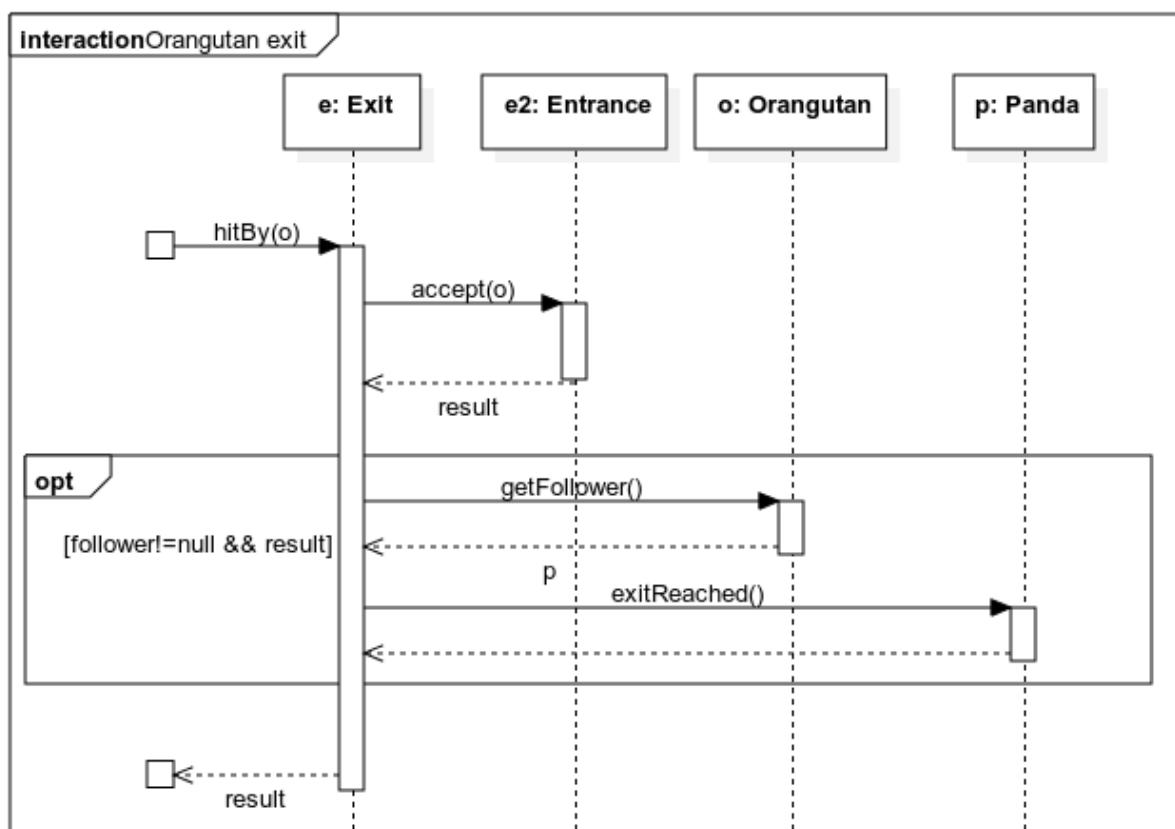
4.4.8 Orangutan die



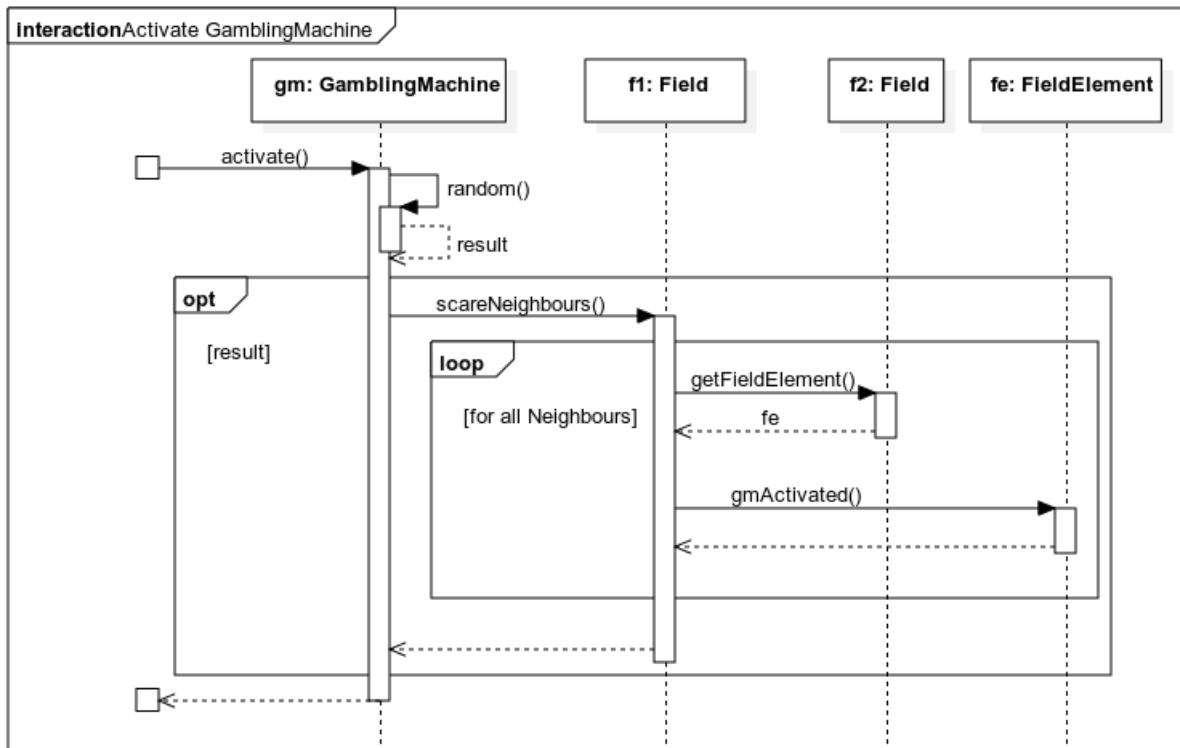
4.4.9 Panda exit



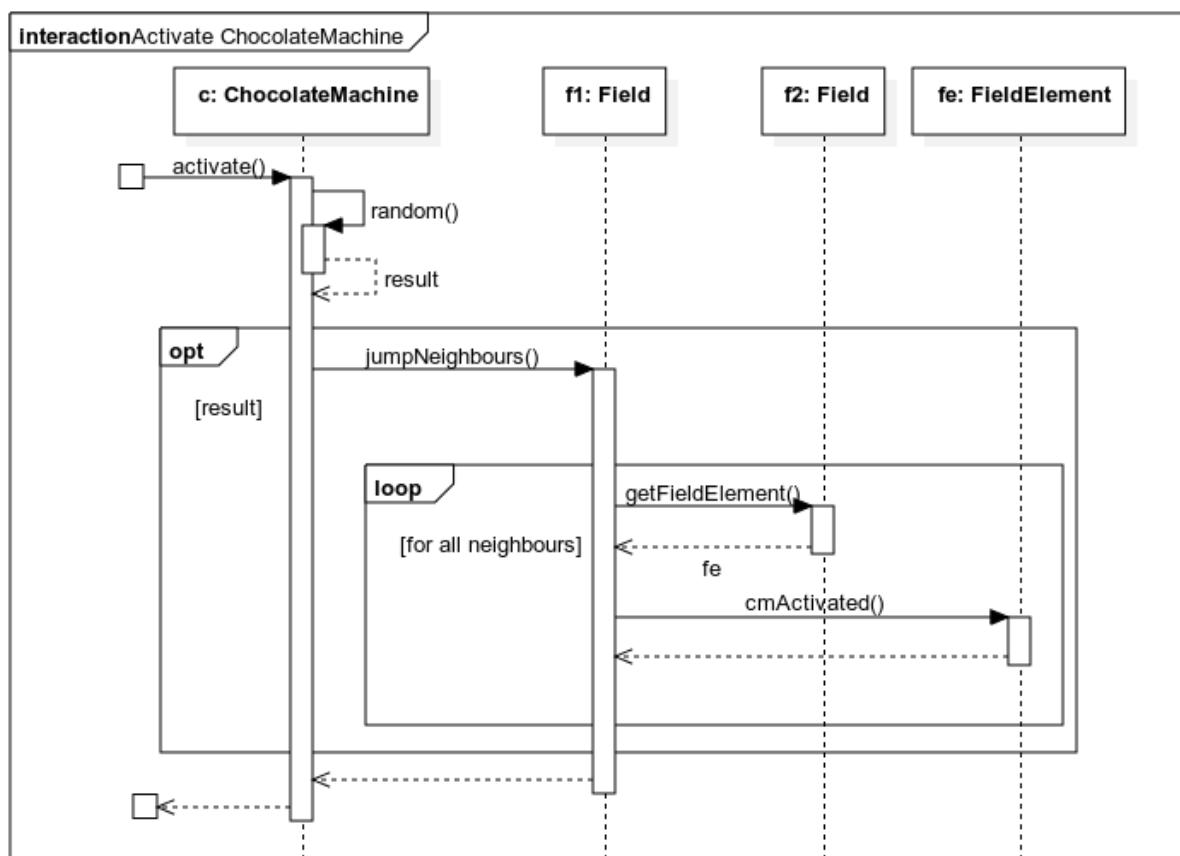
4.4.10 Orangutan exit



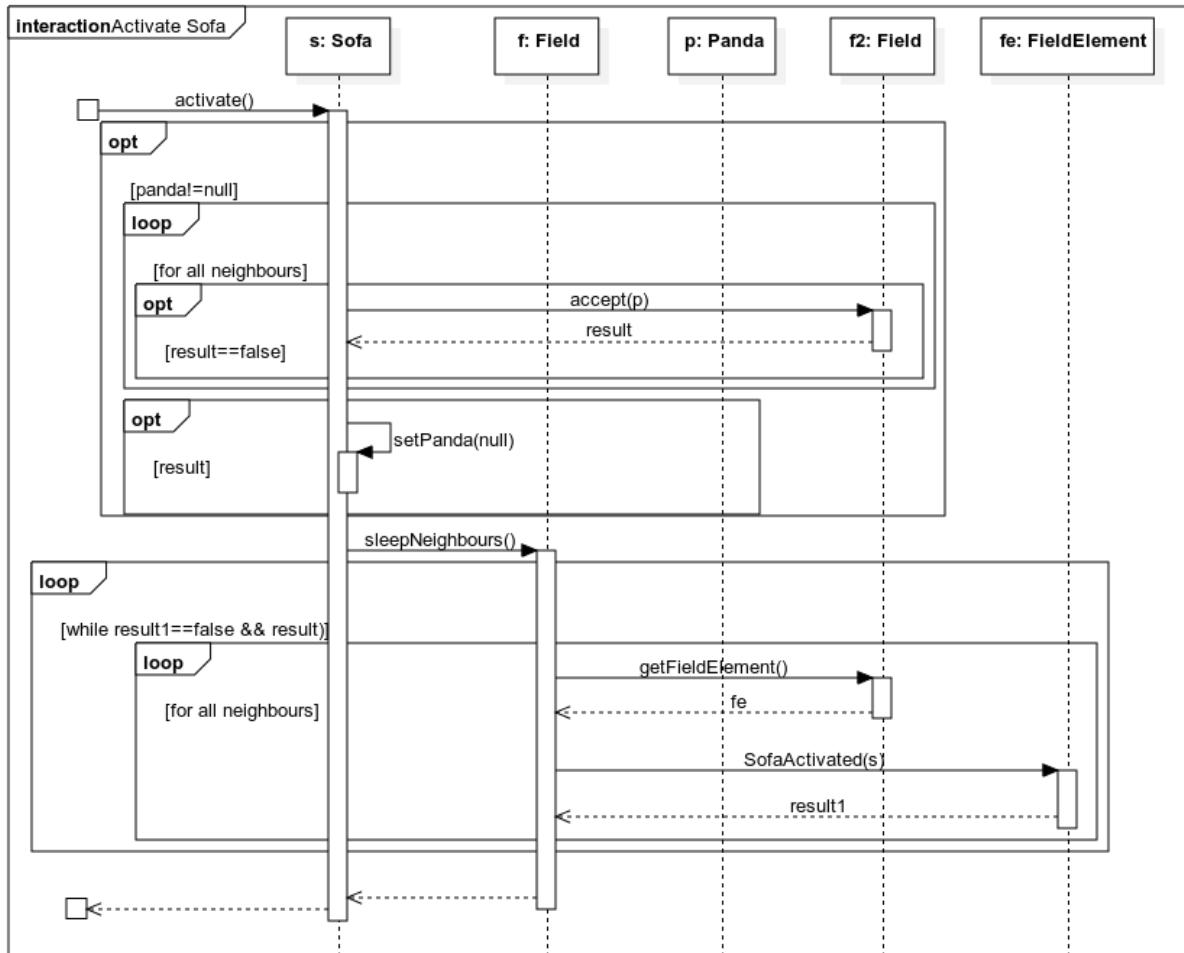
4.4.11 Activate GamblingMachine



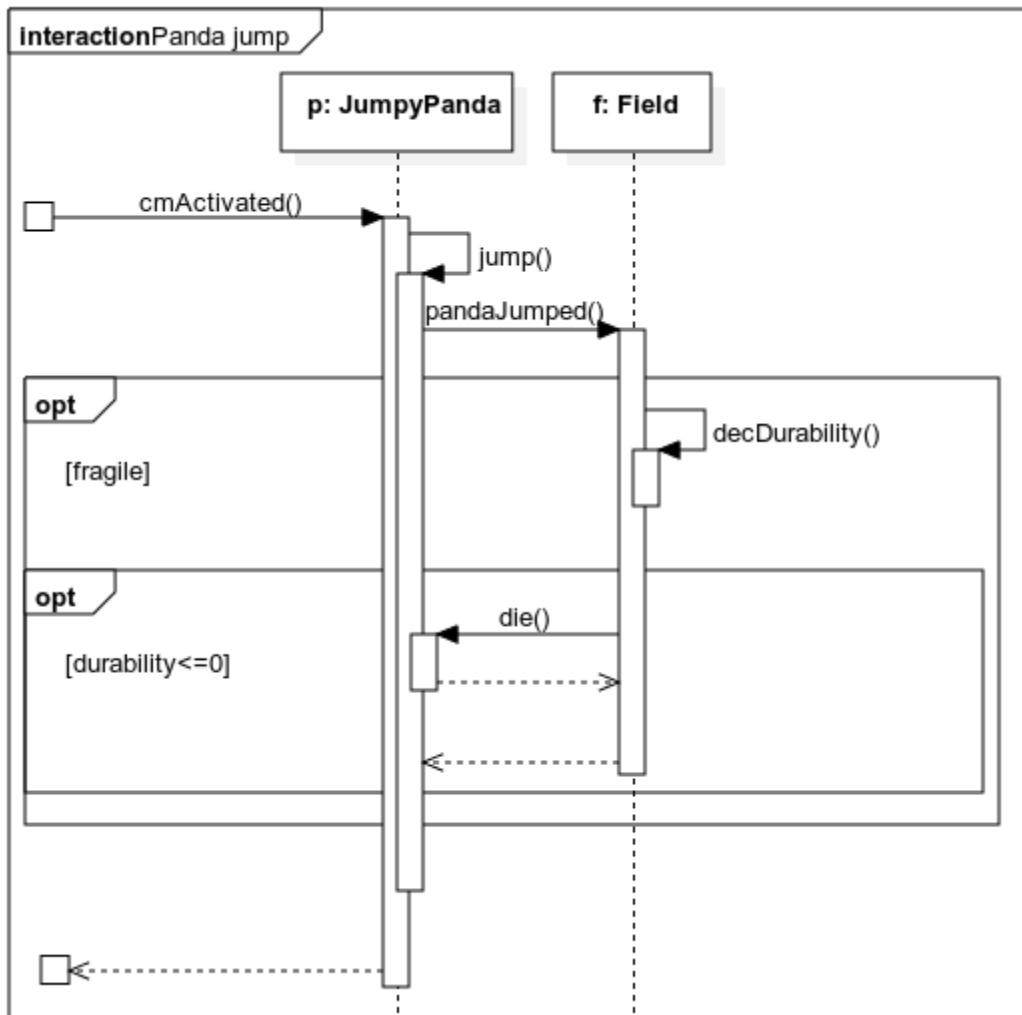
4.4.12 Activate ChocolateMachine



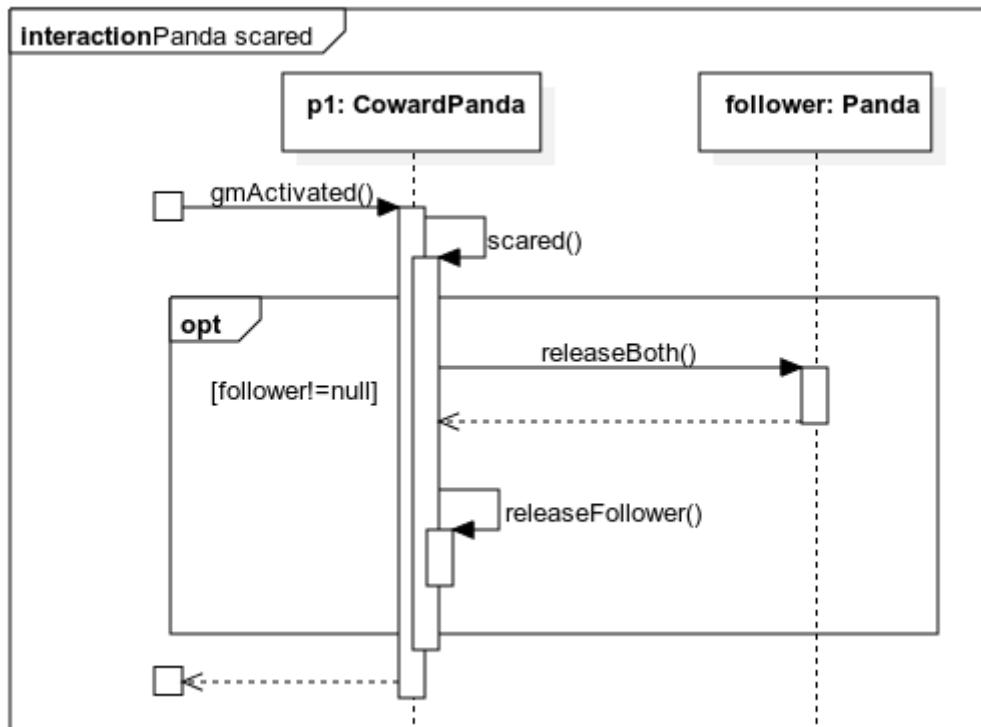
4.4.13 Activate Sofa



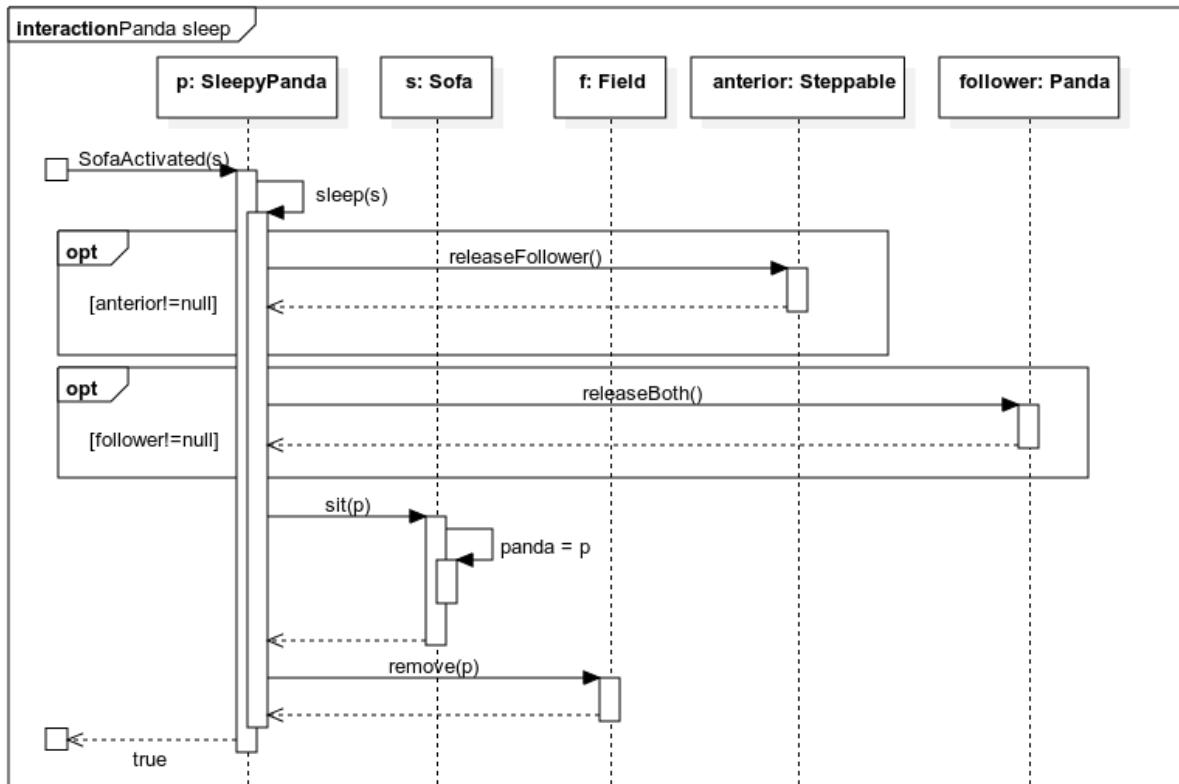
4.4.14 Panda jump



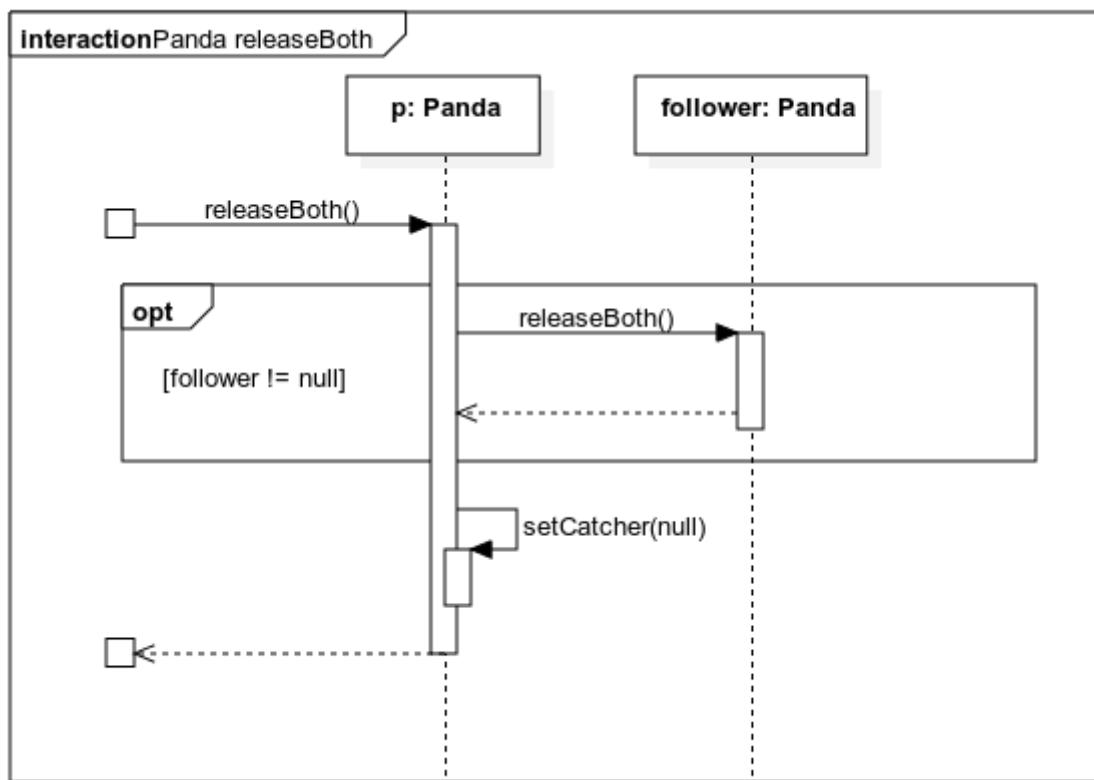
4.4.15 Panda scared



4.4.16 Panda sleep

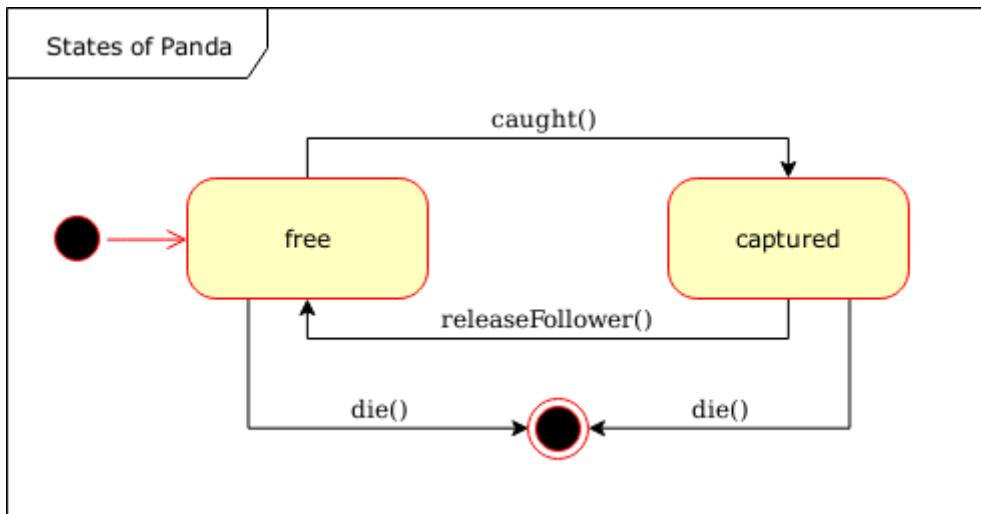


4.4.17 Panda releaseBoth



4.5 State-chartok

4.5.1 States of Panda



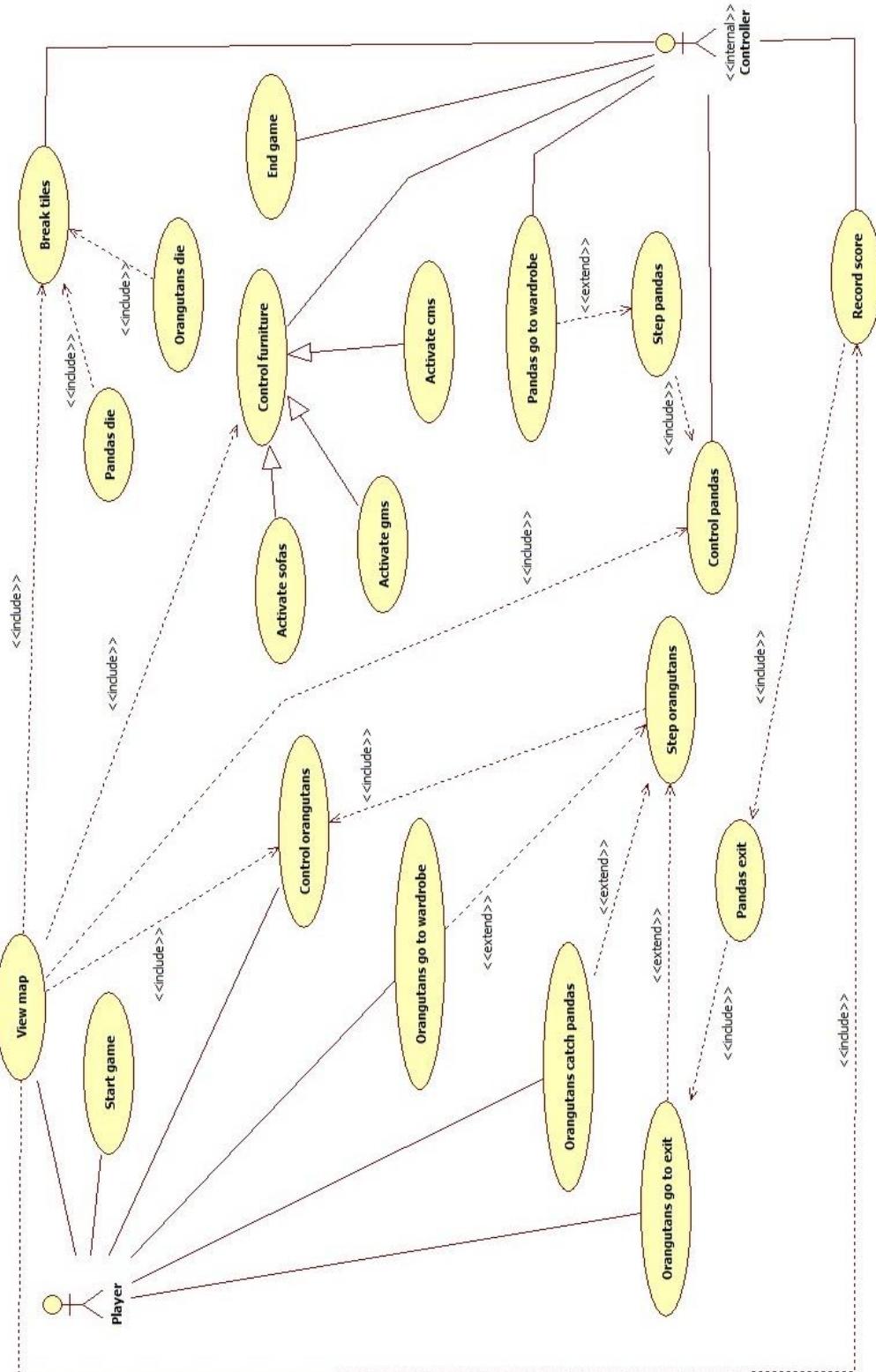
4.6 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2019.02.27. 19:00	1 óra	Borsodi, Bódi, Nyári, Hegedüs	<p>Értekezlet témája: A 3. feladat hibáinak megértése, megbeszélése, a javítási munkák szétosztása a csapattagok között.</p> <p>Döntés:</p> <ul style="list-style-type: none"> • Szekvenciadiagramok javítása: Borsodi, Nyári • Osztálydiagram javítása: Hegedüs, Borsodi • A közös munkát támogató eszközök karbantartása, optimalizálása: Bódi • A dokumentum konzisztenciájának biztosítása, az abban lévő hibák javítása: Bódi, Nyári, Hegedüs
2019.02.27. 20:00	1 óra	Borsodi	Az osztálydiagram javítása
2019.02.27. 20:00	1 óra	Hegedüs	Az osztálydiagram javítása
2019.02.28. 16:00	1 óra	Nyári	Szekvenciadiagramok javítása
2019.02.28. 19:00	1 óra	Hegedüs	Az osztálydiagram újabb hibáinak javítása, pontosítás.
2019.02.28. 19:00	2 óra	Borsodi	Szekvenciadiagramok javítása
2019.03.01. 10:30	0,5 óra	Hegedüs	A dokumentum előkészítése, a hibátlan részek átmásolása az előző feladatból.
2019.03.01. 19:00	2 óra	Bódi	Szekvenciadiagramok hiányzó részeinek befejezése, exportálása és beillesztése a dokumentumba.
2019.03.02. 02:00	1 óra	Nyári	Az osztályok első felének leírása.
2019.03.02. 14:00	1 óra	Hegedüs	Szekvenciadiagramok formai hibáinak javítása, a hibás szekvenciadiagramok eltávolítása a dokumentumból és javított változatok beszúrása.
2019.03.02. 07:00	0,5 óra	Hegedüs	Az Activateable, a ChocolateMachine és a FieldElement osztályok leírásának elkészítése.
2019.03.03. 18:00	2 óra	Nyári	Az osztályok második felének leírása
2019.03.03. 20:00	1 óra	Borsodi	A dokumentum első felének végleges formázása, az előforduló hibák javítása
2019.03.03 21:00	1 óra	Hegedüs	A dokumentum második felének végleges formázása, az előforduló hibák javítása.

5. Szkeleton tervezése

5.1 A szkeleton modell valóságos use-case-ai

5.1.1 Use-case diagram



5.1.2 Use-case leírások

Use-case neve	View map
Rövid leírás	A játékos megtekinti a pályát.
Aktorok	Player
Forgatókönyv	<p>1. A rendszer „kirajzolja” a pályát, az azon lévő objektumokat, megjeleníti azok aktuális állapotát, továbbá az aktuális pontszámot/pontszámokat.</p> <p>2. A játékos mindezt megtekinti, döntése során felhasználja az ezekből nyert információkat.</p>

Use-case neve	Start game
Rövid leírás	A játékos elindítja a játékot.
Aktorok	Player
Forgatókönyv	<p>1. A játékos elindítja a játékot.</p> <p>2. A játékos kiválasztja a neki legjobban tetsző pályát.</p> <p>3. A rendszer elvégzi az inicializációt.</p> <p>4. A játék elindul.</p>

Use-case neve	Control orangutans
Rövid leírás	A játékos irányítja a játékban lévő orángutánokat.
Aktorok	Player
Forgatókönyv	<p>1. A játékos parancsot ad ki a soron lévő orángutánnak.</p> <p>2. Az orángután teljesíti a parancsot.</p>
Alternatív forgatókönyv	<p>2.A.1. Az orángután nem tudja teljesíteni a parancsot</p> <p>2.A.2. A játékos újabb parancsot/parancsokat ad ki az orángutánnak egészen addig ameddig az végre nem tudja hajtani valamelyiket.</p>
Alternatív forgatókönyv	<p>1.B.1. A játékos nem tud parancsot adni az orángutánnak, mert egyértelműen látható, hogy az semmit nem tud végrehajtani aktuális állapotában.</p>

Use-case neve	Step orangutans
Rövid leírás	Az orángutánok lépkednek a pályán.
Aktorok	Player
Forgatókönyv	1. Az orángután a játékos parancsának megfelelően lép.
Alternatív forgatókönyv	1.A.1. Az orángután valamilyen kényszer miatt nem tudja teljesíteni a játékos parancsának megfelelő lépést.

Use-case neve	Orangutans go to wardrobe
----------------------	---------------------------

Rövid leírás	Az orangután egy olyan mezőre lép, melyen szekrény van, így bemegy a szekrénybe, majd kijön egy másikon.
Aktorok	Player
Forgatókönyv	<ol style="list-style-type: none"> Az orangután a játékos parancsának megfelelően lép. Az orangután egy olyan mezőre lépett, amin szekrény van. Az orangután kijön egy másik szekrényből. A körök során a pandák egyesével követik az orangutánt a szekrényeken át, feltéve, hogy az orangutánnak vannak pandái.
Alternatív forgatókönyv	1.A.1. Az orangután valamelyen kényszer miatt nem tud belépni a szekrénybe.

Use-case neve	Orangutans catch pandas
Rövid leírás	Az orangután elkap egy pandát.
Aktorok	Player
Forgatókönyv	<ol style="list-style-type: none"> Az orangután a játékos parancsának megfelelően lép. Az orangután olyan mezőre lépett, melyen panda van. Az orangután helyet cserél az előbb említett pandával és ezzel el is kapja azt. A panda így már követni fogja az orangutánt.
Alternatív forgatókönyv	1.A.1. Az orangután valamelyen kényszer miatt nem tudja elkapni a pandát (pl. másik orangután már elkapta azt).

Use-case neve	Orangutans go to exit
Rövid leírás	Az orangután kimegy a kijáraton.
Aktorok	Player
Forgatókönyv	<ol style="list-style-type: none"> Az orangután a játékos parancsának megfelelően lép. Az orangután olyan mezőre lépett, mely kijárat. Az orangután megjelenik a bejáratnál. Ha van követője az orangutánnak, szól neki, hogy kilépett.
Alternatív forgatókönyv	1.A.1. Az orangután valamelyen kényszer miatt nem tudja teljesíteni a játékos parancsának megfelelő lépést.

Use-case neve	Pandas exit
Rövid leírás	A panda kimegy a kijáraton, mert egy orangután kivezeti.
Aktorok	Player
Forgatókönyv	<ol style="list-style-type: none"> A panda utasítást kap, hogy a befogó orangutánja kiment a kijáraton. A panda kilép a pályáról. Ha a pandának van követője, szól neki, hogy kilépett.

Use-case neve	Record score
Rövid leírás	Az orangután számára a rendszer pontot ír jóvá a kivezetett pandákért cserébe.
Aktorok	Controller
Forgatókönyv	<ol style="list-style-type: none"> Kilép a pályáról egy panda Az orangután pontot kap a kivezetett pandáért.

Use-case neve	Control pandas
Rövid leírás	A játék (Controller) irányítja a pandákat.

Aktorok	Controller
Forgatókönyv	1. A játék (Controller) parancsot ad ki az egyik pandának, hogy lépjön. 2. A panda teljesíti a parancsot.
Alternatív forgatókönyv	1.A.1. A játék (Controller) nem ad parancsot a pandának, mert az semmit nem tud végrehajtani aktuális állapotában.

Use-case neve	Step pandas
Rövid leírás	A pandák lépkednek a pályán.
Aktorok	Controller
Forgatókönyv	1. Az orángután a játék (Controller) utasításának megfelelően lép valahova.

Use-case neve	Pandas go to wardrobe
Rövid leírás	A panda egy olyan mezőre lép, melyen szekrény van, így bemegy a szekrénybe, majd kijön egy másikon.
Aktorok	Controller
Forgatókönyv	1. A panda a játék (Controller) parancsának megfelelően lép. 2. A panda egy olyan mezőre lépett, amin szekrény van. 3. A panda kijön egy másik szekrényből.

Use-case neve	Control furniture
Rövid leírás	A játék (Controller) irányítja a berendezéseket.
Aktorok	Controller
Forgatókönyv	1. A játék (Controller) minden körben véletlenszerűen parancsot ad ki az egyik (vagy több) berendezésnek. 2. Az adott berendezés(ek) teljesíti(k) a parancsot.

Use-case neve	Activate sofas
Rövid leírás	A játék (Controller) aktiválja a fotelt.
Aktorok	Controller
Forgatókönyv	1. A fotel aktiválódik, hirdetni kezdi a szomszédos mezők felé, hogy bele lehet ülni. 2. Ha a szomszédos mezőn áll egy fáradt panda, akkor az lehet, hogy beleül egy körre.

Use-case neve	Activate gms
Rövid leírás	A játék (Controller) aktiválja a játékgépet.
Aktorok	Controller
Forgatókönyv	1. A játékgép aktiválódik, elkezd csilingelni. 2. A szomszédos mezőkön lévők hallják a csilingelést. 3. Ha a szomszédos mezőn ijedős panda áll, akkor az elengedi a sorban mögötte állók kezét.

Use-case neve	Activate cms
Rövid leírás	A játék (Controller) aktiválja a csokiautomatát.
Aktorok	Controller

Forgatókönyv	1. A csokiautomata aktiválódik, elkezd sípolni. 2. A szomszédos mezőkön lévők hallják a sípolást. 3. Ha a szomszédos mezőn ugrós panda áll, akkor az ugrik egyet a sípolás miatt.
---------------------	---

Use-case neve	Break tiles
Rövid leírás	Az emelet törékeny csempéi gyengülnek, majd beszakadnak, ha sokszor lépnek rájuk, vagy sokat ugranak rajtuk.
Aktorok	Controller
Forgatókönyv	1. Egy lépkedni képes objektum rálép a törékeny csempére, vagy egy ugrani képes objektum ugrik egyet rajta 2. A törékeny csempe élettartama csökken eggyel.
Alternatív forgatókönyv	3.A.1. Ha a csökkenés hatására a törékeny csempe élettartama 0 lesz, akkor az beszakad.

Use-case neve	Orangutans die
Aktorok	Controller
Rövid leírás	Az orangután meghal, ha egy csempe leszakad alatta.
Forgatókönyv	1. Az orangután egy beszakadó vagy már korábban beszakadt csempére lép. 2. Az orangután meghal. 3. Az orangutánt követő sor felbomlik.

Use-case neve	Pandas die
Aktorok	Controller
Rövid leírás	A panda meghal, ha egy csempe leszakad alatta.
Forgatókönyv	1. A panda egy beszakadó vagy már korábban beszakadt csempére lép. 2. A panda meghal. 3. A pandát követő sor felbomlik.

Use-case neve	End game
Rövid leírás	A játék véget ér.
Aktorok	Controller
Forgatókönyv	1. A játék (Controller) ellenőrzi, hogy a pályán még vannak-e orangutánok és pandák, továbbá azt, hogy vezet-e még út valamelyik orangutántól valamelyik pandához és a kijárathoz. 2. Ha az előző feltételek egyike nem teljesül, akkor a játék befejeződik.
Alternatív forgatókönyv	3.A.1. Győznek az orangutánok (tehát a játékos). 3.B.1. Győznek a pandák (tehát a gép).

5.2 A szkeleton kezelői felületének terve, dialógusok

A szkeleton program indulásakor létrehoz egy üres pályát, aztán utasításokat fogad a felhasználótól a standard inputon, amikre reagál a standard outputon.

5.2.1 A felhasználói parancsok:

help [command]: Kiírja az összes használható parancsot. Egy konkrét parancsszót mögé írva pedig kiírja, hogy mit csinál az adott parancs.

list [type]: Kiírja az összes type típusú objektumot a pályán, és ha FieldElement, akkor azt is, hogy melyik mezőn áll.

Type lehetséges értékei:

- p: Panda
- o: Orangutan
- w: Wardrobe
- s: Sofa
- cm: ChocolateMachine
- gm: GamblingMachine
- e: Exit
- f: Field

neighbours [Field]: A kapott mező összes szomszédját kiírja, és ha egy mező foglalt, akkor a rajta álló FieldElement típusát is.

step [Field]: Az éppen körön lévő orángután lépést kísérel meg a kapott mezőre.

newgame: Befejezi a játékot, és alaphelyzetbe állítja a pályát.

start: Elindítja a játékot, elkezdődik az első kör.

add [type] [Field]: Hozzáad egy type típusú FieldElementet a kapott mezőhöz, feltéve, hogy a mező üres. A type a list parancsnál használt rövidítéseket ismeri fel, kivéve a panda esetén, mert ekkor az alábbi rövidítéseket használhatjuk:

- sp: SleepyPanda
- jp: JumpyPanda
- cp: CowardPanda

remove [FieldElement]: A kapott FieldElementet eltávolítja a pályáról.

init: Inicializálja a pályát FieldElementekkel, egy előre beállított minta szerint. Használandó, ha a felhasználó nem szeretné őket egyesével hozzáadni.

5.2.2 A program kimenete/válaszüzenetei:

A program a kezdő, üres pálya létrehozása után utasításokra vár, amit a sor elején egy > szimbólum jelez. Ha a játék elildul, a program jelzi melyik orangután van soron (a példában o1) az alábbi módon: **o1>**

A felhasználói parancsokra a rendszer **success**, vagy **error** üzenetet ír a konzolra, attól függően, hogy a parancs sikeresen végrehajtódott, vagy sem.

Az error üzenetben specifikálja a hiba okát:

error: unknown command - a beütött parancs nem létezik.

error: bad parameters – a parancs paramétereinek hibás, vagy nem lehetséges a művelet.

Ha egy játéknak vége, a program kiírja a még játékban lévő orangutánokat pontszámukkal együtt, majd újabb parancsokat vár.

Minden függvény meghívásakor, az kiíródik a kimenetre az alábbi módon:
[osztály neve].[függvény neve](paraméterek)

A kiírás hierarchikusan történik, minden függvény az őt meghívó függvénytől beljebb helyezkedik el, például egy játékautomata aktiválódásakor így (ha csak egy szomszédos mező van):

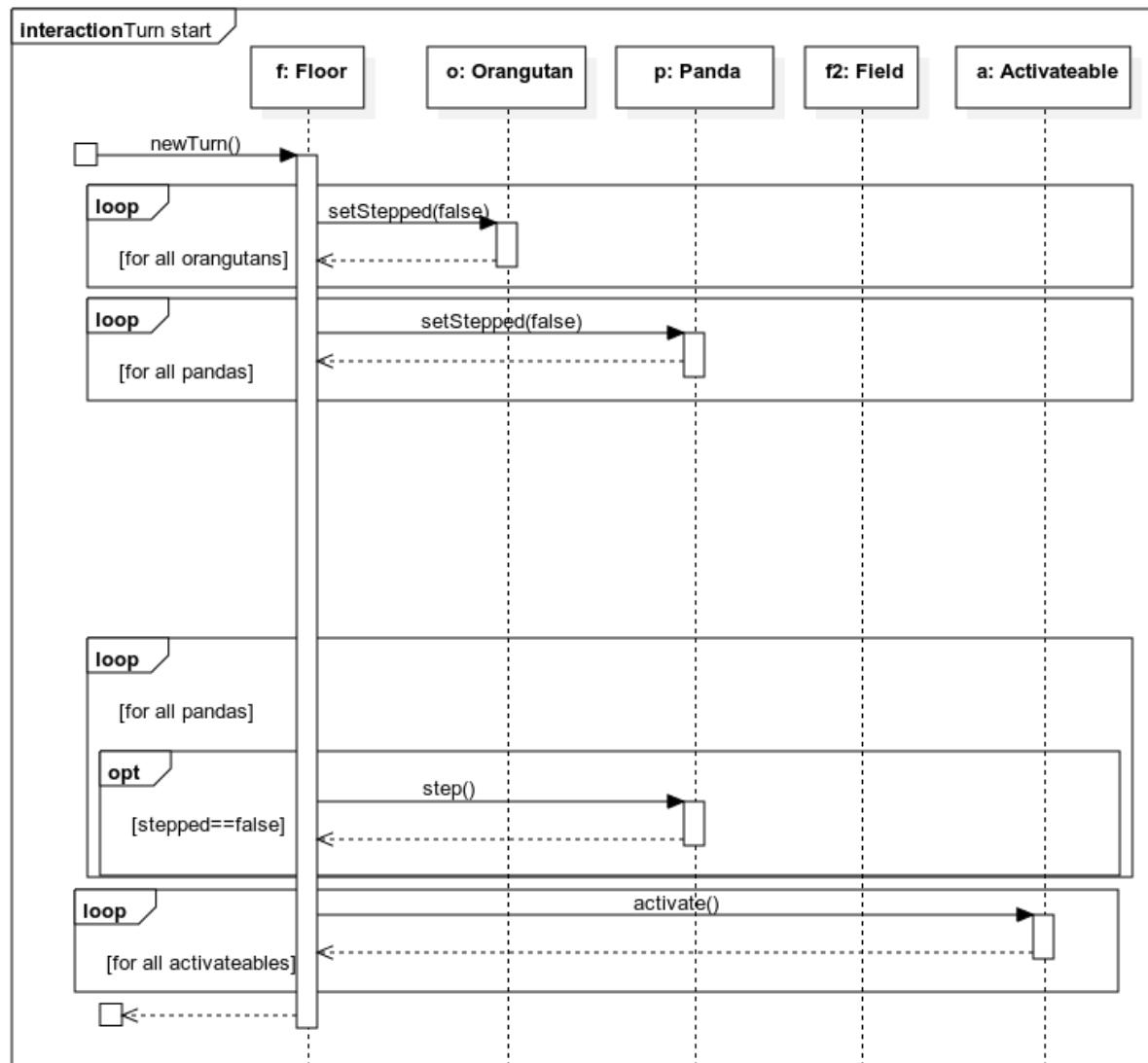
```
GamblingMachine.activate()
    GamblingMachine.random()
        Field.scareNeighbours()
            Field.getFieldElement()
                FieldElement.gmActivated()
```

Példa egy új játék indítására:

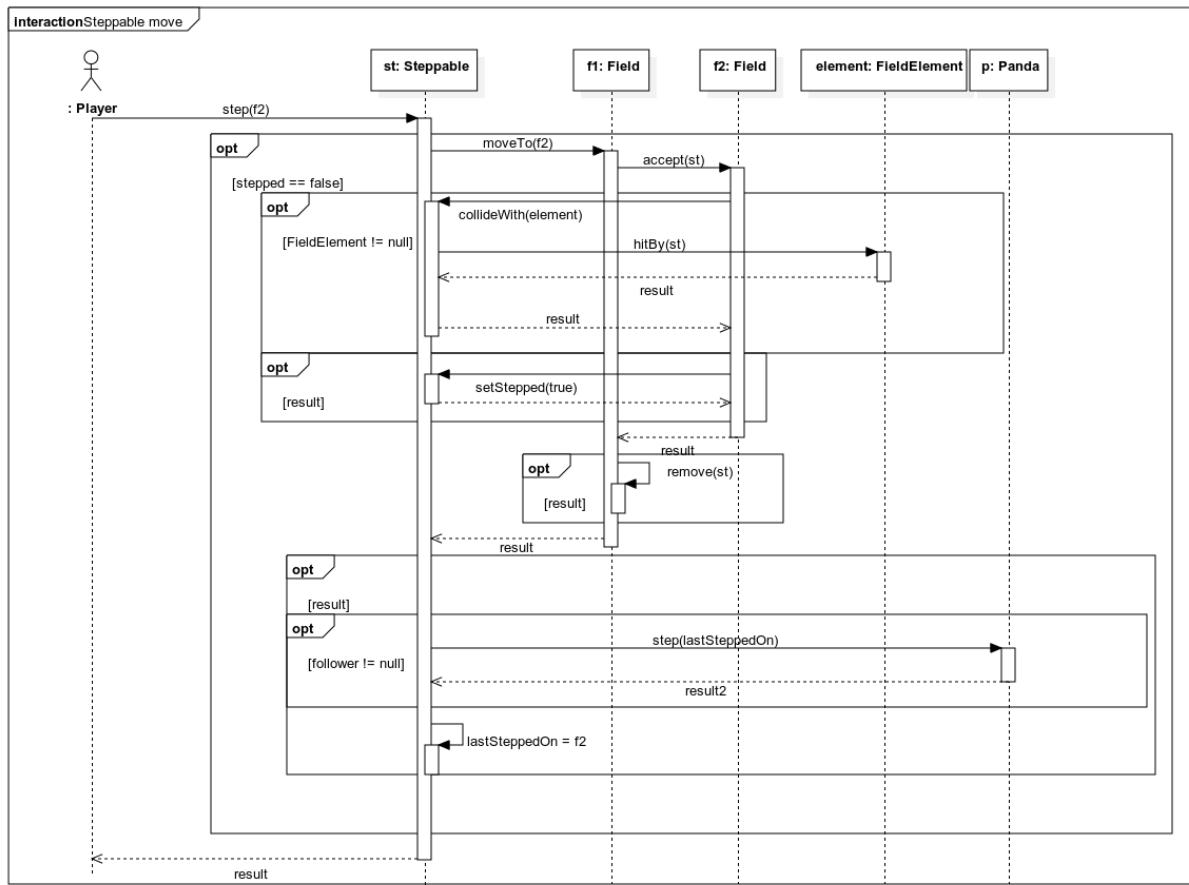
```
>newgame
success
>init
success
>start
success
o1>step f1
Orangutan.step(Field)
    Field.moveTo(Field)
    ...
success
o2>
```

5.3 Szekvencia diagramok a belső működésre

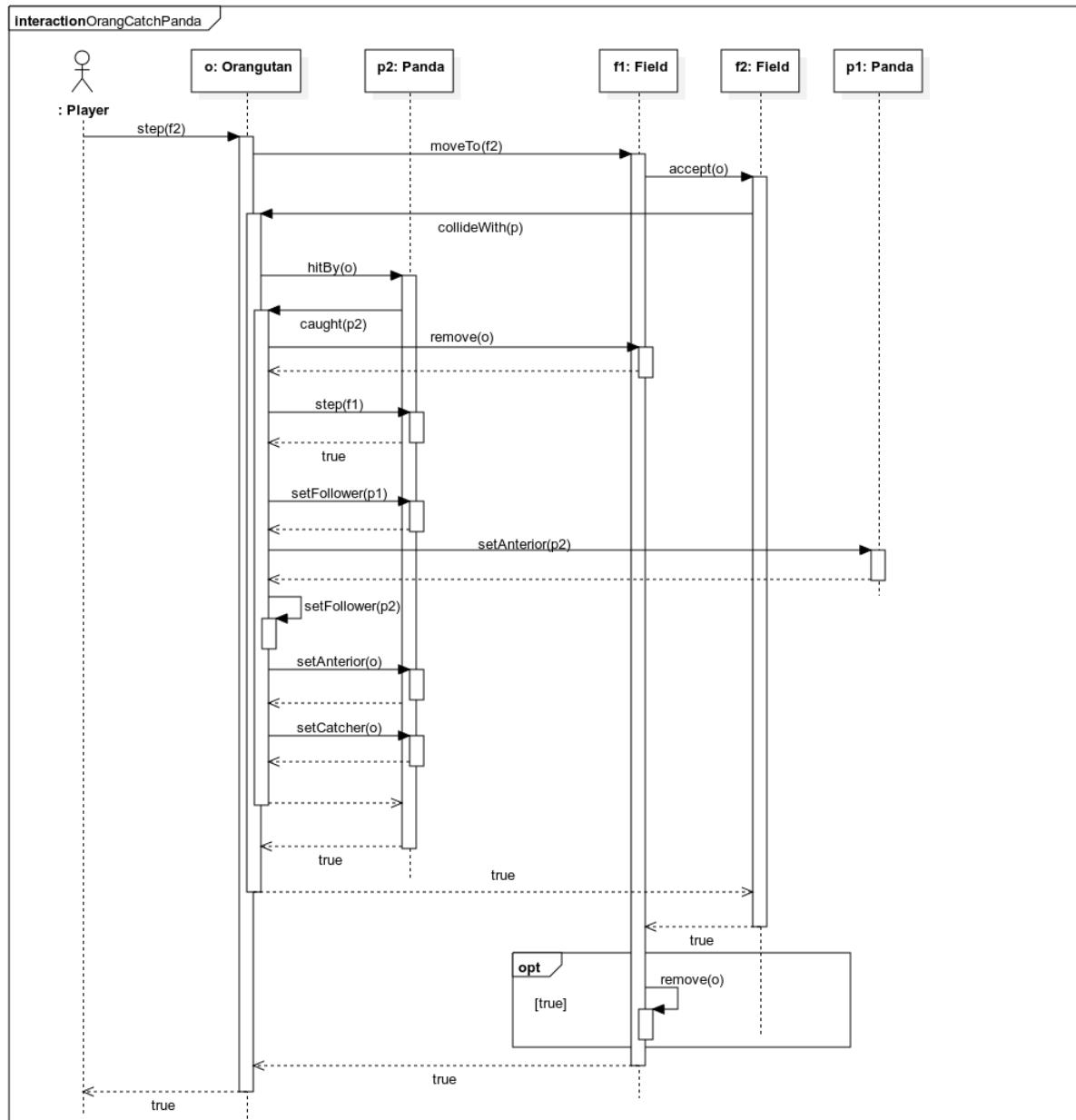
5.3.1 Turn start



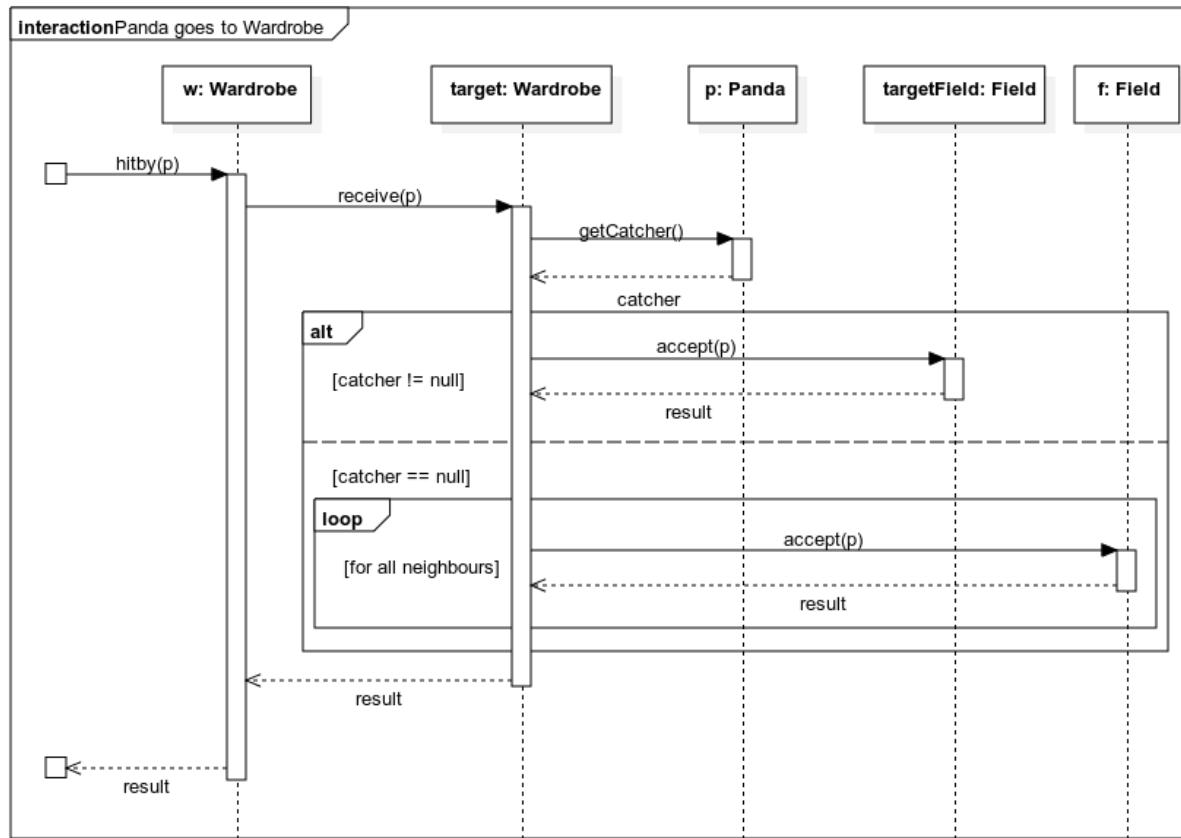
5.3.2 Steppable move



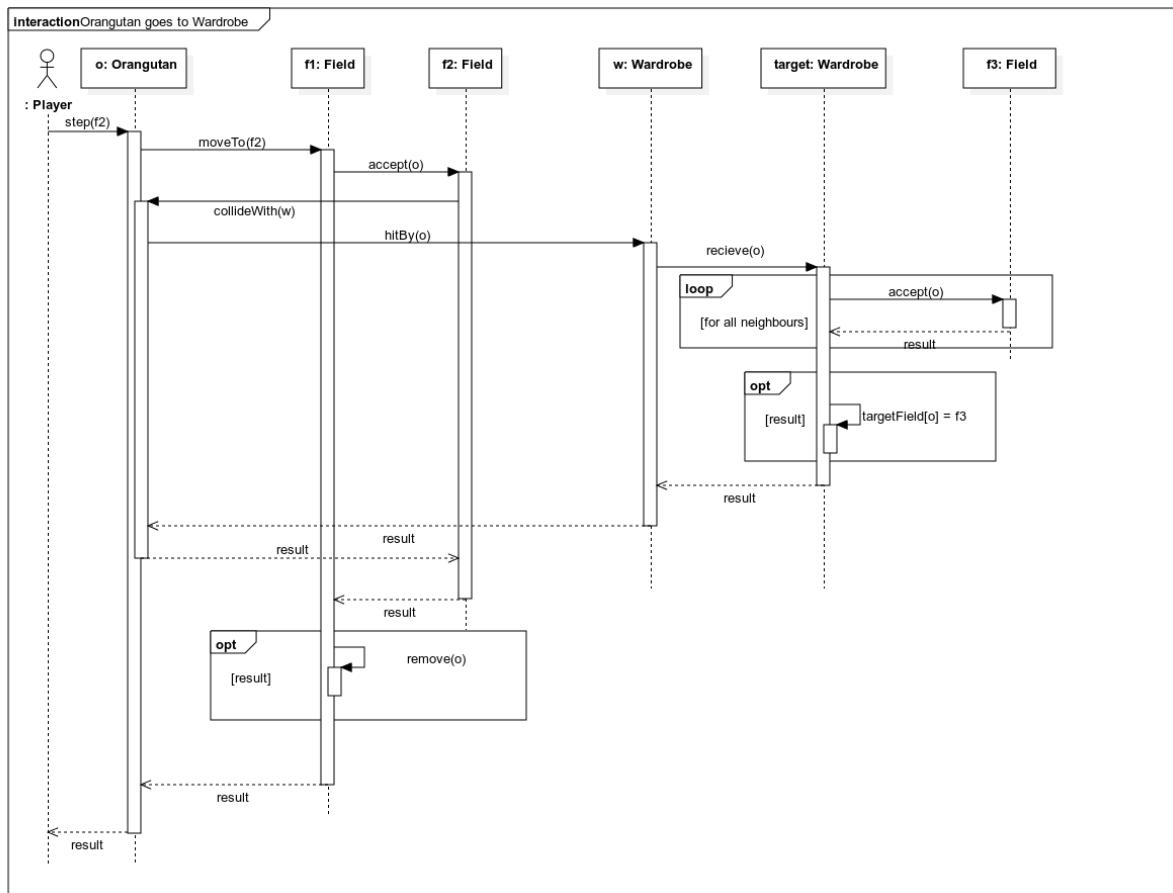
5.3.3 Orangutan catch panda



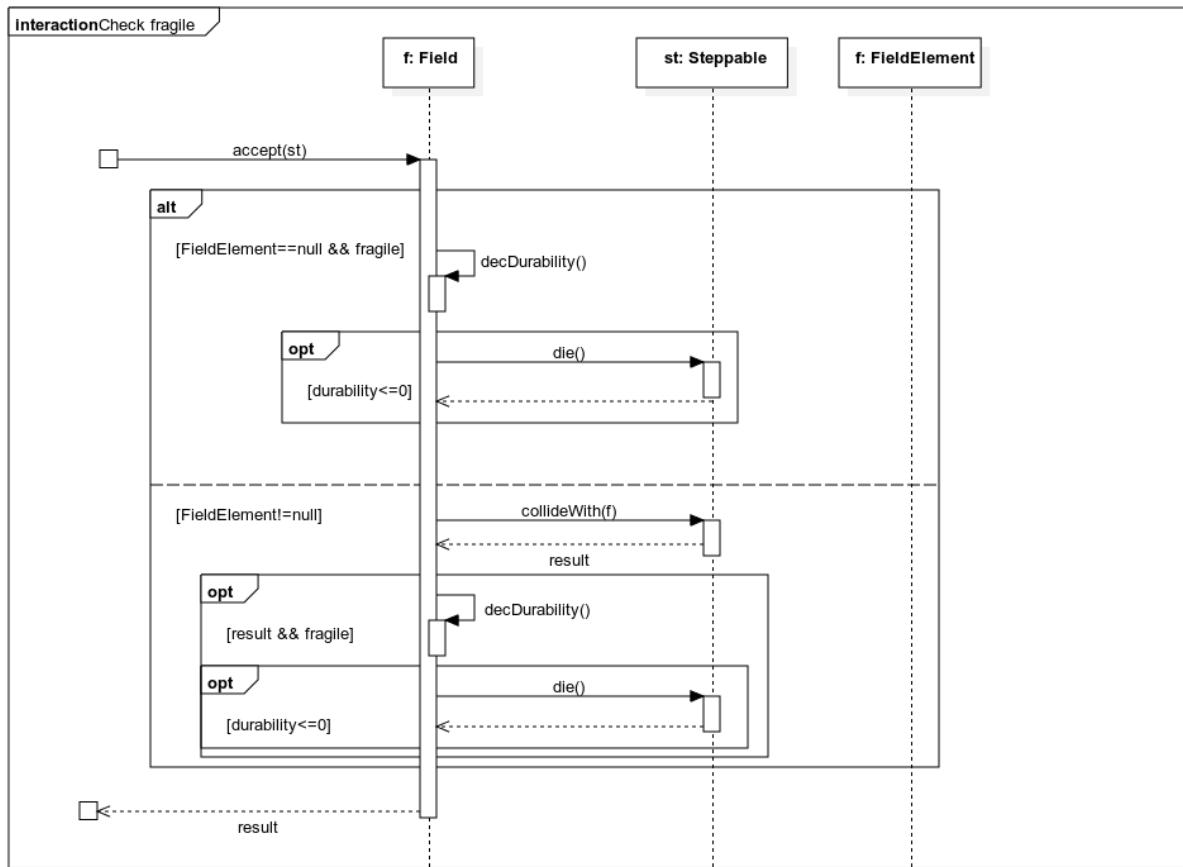
5.3.4 Panda goes to wardrobe



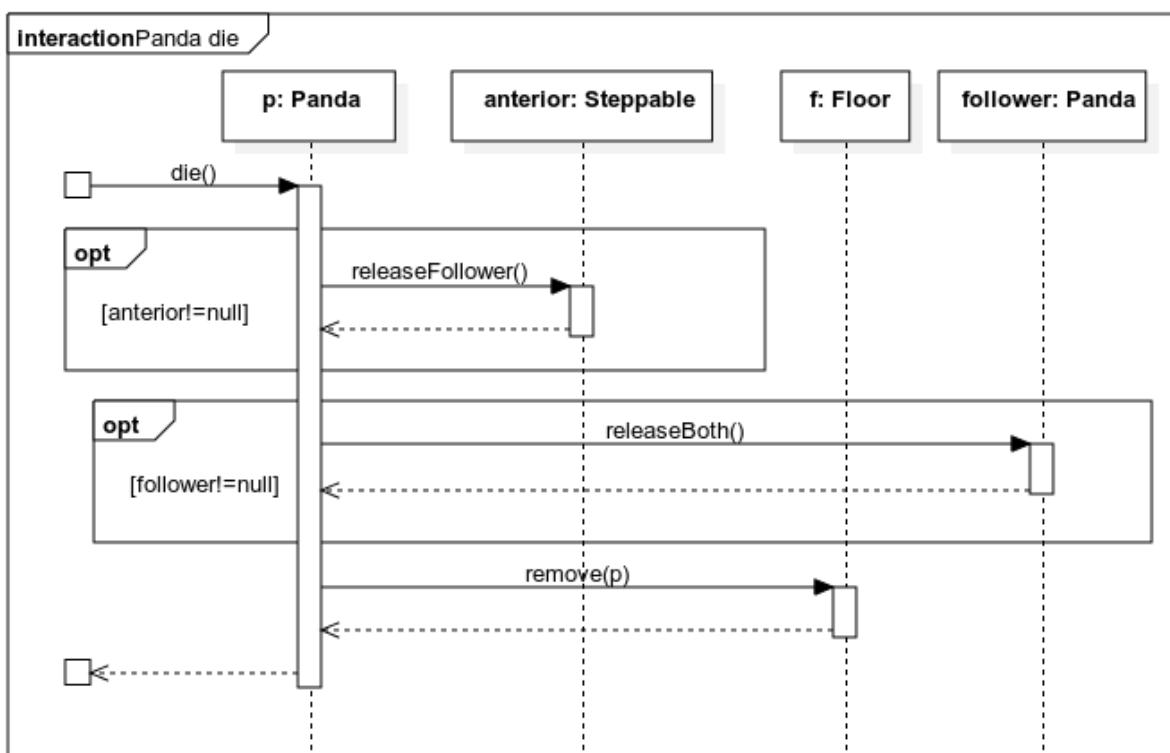
5.3.5 Orangutan goes to wardrobe



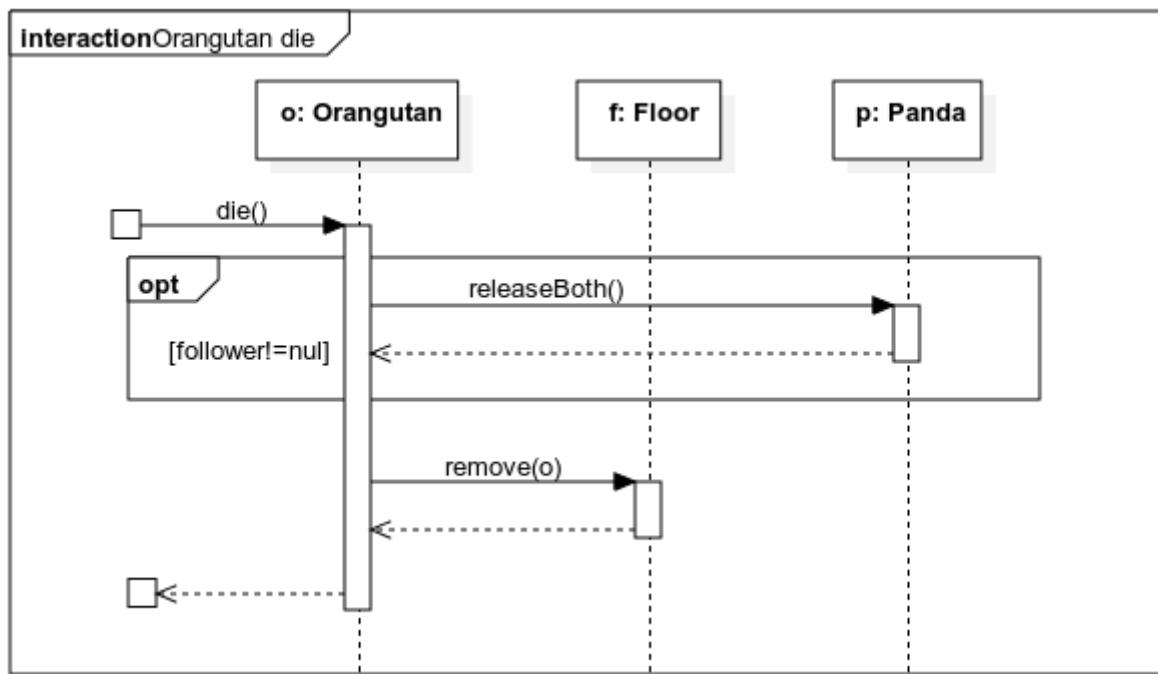
5.3.6 Check fragile



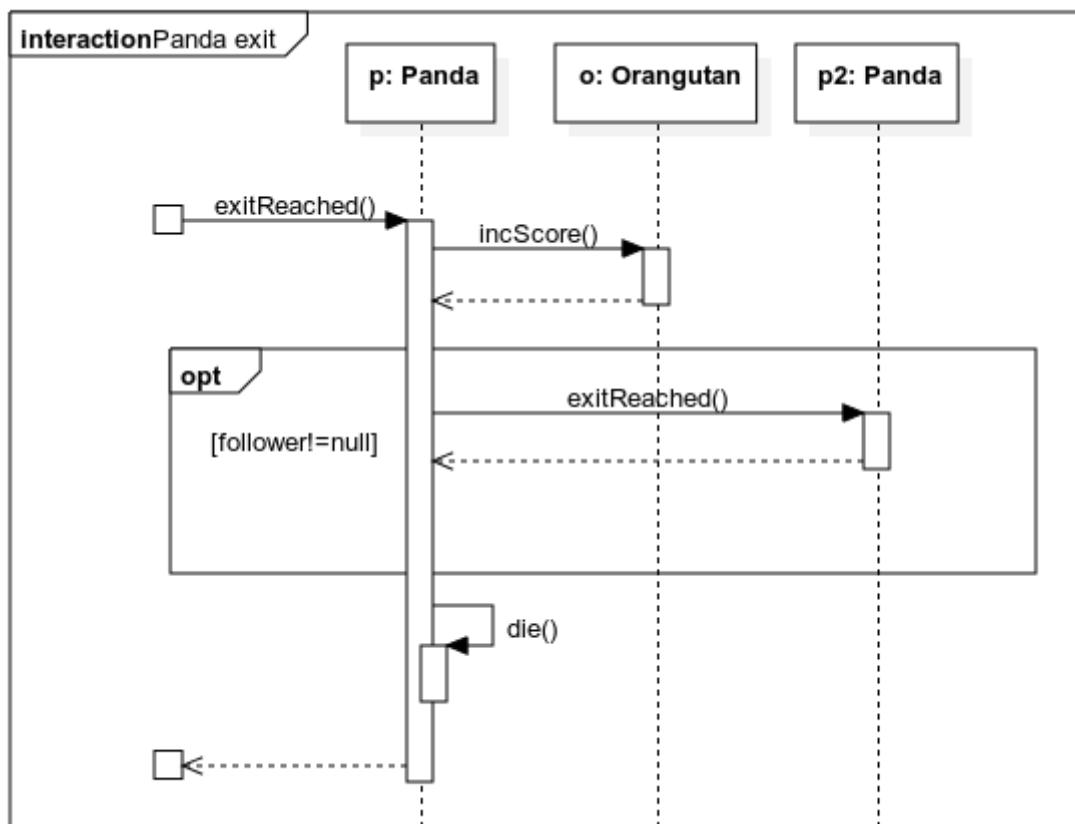
5.3.7 Panda die



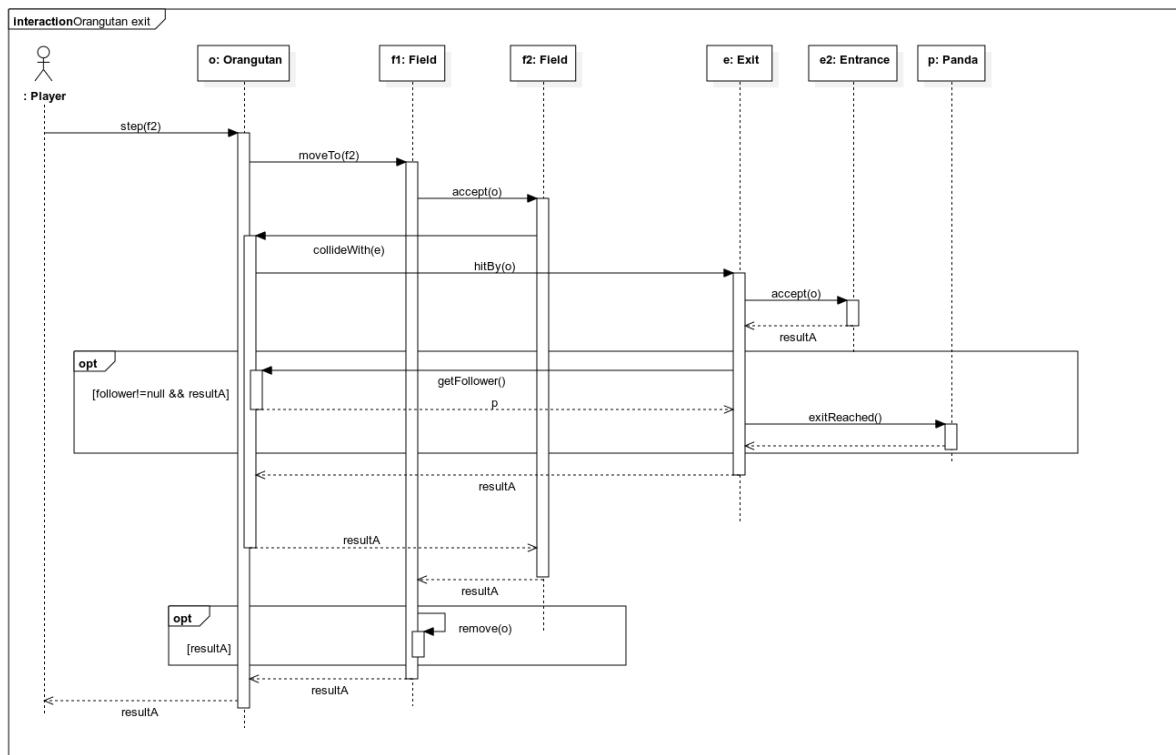
5.3.8 Orangutan die



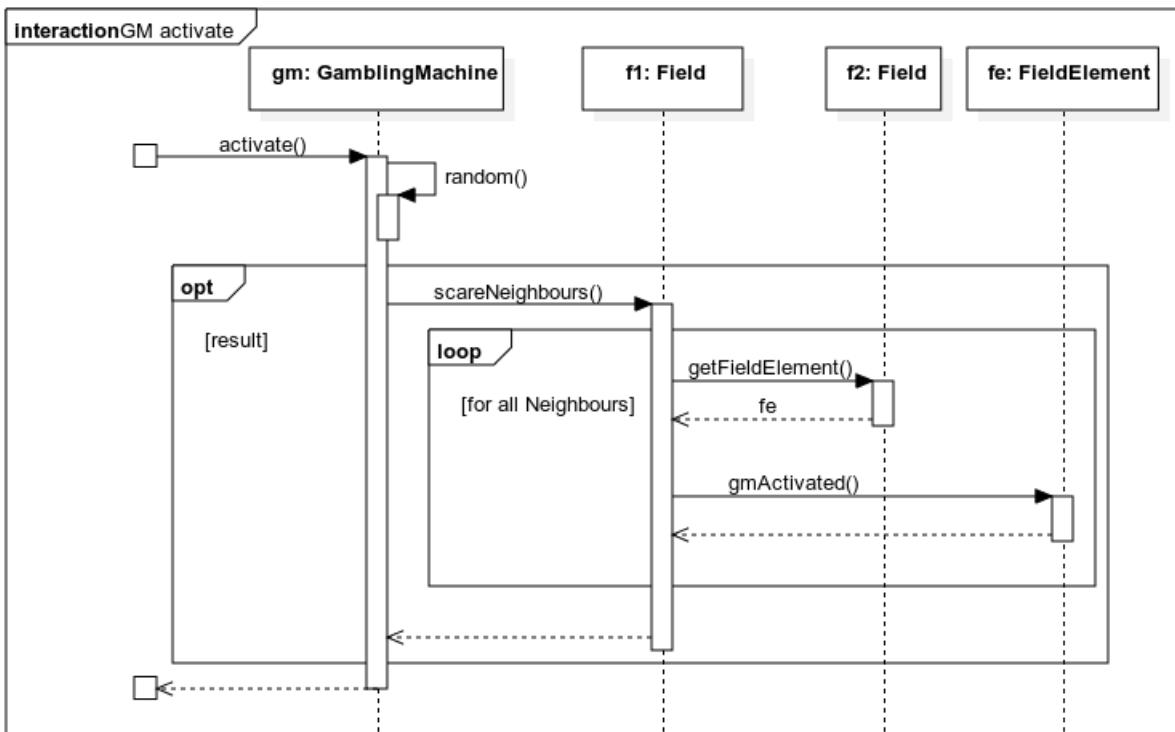
5.3.9 Panda exit



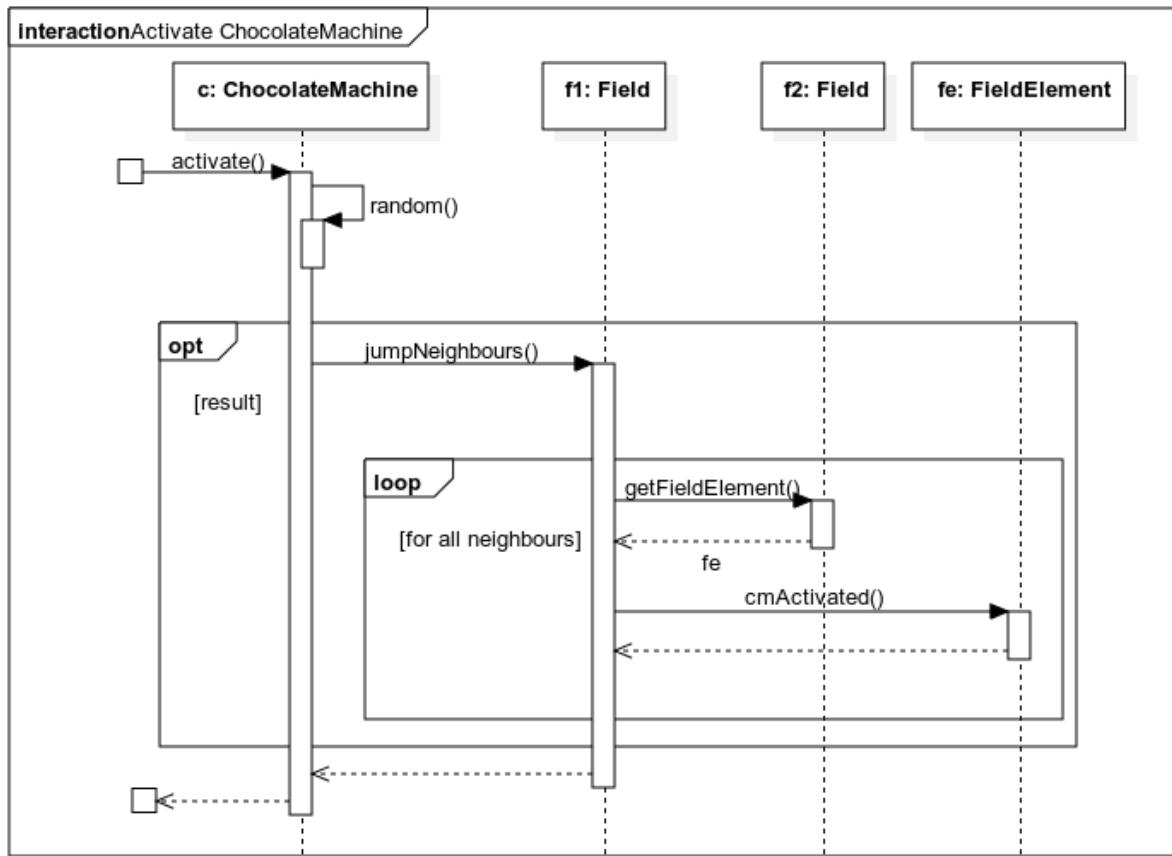
5.3.10 Orangutan exit



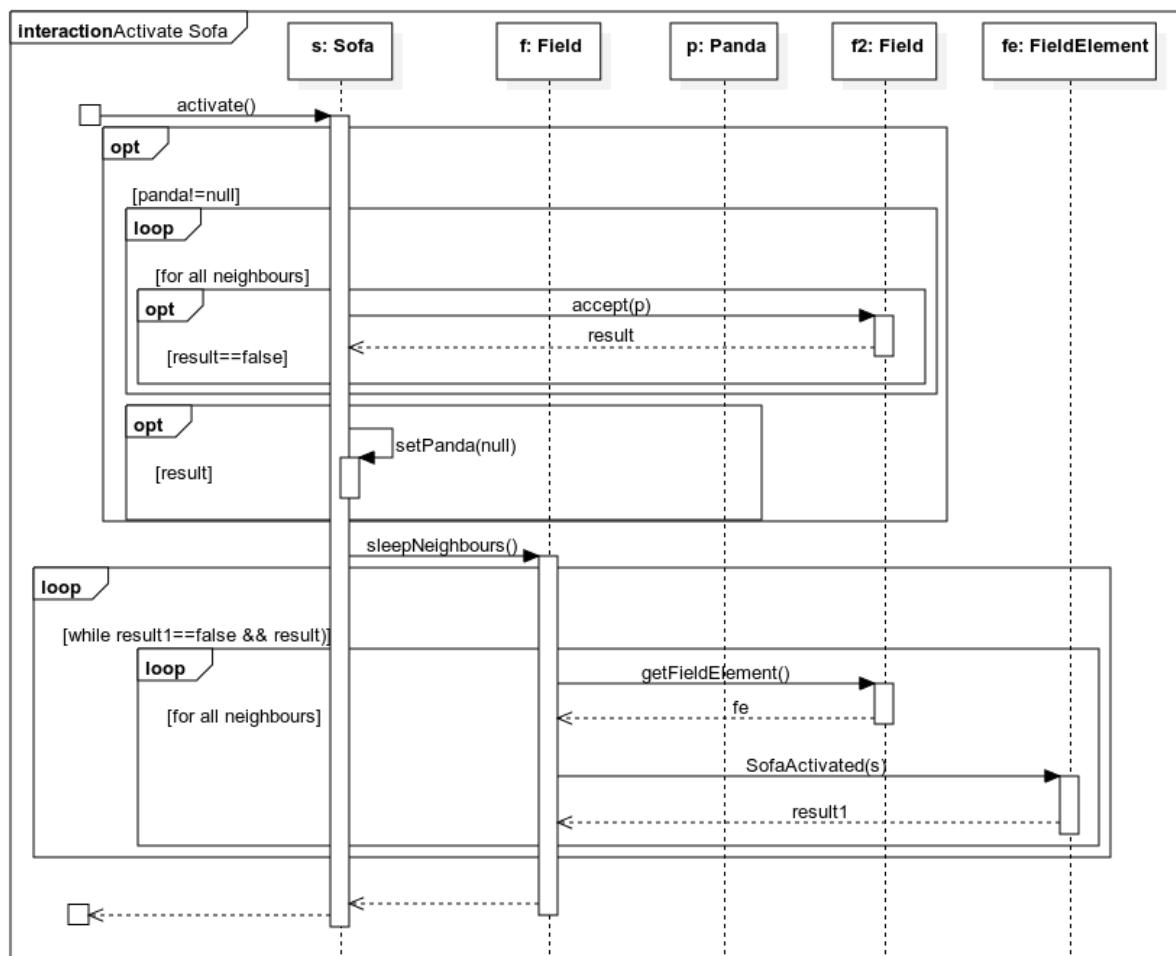
5.3.11 Activate gambling machine



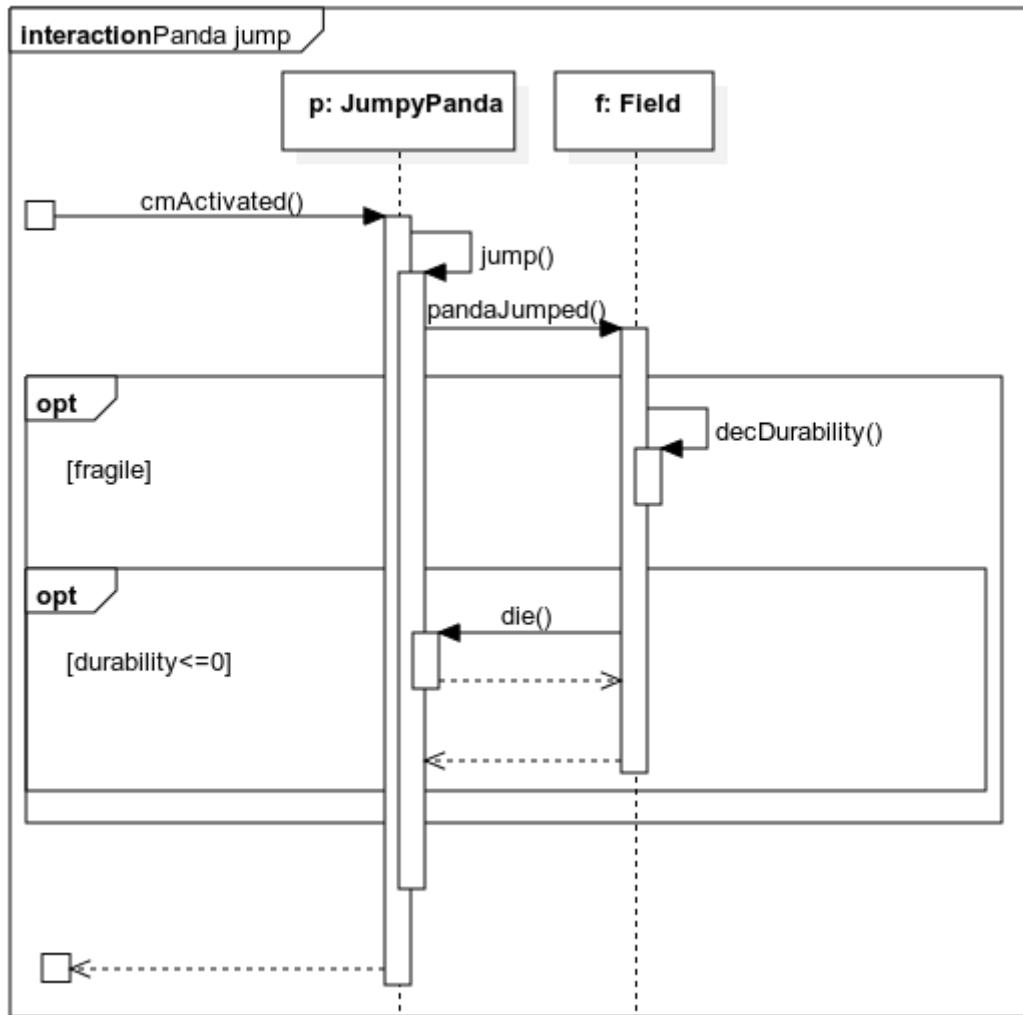
5.3.12 Activate chocolate machine



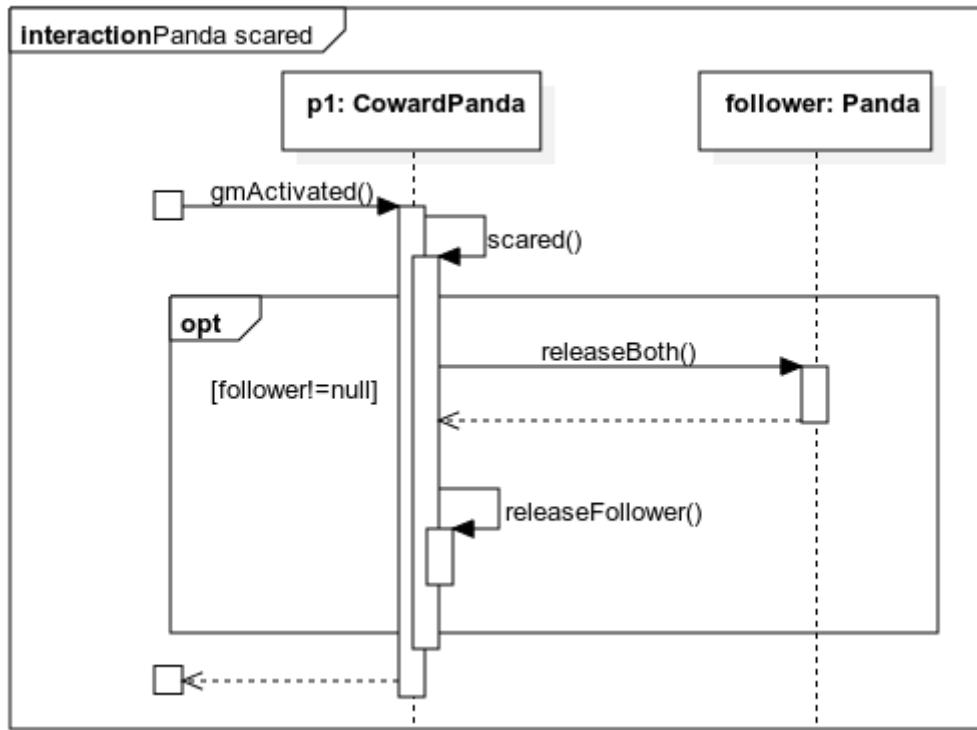
5.3.13 Activate sofa



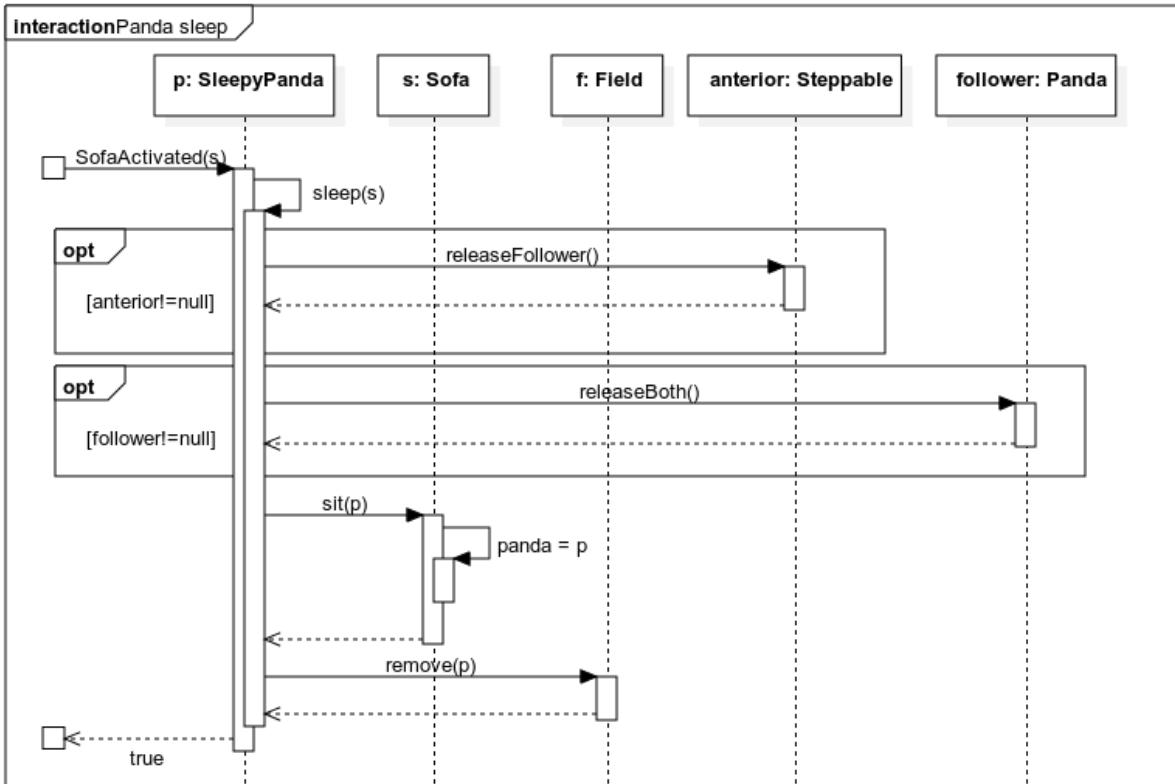
5.3.14 Panda jump



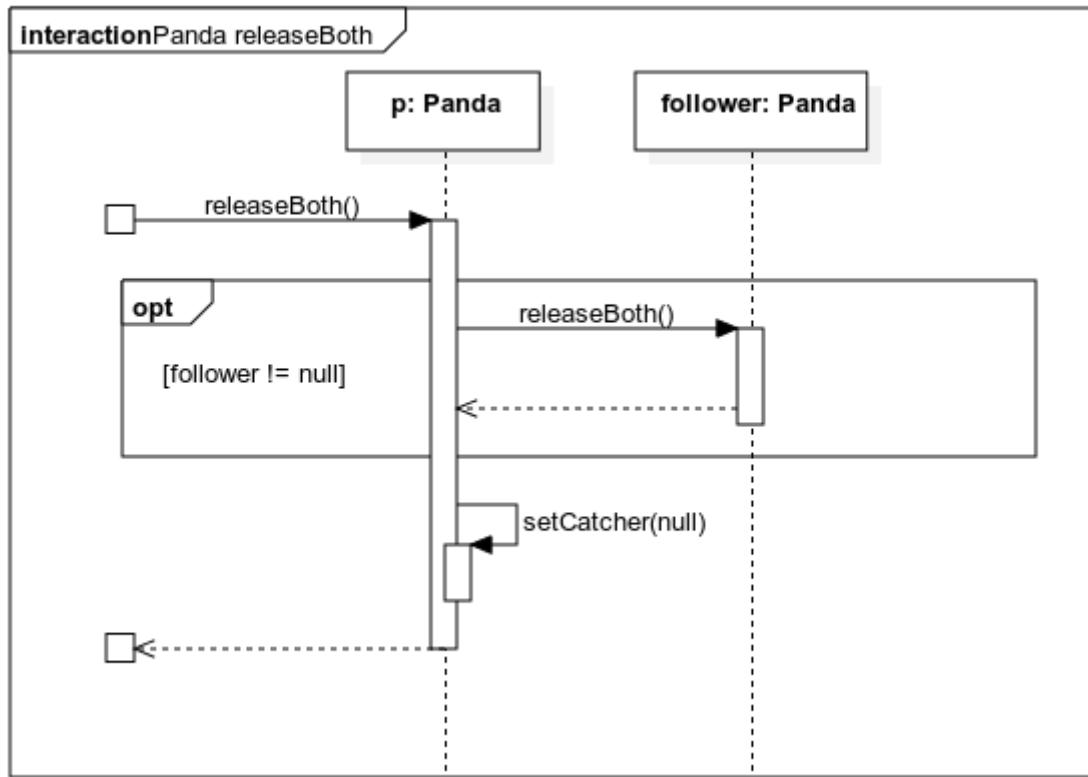
5.3.15 Panda scared



5.3.16 Panda sleep

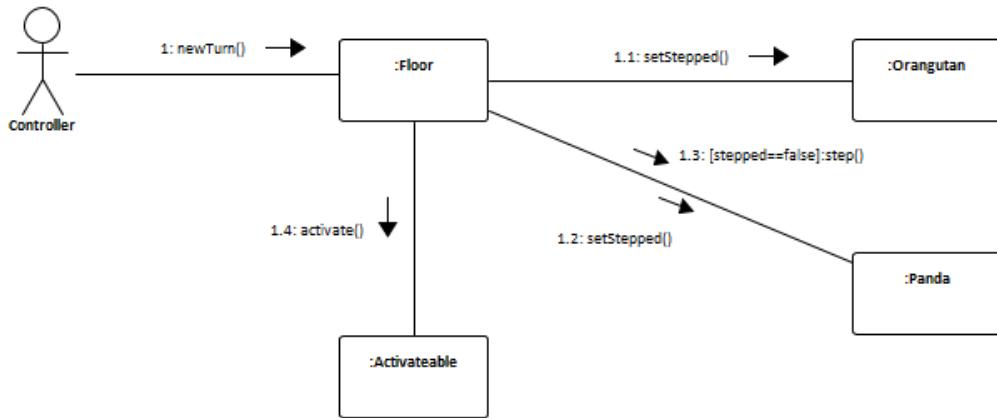


5.3.17 Panda releaseBoth

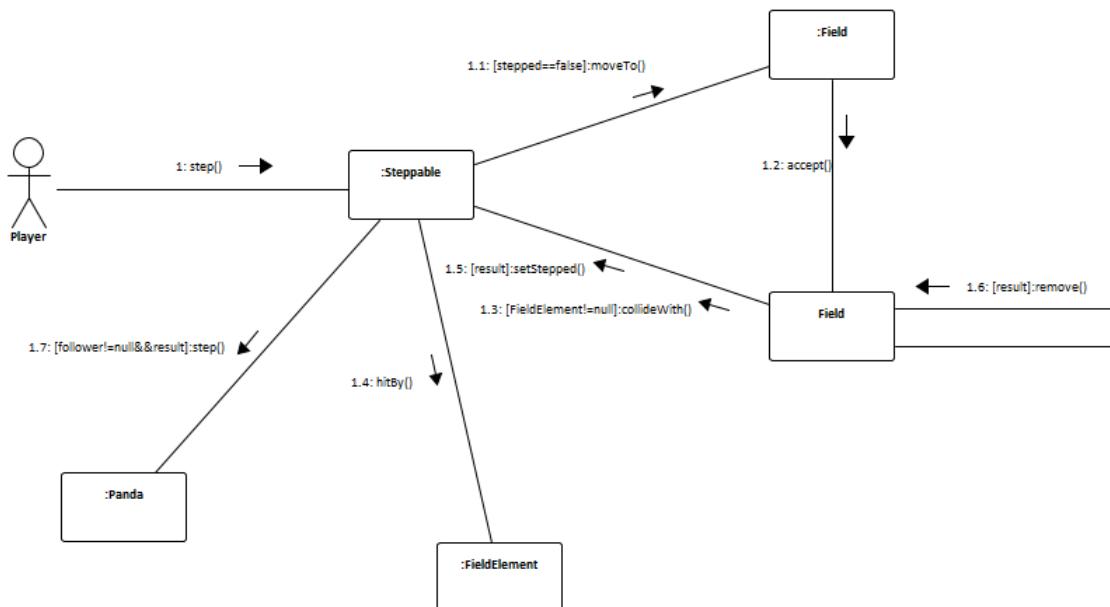


5.4 Kommunikációs diagramok

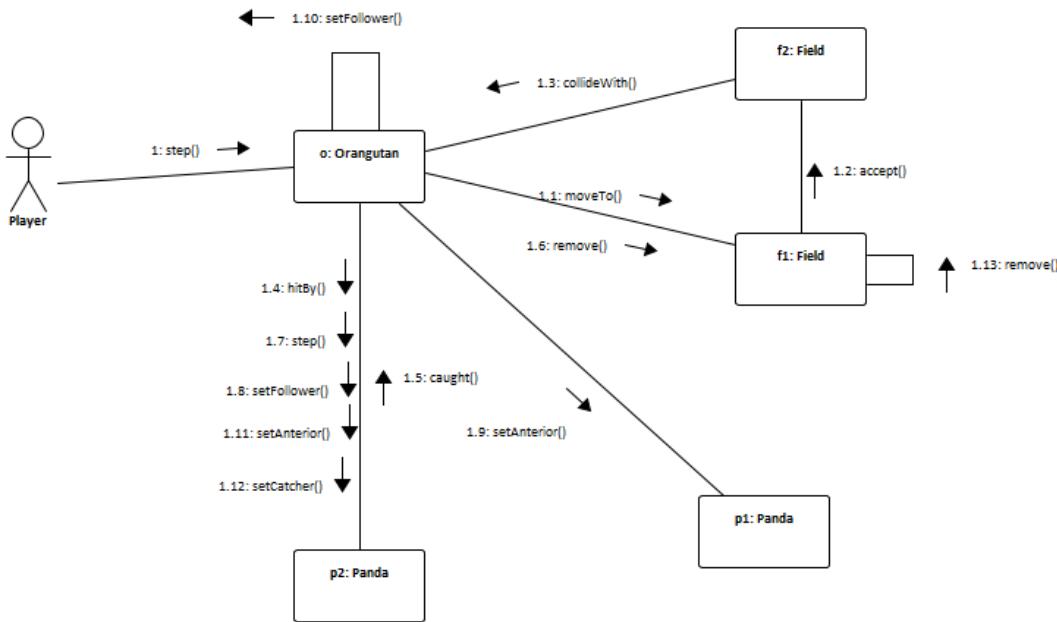
5.4.1 Turn start



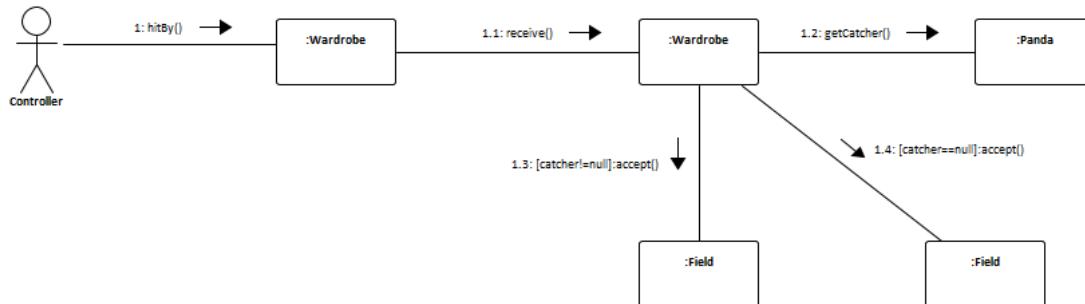
5.4.2 Steppable move



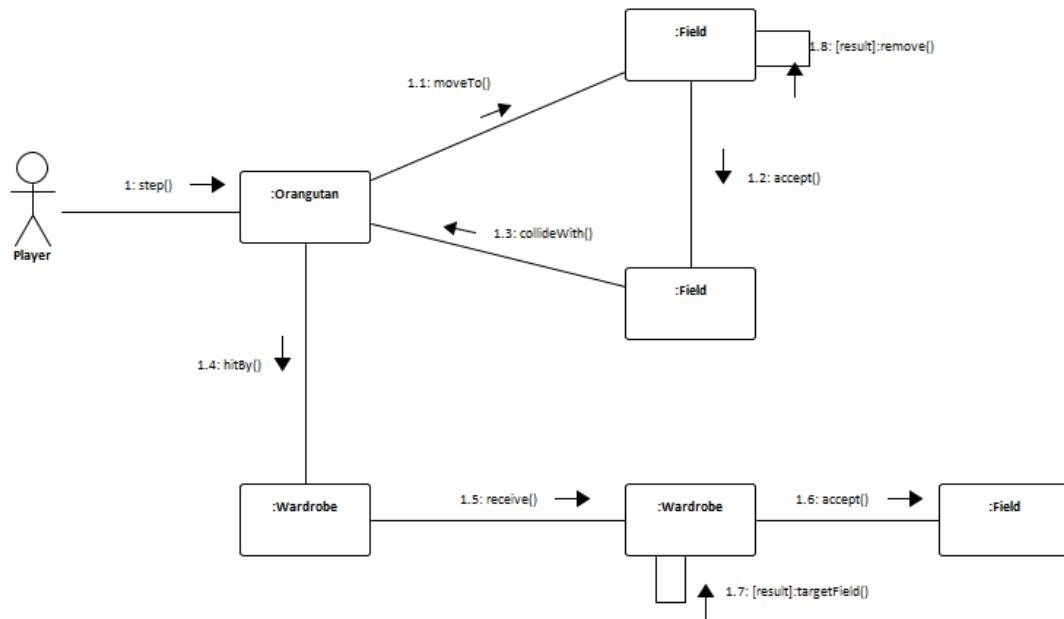
5.4.3 Orangutan catch panda



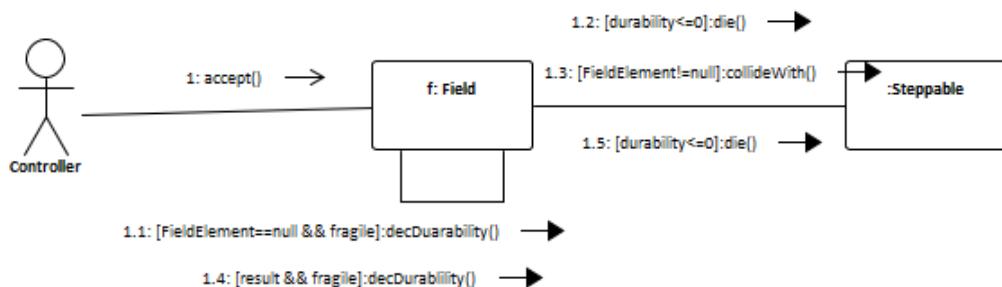
5.4.4 Panda goes to wardrobe



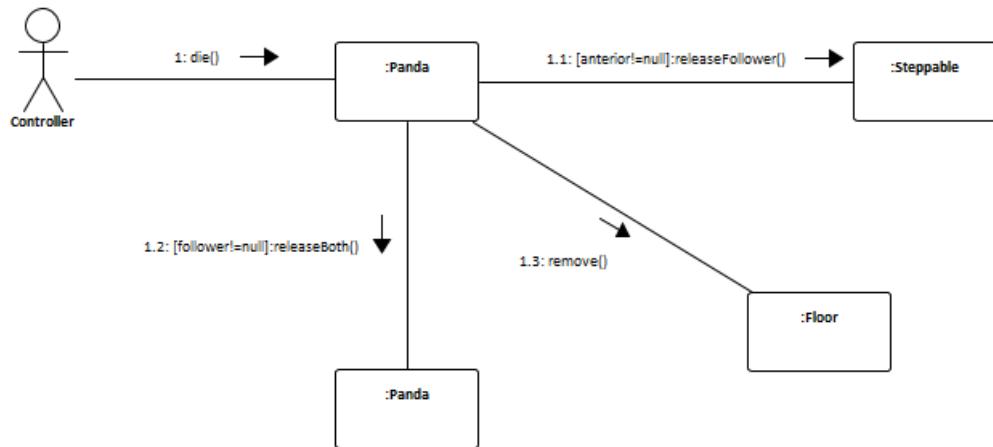
5.4.5 Orangutan goes to wardrobe



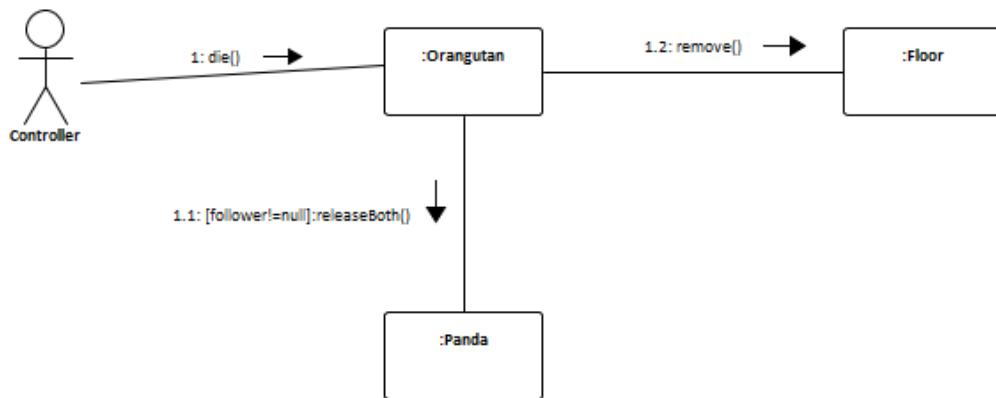
5.4.6 Check fragile



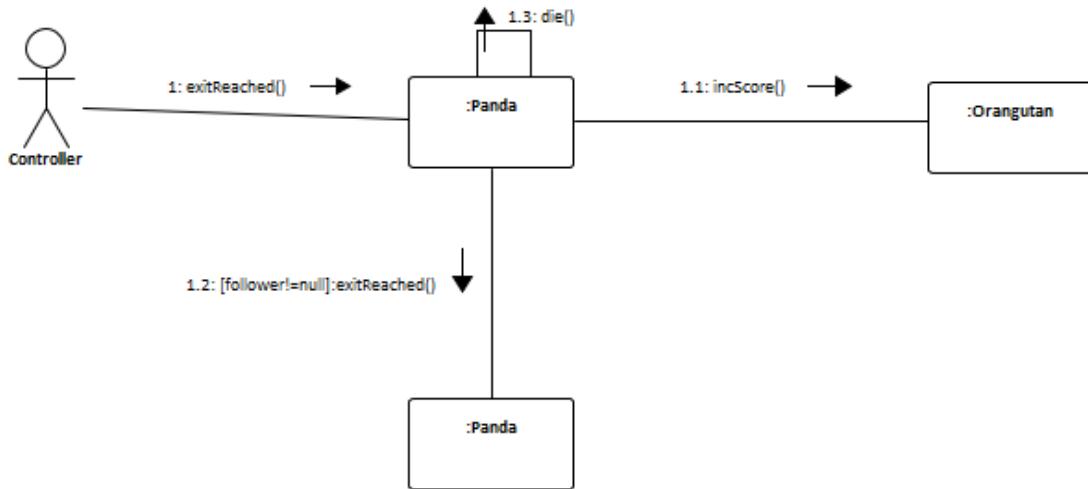
5.4.7 Panda die



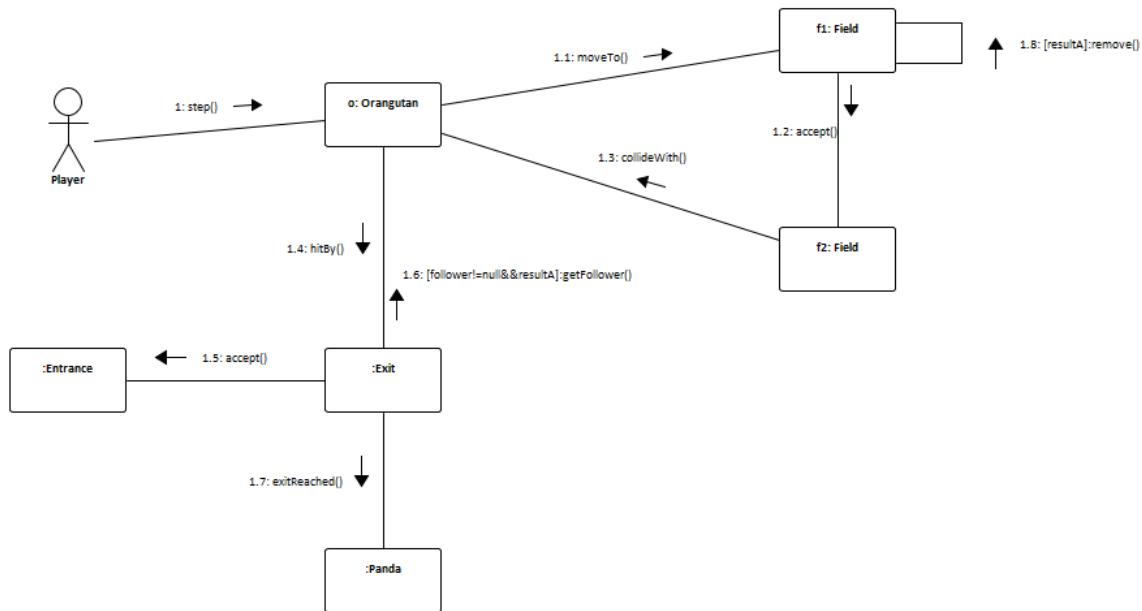
5.4.8 Orangutan die



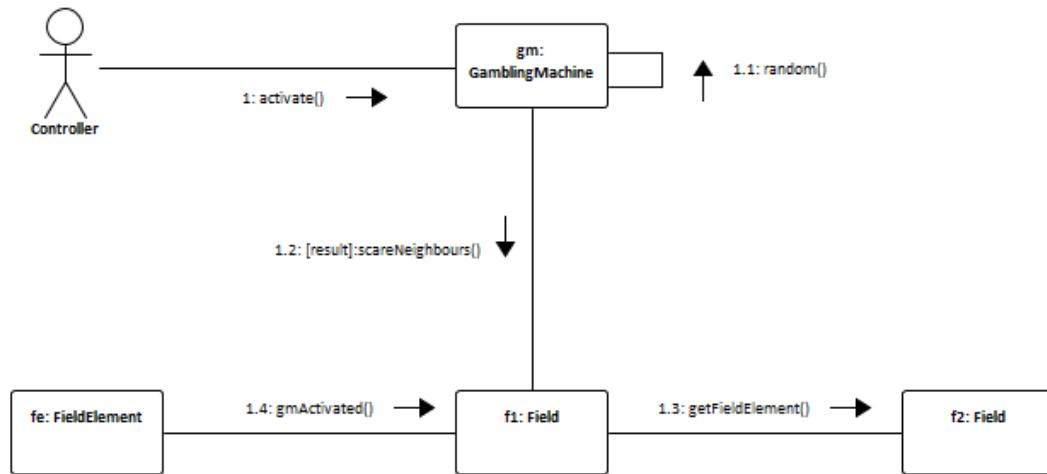
5.4.9 Panda exit



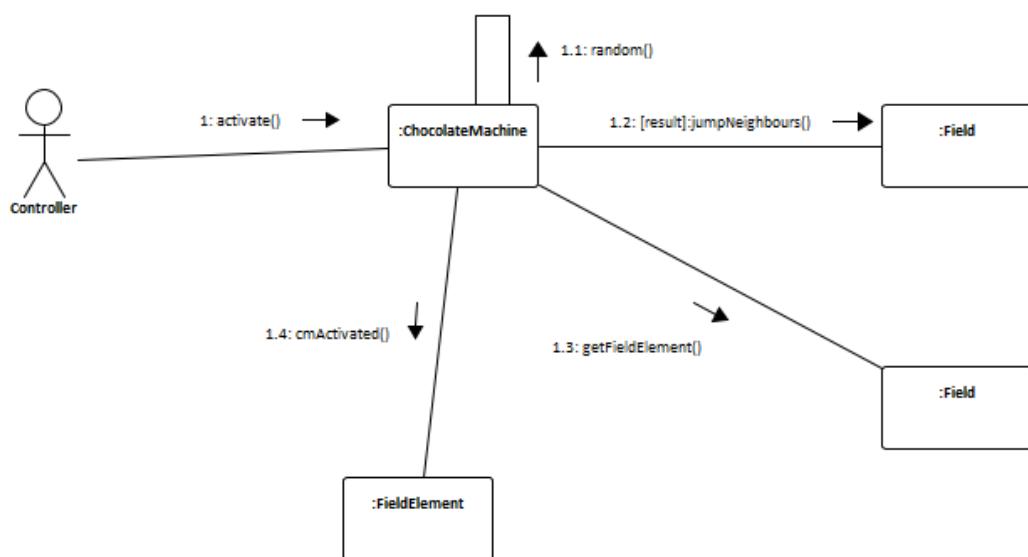
5.4.10 Orangutan exit



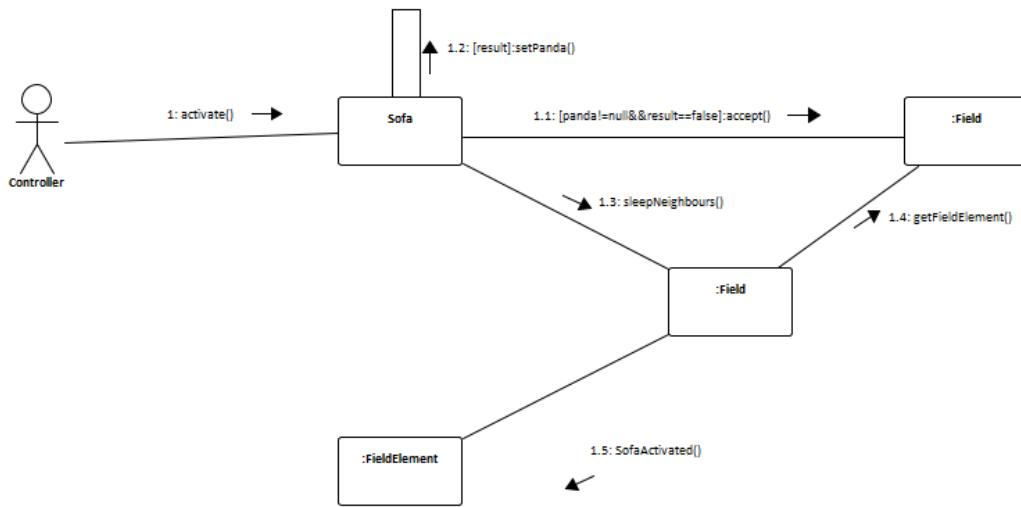
5.4.11 Activate gambling machine



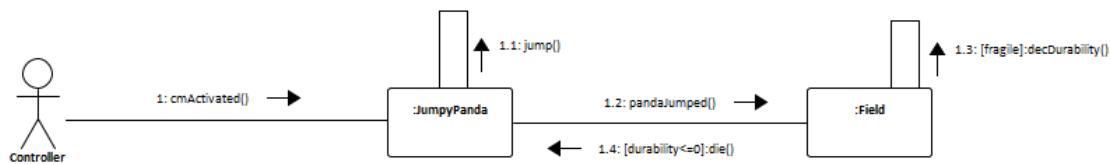
5.4.12 Activate chocolate machine



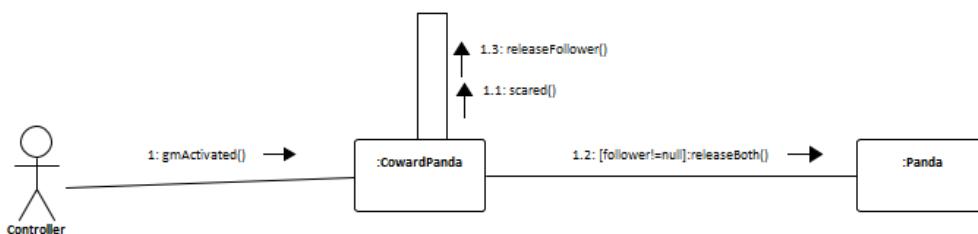
5.4.13 Activate sofa



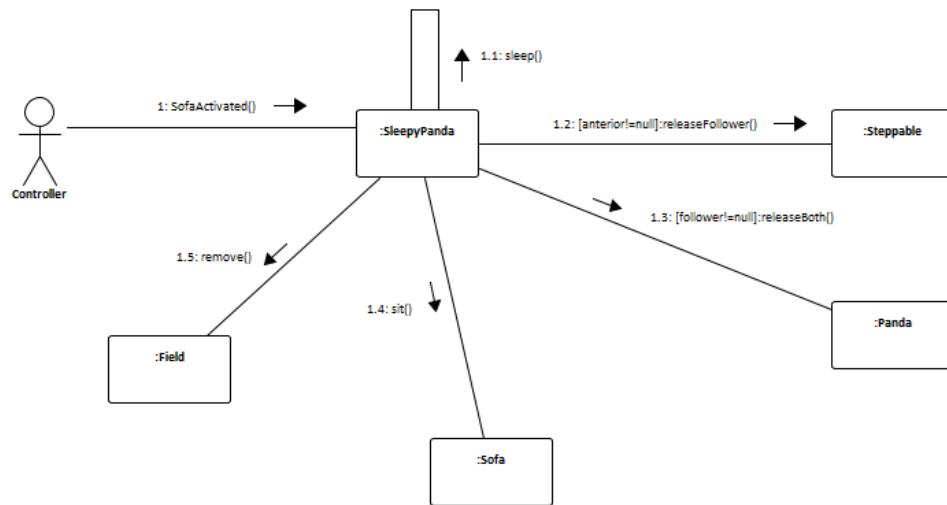
5.4.14 Panda jump



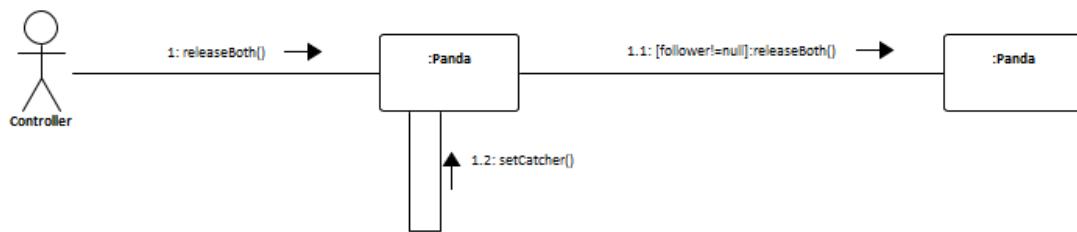
5.4.15 Panda scared



5.4.16 Panda sleep



5.4.17 Panda releaseBoth



5.5 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2019.03.05. 16:00	1 óra	Hegedüs	A szkeleton modell use-case diagramjának elkészítése.
2019.03.05. 19:00	1 óra	Hegedüs	A use-case diagram hibáinak javítása, kiegészítése, végleges formázása.
2019.03.06. 20:00	2 óra	Hegedüs	A use-case leírások elkészítése.
2019.03.07. 16:00	2 óra	Bódi	A feladathoz tartozó szekvenciák elkészítése.
2019.03.08. 22:00	1,5 óra	Borsodi	A kezelői felület tervezése.
2019.03.09. 21:00	2 óra	Bódi	Előző feladatban leadott hibás szekvenciák javítása, az elkészített szekvenciák beillesztése a dokumentumba.
2019.03.10. 19:00	4,5 óra	Nyári	Kommunikációs diagrammok elkészítése.

6.1 Szkeleton beadás

6.2 Fordítási és futtatási útmutató

6.2.1 Fájllista

6.2.1.1 A default package

Fájl neve	Méret	Keletkezés ideje	Tartalom
Main.java	7667 bájt	2019.03.21. 16:08	Főprogram menürendszerrel

6.2.1.2 A businesslogic package

Fájl neve	Méret	Keletkezés ideje	Tartalom
Activateable.java	772 bájt	2019.03.24. 23:35	Activateable osztály
ChocolateMachine.java	626 bájt	2019.03.24. 22:36	ChocolateMachine osztály
CowardPanda.java	856 bájt	2019.03.24. 22:36	CowardPanda osztály
Exit.java	1803 bájt	2019.03.24. 22:36	Exit osztály
Field.java	7183 bájt	2019.03.24. 22:36	Field osztály
FieldElement.java	2276 bájt	2019.03.24. 23:15	FieldElement osztály
Floor.java	3521 bájt	2019.03.24. 23:35	Floor osztály
GamblingMachine.java	724 bájt	2019.03.24. 22:56	GamblingMachine osztály
Game.java	1248 bájt	2019.03.24. 23:15	Game osztály
JumpyPanda.java	795 bájt	2019.03.24. 22:36	JumpyPanda osztály
Orangutan.java	3025 bájt	2019.03.24. 22:52	Orangutan osztály
Panda.java	4305 bájt	2019.03.24. 22:52	Panda osztály
SleepyPanda.java	1052 bájt	2019.03.24. 23:41	SleepyPanda osztály
Sofa.java	1731 bájt	2019.03.24. 22:59	Sofa osztály
Steppable.java	3192 bájt	2019.03.24. 22:41	Steppable osztály
Wardrobe.java	3434 bájt	2019.03.24. 22:38	Wardrobe osztály

6.2.1.3 A test package

Fájl neve	Méret	Keletkezés ideje	Tartalom
DepthWriter.java	1122 bájt	2019.03.24. 22:52	Tabulátorozott szöveget kiíró osztály
TestField.java	741 bájt	2019.03.23. 18:06	Mezőket tesztelő osztály
TestFloor.java	1110 bájt	2019.03.24. 23:35	A szintet tesztelő osztály
TestFurniture.java	2552 bájt	2019.03.23. 18:06	A berendezést tesztelő osztály
TestOrangutan.java	4685 bájt	2019.03.24. 22:36	Orángutánokat tesztelő osztály
TestPanda.java	5678 bájt	2019.03.24. 23:39	Pandákat tesztelő osztály

6.2.2 Fordítás

1. lépés: A letöltött tömörített állomány (zip fájl) kibontása egy tetszőleges helyre.
2. lépés: Windows parancssor megnyitása. (**Win+R** billentyű megnyomása, majd a **cmd** parancs beírása, végül az **ENTER** billentyű leütése.)
3. lépés: Navigálás a kibontás helyére a parancssorral. (A **cd <mappa_neve>** parancs használatával lehet a mappát váltani, a **cd..** parancs segítségével lehet a könyvtárhierarchiában egy szinttel feljebb lépni, a **dir** parancccsal pedig az aktuális helyen lévő fájlok és könyvtárak írathatók ki.)
4. lépés: A következő parancs futtatása: **javac -d bin Main.java businesslogic/*.java test/*.java**

6.2.3 Futtatás

1. lépés: Navigálás a kibontás helyére a parancssorral. (A **cd <mappa_neve>** parancs használatával lehet a mappát váltani, a **cd..** parancs segítségével lehet a könyvtárhierarchiában egy szinttel feljebb lépni, a **dir** parancccsal pedig az aktuális helyen lévő fájlok és könyvtárak írathatók ki.) A fordítás után az éppen aktuális hely ez lesz a parancssorban.
2. lépés: A következő parancs kiadása: **cd bin**
3. lépés: A következő parancs futtatása: **java Main**

6.3 Értékelés

Tag neve	Tag neptun	Munka százalékban
Hegedüs Amadé	P53ZLW	25,5%
Borsodi Regő	HH5GMS	25,5%
Bódi Dani	KM3B5S	24,5%
Nyári Tamás	X49AI8	24,5%

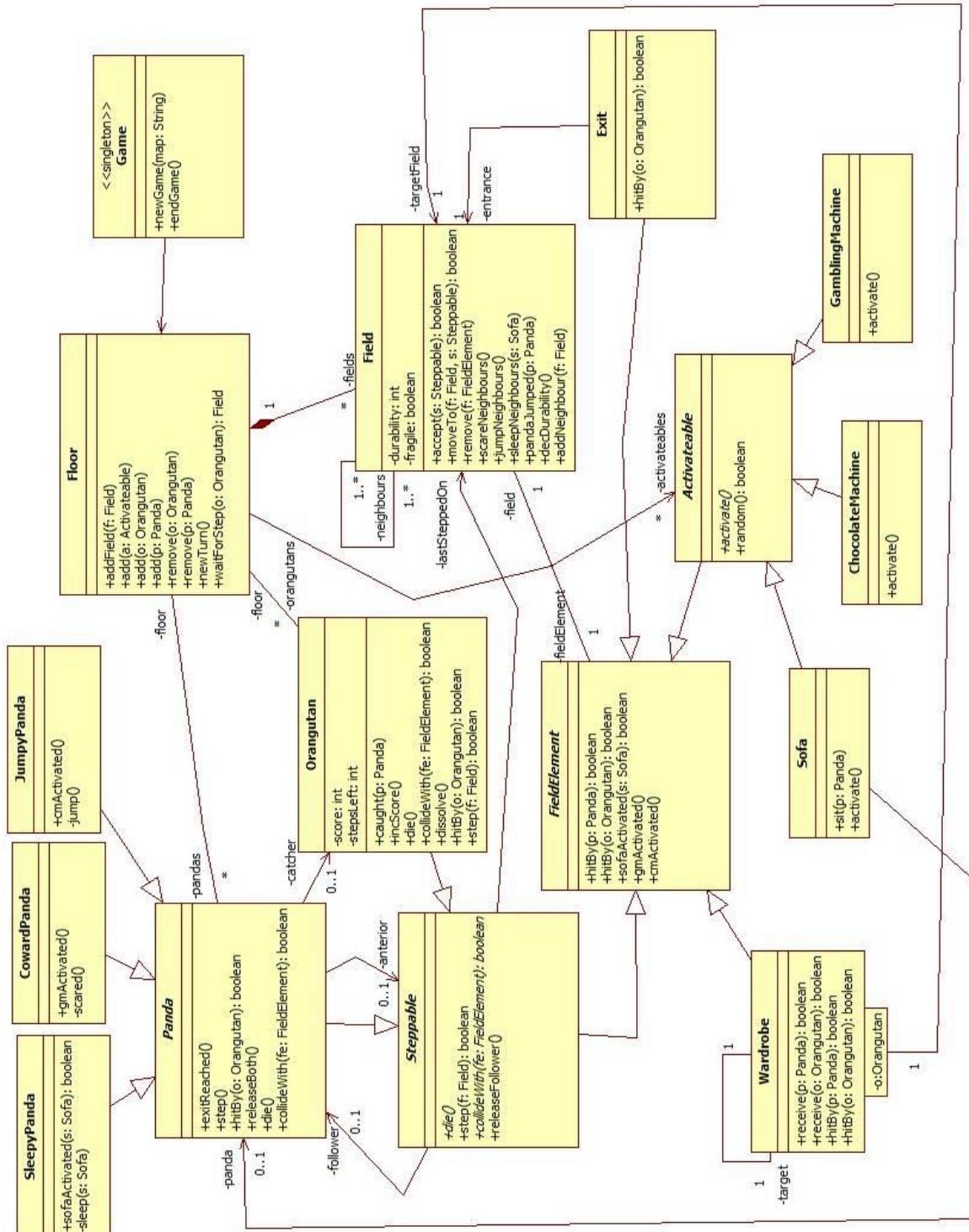
6.4 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2019.03.01. 16:00	1 óra	Hegedüs	Remote repository létrehozása Github-on, a projektfájl elkészítése, branchek létrehozása, munkát segítő pluginok keresése.
2019.03.01. 17:00	0,5 óra	Bódi	A verziókezelő rendszerrel felmerülő problémák megoldása.
2019.03.01. 19:00	3 óra	Hegedüs	Az osztálydiagram leképezése Java kódra. A metódusok minden össze a saját nevüköt írják ki.
2019.03.20. 9:00	2 óra	Borsodi	Menürendszer elkészítése.
2019.03.20. 11:00	5 óra	Bódi	Kiíró osztály a tesztesetekhez és az Orangutanhoz és a Fieldhez tartozó tesztesetek és a hozzájuk szükséges függvények megírása.
2019.03.21. 12:00	2,5 óra	Borsodi	Konzolkiírások, Furniture tesztek és hívott függvények.
2019.03.23. 19:00	1 óra	Bódi	Kommentelés és szintaktikai hibák javítása.
2019.03.24. 15:00	4 óra	Nyári	Pandákhöz tartozó tesztesetek és függvények megírása.
2019.03.24. 19:00	3 óra	Borsodi	Kommentelések és hibák javítása.
2019.03.24. 22:00	3 óra	Nyári	Kommentek befejezése és dokumentáció elkezdése.
2019.03.25. 01:00	1 óra	Hegedüs	Fordítási és futtatási útmutató elkészítése.
2019.03.25. 02:00	1 óra	Bódi	Dokumentum végső formázása.

7. Prototípus koncepciója

7.0 Változás hatása a modellre

7.0.1 Módosult osztálydiagram



7.0.2 Új vagy megváltozó metódusok

7.0.2.1 Orangutan

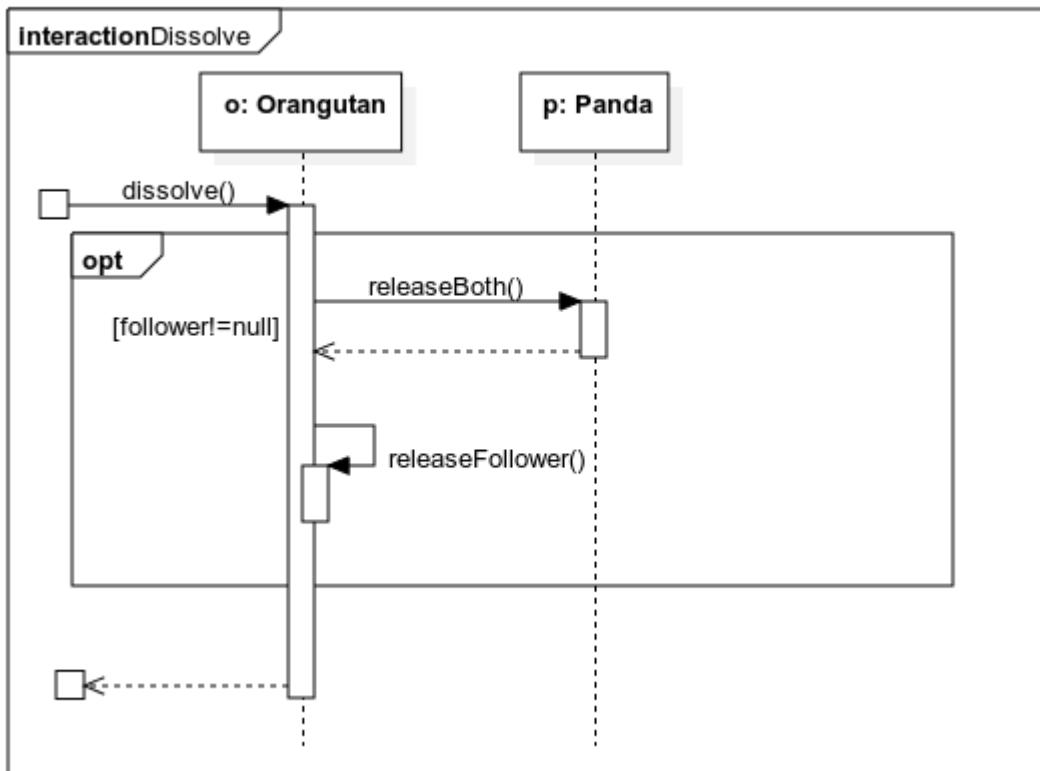
- **int stepsLeft:** Számolja, hogy hány lépés múlva foghat el pandát, vagy rabolhat el másik orangutántól.
- **void dissolve():** Az orangután elengedi a követőjének kezét, és a sor felbomlik.
- **boolean hitBy(o:Orangutan):** Ha a feltételek teljesülnek, a kapott orangután elrabolja a pandákat, és a két orangután helyet cserél. A feltételek: o-tól nem raboltak pandát 3 lépésen belül, és nem követi panda, a kirabolt orangutánt pedig követi legalább egy panda. Visszatér az ütközés sikerességével.
- **boolean step(f:Field):** Az orangután lépést kísérel meg a kapott mezőre, és visszatér a lépés sikerességével, ha sikeres a lépés a követőjét lépteti az előző mezőjére. Sikeres lépés esetén ha stepsLeft nem 0, akkor csökkenti eggyel.

7.0.2.2 Panda

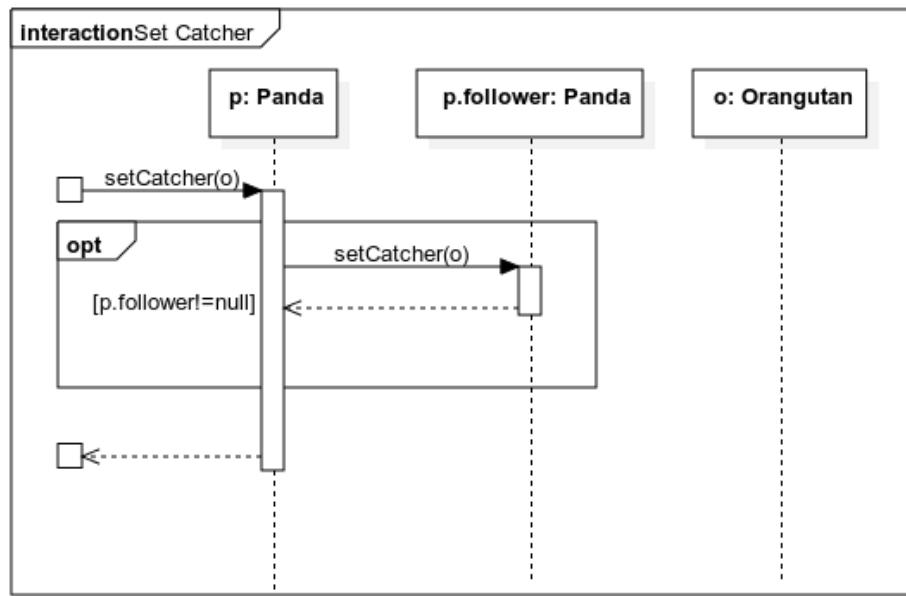
- **boolean hitBy(o:Orangutan):** A kapott orangutánnal történik ütközés, ha a panda szabad, és az orangutántól nem raboltak pandát 3 lépésen belül. Visszatér az ütközés kimenetelével.
- **void setCatcher(o:Orangutan):** Beállítja o-t catchernek, és ha a pandának van követője, meghívja rajta ezt a függvényt o-val.

7.0.3 Szekvencia-diagramok

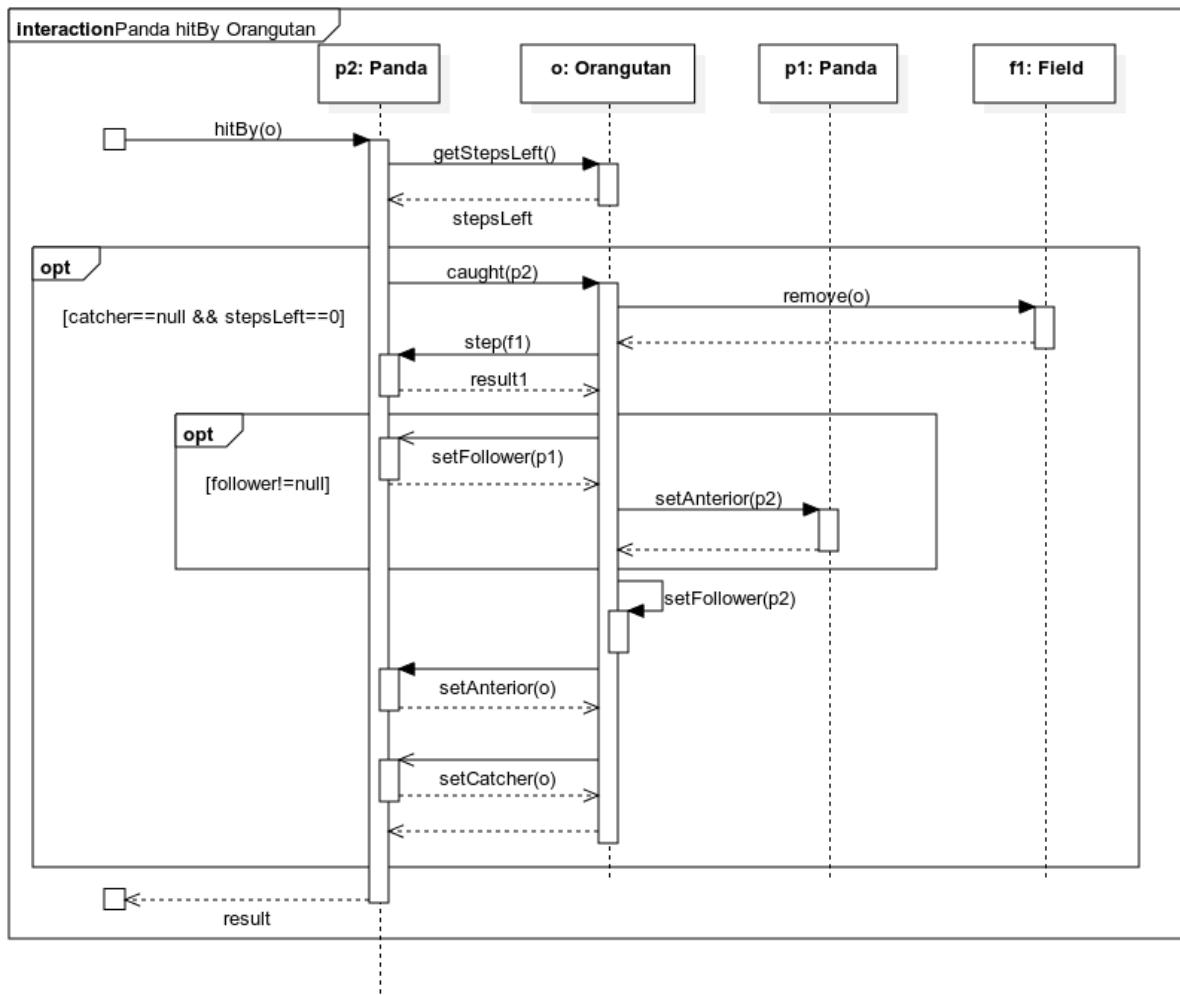
7.0.3.1 Dissolve



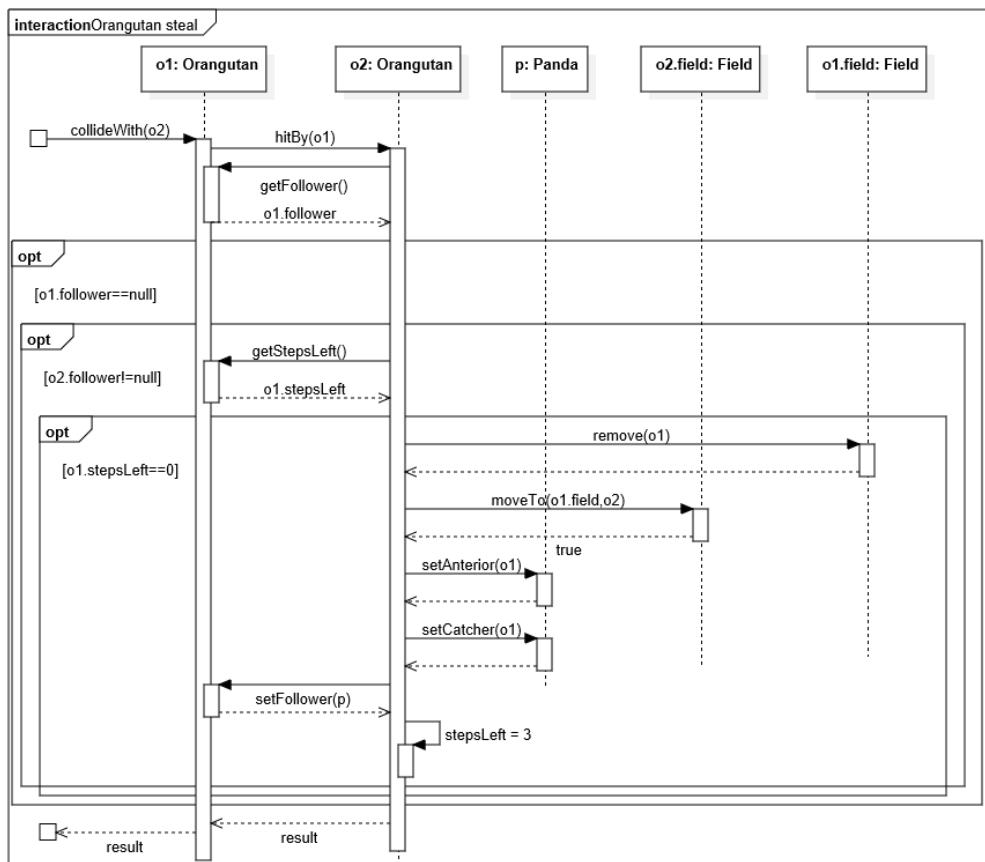
7.0.3.2 Set Catcher



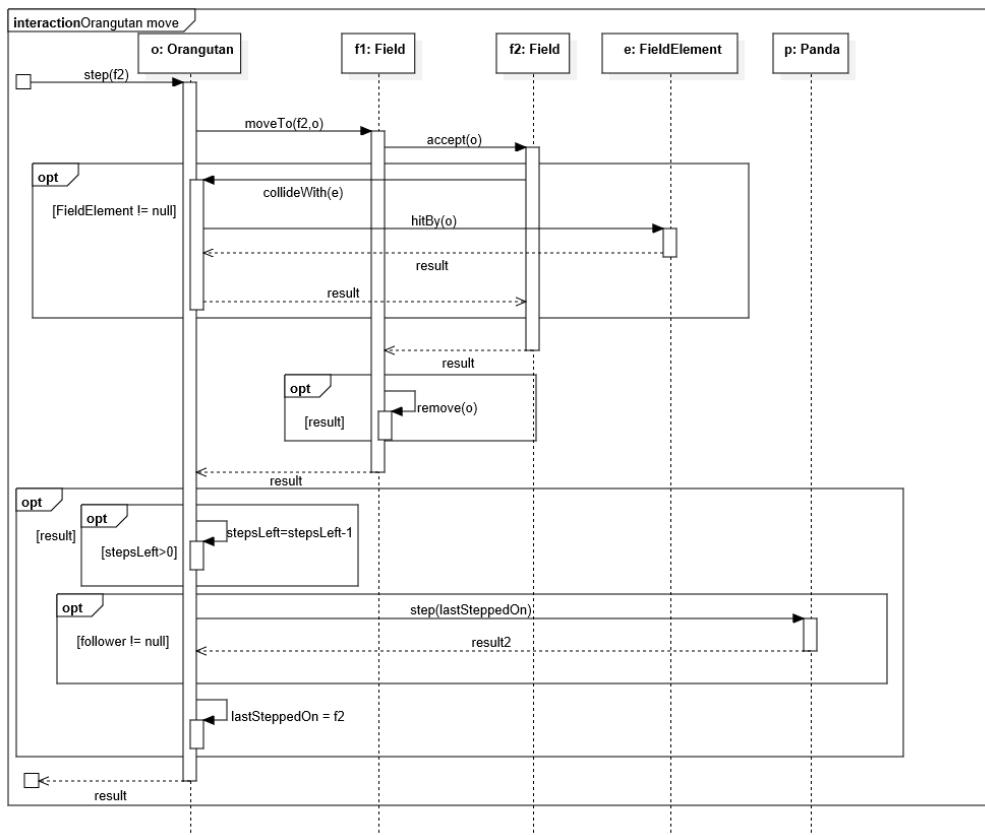
7.0.3.3 Panda hitBy Orangutan



7.0.3.4 Orangutan steal



7.0.3.5 Orangutan move



7.1 Prototípus interface-definíciója

7.1.1 Az interfész általános leírása

7.1.2 Bemeneti nyelv

A játékban a lépések előre meghatározott sorrendben követik egymást, így nem a játékos feladata észben tartani azt, hogy melyik orángutánnal lépett már és melyikkel nem. Az orángutánok körében a játék sorban megkérdezi a felhasználót, hogy az adott orángután melyik mezőre lépjén, a válasz pedig minden esetben egy ID, ami annak a mezőnek az azonosítója, melyre az orángutánt léptetni kívánja irányítója. A formális bemeneti nyelv ennek megfelelően csupán a játék indítását, leállítását definiálja, továbbá döntést segítő, listázó utasításokat tartalmaz azért, hogy ha a felhasználó nincs tisztában lehetőségeivel, akkor a listázó utasítások kimenetének tanulmányozásával megalapozott döntést tudjon hozni. A LIST* utasítások a játék indítása és a játék leállítása között használhatók.

7.1.2.1 Felhasználói utasítások

7.1.2.1.1 LOAD MAP -ID

Leírás: A pályaleíró nyelven definiált, tetszőleges, szintaktikailag helyes pályafájl betöltésére szolgáló utasítás.

Opciók:

- **ID:** A pályafájl azonosítója, amely egyben a pályafájl neve is. (A 14-es azonosítójú pályafájl tehát a 14.map fájlból kerül leírásra.) Érték megadása kötelező!

7.1.2.1.2 START

Leírás: Elindítja a játékot a betöltött pályafájl leírásának megfelelően.

7.1.2.1.3 STOP

Leírás: Leállítja a játékot.

7.1.2.1.4 LIST FIELDS

Leírás: Kilistázza a pálya összes mezőjét.

7.1.2.1.5 LIST NEIGHBOURS

Leírás: A lépni kívánó orángután lehetőségeit listázza ki a parancs. Ennek segítségével a játékos kilistáztathatja, hogy az orángután aktuális mezője mellett mely mezők helyezkednek el, amikre ráléphet, illetve, ha a mezők tartalmaznak objektumokat, akkor azokat is listázza mezőnként.

7.1.2.1.6 LIST PANDAS

Leírás: Listázza a játékban lévő, még élő pandákat.

7.1.2.1.7 LIST ORANGUTANS

Leírás: Listázza a játékban lévő, még élő orángutánokat.

7.1.2.1.8 LIST ACTIVATEABLES

Leírás: Listázza a játékban lévő csokiautomatákat, játékgépeket és foteleket.

7.1.2.1.9 LIST WARDROBES

Leírás: Listázza a játékban lévő szekrényeket, azok párájával együtt.

7.1.2.1.10 LIST EXITS

Leírás: Listázza a játékban lévő kilépési pontokat.

7.1.2.1.11 LIST CAUGHT PANDAS

Leírás: Listázza azokat az orángutánokat, akiknek van pandájuk, és listázza a mögötte lévő pandákat is sorban.

7.1.2.1.12 SAVE MAP -ID

Leírás: A pályaleíró nyelven definiált, tetszőleges, szintaktikailag helyes pályafájl mentésére szolgáló utasítás. A játékban lévő objektumok aktuális állapotának mentésére használható. Segítségével új pályakonfigurációs fájl hozható létre, illetve ellenőrizhető vele bizonyos utasítások lefuttatásának eredménye.

Opciók:

- **ID:** A pályafájl mentési azonosítója, amely egyben a pályafájl neve is lesz. (A 15-ös azonosítójú pályafájl tehát a 15.map fájlban kerül mentésre.) Érték megadása kötelező!

7.1.2.2 Bemeneti pályafájl

A játék inicializálása többszintű, rögzített szintakszisú, tagekkel ellátott szöveges dokumentumok alapján történik. Az említett dokumentumok egy-egy pályát reprezentálnak, így ez dokumentum pályafájként hivatkozik rájuk. A pályafájlok többek között a játékban előforduló objektumokat, azok speciális tulajdonságait és orientációjukat tartalmazzák. Az egyes szintek tabuláltak, ami segíti a fájl átláthatóságának javításában. Mindössze két (és egy nulladik) szint definiálására került sor a pályafájl tervezése során. Az első szint minden egy objektumot, a második szint pedig az adott objektum tulajdonságait, tartalmazza. A fájl felépítése kissé hasonlít a HTML dokumentumok felépítéséhez, hiszen a tagolás itt is tagekkel van megvalósítva, azonban a HTML-hez hasonlóan, ez sem egy érvényes XML dokumentum.

7.1.2.2.1 Alapszabályok

- minden értelmezendő adat nyitó- és zárótagok között helyezkedik el (pl. <name>Testmap</name>).
- minden nyitótaget kötelezően le kell zárni a megfelelő zárótaggal. Ezt nagyon egyszerű átlátni, hiszen a zárótag valójában egy nyitótag „/” karakterrel kiegészítve, ahogy az az előző pontban is látható.
- Kommenteket bárhol el lehet elhelyezni. A kommentek azonosítását a sor elején lévő „#” karakter segíti. (Azok a szövegrészletek, melyek nincsenek tagekkel körülvéve nem kerülnek értelmezésre, azonban az átláthatóság miatt ajánlott a kommentek előtt a „#” karaktert használni.)
- A pályafájl elejét a <mapfile>, végét pedig a </mapfile> tag jelzi (Ez az egyetlen nulladik szintű tag). Használata kötelező, és minden pályafájl csak egy ilyen nyitó- és egy zárótaget tartalmazhat.
- A pályaleírás (a map és azon belül az id és name tagek) megadása kötelező.
- Azonosító, (pl. <id>12345</id>) mint másodszintű tag, megadása minden esetben kötelező és egyedi.
- A szintek megkülönböztetésére tabulálást ajánlott alkalmazni.

7.1.2.2.2 Objektumok leírása

A leírás informális, hiszen lehetőségeket mutat be és nem szintenként csoportosítva kerülnek leírásra a tagek. Ez egyrészt a megértést segíti elő, másrészt pedig egyértelműen látható így, hogy az egyes elsőszintű tagokban, mely másodszintű tagok használhatók. A „#” karakterrel jelölt pályafájl kommentek a további megértést segítik.

7.1.2.2.2.1 Pálya leírása

```
<map>
    #pályaazonosító
    <id>1</id>
    #pálya neve
    <name>Test map</name>
</map>
```

7.1.2.2.2.2 Mező leírása

```
<field>
    #mezőazonosító
    <id>2</id>
    #mező egy x,y kontrollpontja
    <coord>0,0</coord>
    #mező egy x,y kontrollpontja
    <coord>10,0</coord>
    #mező egy x,y kontrollpontja
    <coord>10,10</coord>
    #törékenység (1: törékeny, 0: nem törékeny)
    <fragile>1</fragile>
    #élettartam
    <durability>14</durability>
    #egy szomszédos mező ID-ja
    <neighbour>3</neighbour>
</field>
```

7.1.2.2.2.3 Panda leírása

```
<panda>
    #objektumazonosító
    <id>4</id>
    #panda típusa (sleepy, coward, jumpy)
    <type>sleepy</type>
    #objektum gazdamezőjének ID-ja
    <host>2</host>
</panda>
```

7.1.2.2.2.4 Orángután leírása

```
<orangutan>
    #objektumazonosító
    <id>4</id>
    #objektum gazdamezőjének ID-ja
    <host>2</host>
</orangutan>
```

7.1.2.2.5 Activateable leírása

```
<activateable>
    #objektumazonosító
    <id>4</id>
    #activateable típusa (gamblingmachine, chocolatemachine, sofa)
    <type>sofa</type>
    #objektum gazdamezőjének ID-ja
    <host>2</host>
</activateable>
```

7.1.2.2.6 Szekrény leírása

```
<wardrobe>
    #objektumazonosító
    <id>4</id>
    #a wardrobe párjának ID-ja
    <target>12</target>
    #a wardrobe célmezőjének ID-ja
    <targetField>5</targetField>
    #objektum gazdamezőjének ID-ja
    <host>2</host>
</wardrobe>
```

7.1.2.2.7 Kilépési pont leírása

```
<exit>
    #objektumazonosító
    <id>4</id>
    #objektum gazdamezőjének ID-ja
    <host>2</host>
</exit>
```

7.1.3 Kimeneti nyelv

A kimeneti nyelv a felhasználó utasítására adott válaszokat, a játékmenet alakítására szolgáló, inputbekérő üzeneteket, továbbá a játékmenet megértését, átlátását segítő üzeneteket (MESSAGE-eket) definiálja. Az utóbbiak jelzik a játékos felé, ha egy panda beleült egy fotelbe, vagy ha egy orángután nem tud a megadott mezőre lépni stb.

7.1.3.1 Felhasználói utasításokra adott lehetséges válaszok

7.1.3.1.1 *LOAD MAP -ID* utasításra adott lehetséges válaszok

7.1.3.1.1.1 MESSAGE: Map loading successful!

Leírás: A pálya betöltése sikeres.

7.1.3.1.1.2 MESSAGE: Map loading failed!

Leírás: A pálya betöltése sikertelen. Oka: a pályafájl nem létezik vagy szintaktikailag helytelen.

7.1.3.1.2 *START* utasításra adott lehetséges válaszok

7.1.3.1.2.1 MESSAGE: Game started.

Leírás: A játék sikeresen elindult.

7.1.3.1.2.2 MESSAGE: Game could not start.

Leírás: A játék indítása sikertelen volt. Oka: Nem került betöltésre még szintaktikailag helyes pályafájl.

7.1.3.1.3 STOP utasításra adott lehetséges válaszok

7.1.3.1.3.1 MESSAGE: Game stopped.

Leírás: A játék sikeresen leállt.

7.1.3.1.4 LIST FIELDS utasításra adott lehetséges válaszok

7.1.3.1.4.1 Field <ID>, object ID: <object ID>, fragile:<fragile>, durability: <durability>

Leírás: A játék kilistázza az összes mezőt a fentebb leírt formában több sor felhasználásával. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- **<ID>:** Mezőazonosító.
- **<object ID>:** A mezőn lévő objektum azonosítója. (Ha nincs rajta objektum, akkor NULL.)
- **<fragile>:** A mező törékenységét jelző flag értéke (1: törékeny, 0: nem törékeny).
- **<durability>:** Akkor értelmezett, ha a mező törékeny. Ilyenkor azt az értéket tartalmazza ahányszor még ráléphetnek az adott mezőre leszakadása előtt.

7.1.3.1.4.2 MESSAGE: Game cannot list.

Leírás: A mezők nem listázhatók ki. Oka: A játék nincs elindítva.

7.1.3.1.5 LIST NEIGHBOURS utasításra adott lehetséges válaszok

7.1.3.1.5.1 Field <ID>, object ID: <object ID>, fragile:<fragile>, durability: <durability>

Leírás: A játék kilistázza az összes szomszédos mezőt a fentebb leírt formában több sor felhasználásával. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- **<ID>:** Mezőazonosító.
- **<object ID>:** A mezőn lévő objektum azonosítója. (Ha nincs rajta objektum, akkor NULL.)
- **<fragile>:** A mező törékenységét jelző flag értéke (1: törékeny, 0: nem törékeny).
- **<durability>:** Akkor értelmezett, ha a mező törékeny. Ilyenkor azt az értéket tartalmazza ahányszor még ráléphetnek az adott mezőre leszakadása előtt.

7.1.3.1.5.2 MESSAGE: Game cannot list.

Leírás: A mezők nem listázhatók ki. Oka: A játék nincs elindítva.

7.1.3.1.6 LIST PANDAS utasításra adott lehetséges válaszok

7.1.3.1.6.1 Panda <ID>, type: <type>, host ID: <host ID>

Leírás: A játék kilistázza az összes, még élő pandát a fentebb leírt formában több sor felhasználásával. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- **<ID>:** Panda azonosítója.
- **<type>:** A panda típusa (sleepy, coward, jumpy).
- **<host ID>:** A panda gazdamezőjének (amin éppen áll) azonosítója.

7.1.3.1.6.2 MESSAGE: Game cannot list.

Leírás: A pandák nem listázhatók ki. Oka: A játék nincs elindítva.

7.1.3.1.7 LIST ORANGUTANS utasításra adott lehetséges válaszok**7.1.3.1.7.1 Orangutan <ID>, host ID: <host ID>**

Leírás: A játék kilistázza az összes, még élő orangutánt a fentebb leírt formában több sor felhasználásával. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <ID>: Orangután azonosítója.
- <host ID>: Az orangután gazdamezőjének (amin éppen áll) azonosítója.

7.1.3.1.7.2 MESSAGE: Game cannot list.

Leírás: Az orangutánok nem listázhatók ki. Oka: A játék nincs elindítva.

7.1.3.1.8 LIST ACTIVATEABLES utasításra adott lehetséges válaszok**7.1.3.1.8.1 Activateable <ID>, type: <type>, host ID: <host ID>**

Leírás: A játék kilistázza az összes, activateable-t a fentebb leírt formában több sor felhasználásával. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <ID>: Activateable azonosítója
- <type>: Az activateable típusa (gamblingmachine, chocolatemachine, sofa).
- <host ID>: Az activateable gazdamezőjének azonosítója.

7.1.3.1.8.2 MESSAGE: Game cannot list.

Leírás: Az activateable-ök nem listázhatók ki. Oka: A játék nincs elindítva.

7.1.3.1.9 LIST WARDROBES utasításra adott lehetséges válaszok**7.1.3.1.9.1 Wardrobe <ID>, target ID: <target ID>, target field ID: <target field ID>, host ID: <host ID>**

Leírás: A játék kilistázza az összes, szekrényt a fentebb leírt formában több sor felhasználásával. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <ID>: Szekrény azonosítója.
- <target ID>: A szekrény párjának azonosítója.
- <target field ID>: A szekrény célmezőjének azonosítója. (Ha egy panda vagy orangután kilép a szekrényből, akkor erre a mezőre kerül.)
- <host ID>: A szekrény gazdamezőjének azonosítója.

7.1.3.1.9.2 MESSAGE: Game cannot list.

Leírás: A szekrények nem listázhatók ki. Oka: A játék nincs elindítva.

7.1.3.1.10 LIST EXITS utasításra adott lehetséges válaszok**7.1.3.1.10.1 Exit <ID>, host ID: <host ID>**

Leírás: A játék kilistázza az összes kilépési pontot a fentebb leírt formában több sor felhasználásával. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <ID>: Kilépési pont azonosítója.
- <host ID>: A kilépési pont gazdamezőjének azonosítója.

7.1.3.1.10.2 MESSAGE: Game cannot list.

Leírás: A kilépési pontok nem listázhatók ki. Oka: A játék nincs elindítva.

7.1.3.1.11 LIST CAUGHT PANDAS utasításra adott lehetséges válaszok**7.1.3.1.11.1 Orangutan <orangutan ID>-Panda <panda(1) ID>-Panda <panda(2) ID>-.... Panda <panda(n) ID>**

Leírás: A játék kilistázza az összes pandával rendelkező orángutánt és az általuk elkapott pandákat a fentebb leírt formában több sor felhasználásával. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <panda ID>: Az elkapó orángután azonosítója
- <panda(i) ID>: Az elkapott pandák azonosítója. (i = 1 ... {pandák száma})

7.1.3.1.11.2 MESSAGE: Game cannot list.

Leírás: Az elkapók és elkapottjaik nem listázhatók ki. Oka: A játék nincs elindítva.

7.1.3.1.12 SAVE MAP -ID utasításra adott lehetséges válaszok**7.1.3.1.12.1 MESSAGE: Map saving successful!**

Leírás: A pálya (és állapot) mentése sikeres.

7.1.3.1.12.2 MESSAGE: Map saving failed!

Leírás: A pálya mentése sikertelen. Oka: a játék nincs leállítva.

7.1.3.2 A játékmenet alakítására szolgáló, inputbekérő üzenetek

Az inputbekérő üzenetek sajátossága, hogy tartalmazzák azt zárójelek között, hogy a választ milyen formában várják, így a lehetséges válaszok szintaktikáját nem kell megtanulnia a játékosnak.

7.1.3.2.1 MESSAGE: Orangutan <ID> has to step. Give a field ID! (ID)

Leírás: Az adott orángutánnak lépnie kell. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <ID>: Annak az orángutánnak az azonosítója, melynek lépnie kell.

7.1.3.2.2 MESSAGE: Does Orangutan <ID> set free his pandas? (Y/N)

Leírás: Az orángután szabadon engedje eddig elkapott pandáit?

- <ID>: Annak az orángutánnak az azonosítója, mely soron van épp.

7.1.3.3 A játékmenet megértését, átlátását segítő üzenetek**7.1.3.3.1 MESSAGE: Round of orangutans.**

Leírás: Az orángutánok köre elkezdődött.

7.1.3.3.2 MESSAGE: Round of pandas.

Leírás: A pandák köre elkezdődött.

7.1.3.3.3 MESSAGE: Panda <panda ID> stepped to Field <field ID>.

Leírás: A panda rálépett egy mezőre. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <panda ID>: A lépést végrehajtó panda azonosítója.
- <field ID>: Annak a mezőnek az azonosítója, melyre a panda sikeresen átlépett.

7.1.3.3.4 MESSAGE: Orangutan <orangutan ID> stepped to Field <field ID>.

Leírás: A panda rálépett egy mezőre. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <orangutan ID>: A lépést végrehajtó orángután azonosítója.
- <field ID>: Annak a mezőnek az azonosítója, melyre az orángután sikeresen átlépett.

7.1.3.3.5 MESSAGE: Orangutan <orangutan ID> could not step to Field <field ID>.

Leírás: Az orángután nem tudott rálépn az adott mezőre. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <orangutan ID>: A lépési kísérletet végrehajtó orángután azonosítója.
- <field ID>: Annak a mezőnek az azonosítója, melyre az orángután át kívánt volna lépni.

7.1.3.3.6 MESSAGE: Panda <panda ID> jumped on Field <field ID>.

Leírás: A panda ugrott. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <panda ID>: Az ugrást végrehajtó panda azonosítója.
- <field ID>: Annak a mezőnek az azonosítója, melyen a panda ugrott.

7.1.3.3.7 MESSAGE: Panda <panda ID> got scared on Field <field ID>.

Leírás: A panda megijedt. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <panda ID>: A megijedő panda azonosítója.
- <field ID>: Annak a mezőnek az azonosítója, melyen a panda megijedt.

7.1.3.3.8 MESSAGE: Panda <panda ID> sat down on Field <field ID>.

Leírás: A panda leült egy fotelbe. Az egyes tagekben lévő tulajdonságok jelentése a következő:

- <panda ID>: A leülést végrehajtó panda azonosítója.
- <field ID>: Annak a mezőnek az azonosítója, melyen a panda leült.

7.1.3.3.9 MESSAGE: Durability decrementation of Field <field ID>

Leírás: Az adott mezőn lévő csempe élettartama csökken, mert valami rálépett, vagy egy panda ugrott rajta.

- <field ID>: Az élettartamcsökkenést elszennvedő mező azonosítója.

7.1.3.3.10 MESSAGE: Field <field ID> has broken.

Leírás: Az adott mezőn lévő csempe összetört.

- <field ID>: A törést elszennvedő mező azonosítója.

7.1.3.3.11 MESSAGE: Panda <panda ID> died.

Leírás: Az adott panda meghalt. Vagy valamelyik orángután kivitte a pályáról, vagy lezuhant.

- <panda ID>: A meghalt panda azonosítója.

7.1.3.3.12 MESSAGE: Orangutan <orangutan ID> died.

Leírás: Az adott orángután meghalt, mert lezuhant.

- <orangutan ID>: A meghalt orángután azonosítója.

7.1.3.3.13 MESSAGE: Panda <panda ID> stepped into Wardrobe <wardrobe ID>.

Leírás: Az adott panda belépett egy szekrénybe.

- <panda ID>: A szekrénybe belépett panda azonosítója.
- <wardrobe ID>: Annak a szekrénynek az azonosítója, melybe a panda belépett.

7.1.3.3.14 MESSAGE: Orangutan <orangutan ID> stepped into Wardrobe <wardrobe ID>.

Leírás: Az adott orángután belépett egy szekrénybe.

- <orangutan ID>: A szekrénybe belépett orángután azonosítója.
- <wardrobe ID>: Annak a szekrénynek az azonosítója, melybe az orángután belépett.

7.1.3.3.15 MESSAGE: Panda <panda ID> stepped out from Wardrobe <wardrobe ID>.

Leírás: Az adott panda kilépett egy szekrényből.

- <panda ID>: A szekrényből kiléző panda azonosítója.
- <wardrobe ID>: Annak a szekrénynek az azonosítója, melyből a panda kilépett.

7.1.3.3.16 MESSAGE: Orangutan <orangutan ID> stepped out from Wardrobe <wardrobe ID>.

Leírás: Az adott orángután kilépett egy szekrényből.

- <orangutan ID>: A szekrényből kilépett orángután azonosítója.
- <wardrobe ID>: Annak a szekrénynek az azonosítója, melyből az orángután kilépett.

7.1.3.3.17 MESSAGE: Panda <panda ID> exited.

Leírás: Az adott panda kilépett a plázából egy kilépési ponton.

- <panda ID>: A kilépett panda azonosítója.

7.1.3.3.18 MESSAGE: Orangutan <orangutan ID> exited.

Leírás: Az adott orángután meghalt, mert lezuhant.

- <panda ID>: A kilépett orángután azonosítója.

7.1.3.3.19 MESSAGE: Activateable <ID> (<type>) was activated.

Leírás: Egy activateable aktiválódott.

- <ID>: Az activateable azonosítója.
- <type>: Az activateable típusa (chocolatemachine, gamblingmachine, sofa).

7.1.3.3.20 MESSAGE: Orangutan <orangutan ID> caught Panda <panda ID>.

Leírás: Egy orangután elkapott egy pandát.

- <orangutan ID>: Az elkapó orangután azonosítója.
- <panda ID>: Az elkapott panda azonosítója.

7.1.3.3.21 MESSAGE: Orangutan <orangutan ID> caught Panda <panda ID>.

Leírás: Egy orangután elvesztett egy pandát.

- <orangutan ID>: A pandát vesztő orangután azonosítója.
- <panda ID>: A megmenekült panda azonosítója.

7.1.3.3.22 MESSAGE: Orangutan <orangutan ID> stole Pandas from Orangutan <orangutanID2>.

Leírás: Egy orangután ellopta egy mási orangután pandáit.

- <orangutan ID>: A pandákat ellopó orangután azonosítója.
- <orangutan ID2>: A pandákat elvesztő orangután azonosítója.
- <panda ID>: A megmenekült panda azonosítója.

7.2 Összes részletes use-case

Use-case neve	loadMap
Rövid leírás	Betölti a játék térképét
Aktorok	Tesztelő
Forgatókönyv	A program megnyitja az előre elmentett térképet. A megadott paraméternek megfelelő térképet tölti be.

Use-case neve	start
Rövid leírás	Elindítja a játékot
Aktorok	Tesztelő
Forgatókönyv	A program a betöltött térképen elindítja a játékot.

Use-case neve	stop
Rövid leírás	Leállítja a játékot
Aktorok	Tesztelő
Forgatókönyv	A program leállítja a játékot.

Use-case neve	listFields
Rövid leírás	Kilistázza az összes mezőt
Aktorok	Tesztelő
Forgatókönyv	A megadott térképen az összes mezőt kilistázza.

Use-case neve	listNeighbours
Rövid leírás	Egy orángután szomszédjait listázza ki
Aktorok	Tesztelő
Forgatókönyv	Amelyik pandával éppen lépünk, kilistázza a szomszágjait, hogy tudjuk, hova tud lépni. A rajta levő objektumokat is kilistázza.

Use-case neve	listPandas
Rövid leírás	Kilistázza a pandákat
Aktorok	Tesztelő
Forgatókönyv	Az összes élő pandát kilistázza a pályáról.

Use-case neve	listOrangutans
Rövid leírás	Kilistázza az orángutánokat
Aktorok	Tesztelő
Forgatókönyv	Kilistázza az összes élő orángutánt.

Use-case neve	listActivateables
Rövid leírás	Kilistázza az objektumokat
Aktorok	Tesztelő
Forgatókönyv	Az összes pályán lévő csokiautomatát, játékgépet és fotelt kilistázza.

Use-case neve	listWardrobes
Rövid leírás	Kilistázza a szekrényeket
Aktorok	Tesztelő
Forgatókönyv	A pályán levő szekrényeket a párukkal együtt kilistázza.

Use-case neve	listExits
Rövid leírás	Kilistázza a kilépési pontokat
Aktorok	Tesztelő
Forgatókönyv	Minden kilépési pontot kilistáz.

Use-case neve	listCaughtPandas
Rövid leírás	Kilistázza az elkapott pandákat és elkapódikat
Aktorok	Tesztelő
Forgatókönyv	Az összes orángutánt, ami kapott el pandát, kilistázza a mögöttük lévő sorral együtt.

Use-case neve	saveMap
Rövid leírás	Elmenti a pályát az objektumokkal együtt
Aktorok	Tesztelő
Forgatókönyv	Egy szabályos pályakonfigurációt ment el a pályáról és az objektumokról, ezért később be lehet tölteni.

7.3 Tesztelési terv

Teszt-eset neve	1.1 Orangutan step
Rövid leírás	Orangutan lépésének a tesztelése.
Teszt célja	Teszeli, hogy az Orangutan megfelőlen lép-e a megadott mezőre.

Teszt-eset neve	1.2 Orangutan catches Panda
Rövid leírás	Orangutan egy szabad pandaval való ütközés tesztelése.
Teszt célja	Teszeli, hogy ha az Orangutan egy szabad Pandaval ütközik, akkor azt megfelelően berakja-e a követői sorába.

Teszt-eset neve	1.3 Orangutan exit
Rövid leírás	Orangutan „kimegy” a kijáraton
Teszt célja	Teszeli, hogy amennyiben az Orangutan több Panda követővel használja a kijáratot akkor megkapja-e a pontot és a Pandák elhagyják-e a játékot és ő pedig a Bejárat-ra kerül.

Teszt-eset neve	1.4 Orangutan goes Wardrobe
Rövid leírás	Orangutan és a Szekrény interakciójának tesztelése.
Teszt célja	Teszeli, hogy az Orangutan átkerül-e egy másik Szekrénybe és a követői megfelelően követik-e őt

Teszt-eset neve	1.5 Orangutan die
Rövid leírás	Orangutan halálának tesztelése.
Teszt célja	Teszeli, hogy amennyiben az Orangutan meghal akkor a követői által alkotott sor fel bomlik-e, illetve ő kikerül a játékból.

Teszt-eset neve	1.6 Orangutan steal Pandas
Rövid leírás	Orangutan ellopja egy másik-tól az elfogott Pandakat
Teszt célja	Teszeli, hogy amennyiben egy követő nélküli Orangutan egy másiknak megy, amely rendelkezik követőkkel akkor azokat „ellopja-e” és helyet cserél-e az Orangutannal.

Teszt-eset neve	1.7 Orangutan set Pandas free
Rövid leírás	Orangutan elengedi a követőit
Teszt célja	Teszeli, hogy ha egy Orangutan elakarja engedni a követőit akkor az őt követő sor fel bomlik-e.

Teszt-eset neve	2.1 Panda step
Rövid leírás	Panda lépésének tesztelése.
Teszt célja	Teszeli, hogy a Panda megfelelően lép-e külön szabad Pandara és elfogott Pandara

Teszt-eset neve	2.2 Panda exit
Rövid leírás	Panda megpróbál kimenni a kijáraton
Teszt célja	Teszeli, hogy amennyiben egy szabad Panda megpróbálja elhagyni a Szintet akkor az helyesen meghiusul-e.

Teszt-eset neve	2.3 Panda enters Wardrobe
Rövid leírás	Panda és a Wardrobe interakciója
Teszt célja	Teszeli, hogy egy szabad Panda átkerül-e egy másik Szekrénybe-e.

Teszt-eset neve	2.4 Panda die
Rövid leírás	Panda halálának a tesztelése
Teszt célja	Teszeli, hogy ha egy Panda meghal, akkor amennyiben van követője a sor mögötte helyesen felbomlik-e, illetve ő kikerül-e a játékból.

Teszt-eset neve	3.1 Sofa activate
Rövid leírás	Fotel interakciója a környező mezőkön lévő állatokkal
Teszt célja	Teszeli, hogy a Fotel aktiválódik-e, illetve a szomszédos mezőn lévő állatok megfelelően reagálnak-e rá és az Álmos Panda bele ül-e.

Teszt-eset neve	3.2 GamblingMachine activate
Rövid leírás	Játékgép interakciója a környező mezőkön lévő állatokkal
Teszt célja	Teszeli, hogy a Játékgép aktiválódik-e, illetve a szomszédos mezőn lévő állatok megfelelően reagálnak-e rá és a Félős Panda felbontja-e a mögötte lévő sort.

Teszt-eset neve	3.3 ChocolateMachine activate
Rövid leírás	Csokiautomata interakciója a környező mezőkön lévő állatokkal
Teszt célja	Teszeli, hogy a Csokiautomata aktiválódik-e, illetve a szomszédos mezőn lévő állatok megfelelően reagálnak-e rá és az Ugrós Panda ugor-e így csökkentve a Törékeny csempe élettartalmát.

Teszt-eset neve	4.1 Field breaks
Rövid leírás	Csempe eltörök
Teszt célja	Teszeli, hogy amennyiben egy 1 élettartamú csempére rálépnek akkor az eltörök-e és a rálépő állat meghal-e

Teszt-eset neve	5.1 New Turn
Rövid leírás	Új kör elkezdésének a tesztje
Teszt célja	Teszeli, hogy a kiválasztott pályát megfelelően betölts-e, illetve a rajta lévő állatokat és tárgyakat elhelyezze-e a pályán.

7.4 Tesztelést támogató segéd- és fordítóprogramok specifikálása

Mivel a játék minden lehetőségét nehézkes lenne kiképyszeríteni véges idő alatt, azért a tesztelés segédprogrammal történik. Ez a segédprogram egyrészt a standard input-ot használja előre definiált parancssorozatok bevitelére, illetve a standard outputon megjelenő eredmények alapján eldönti, hogy a teszt sikeres volt-e vagy sem. Ezen felül bizonyos esetekben a pályafájlokat is kiértékeli, hiszen ezeken strukturáltan vannak az adatok elhelyezve, így ennek elemzése valamivel egyszerűbb a standard outputnál. Fontos, hogy a segédprogram csak a prototípus interface-definíciója szakasz által tartalmazott parancsok, utasítások és üzenetek értelmezésére képes, ezzel egy valós játékost szimulálva.

7.5 Napló

Kezdet	Időtartam	Résznevők	Leírás
2019.03.28. 21:00	2 óra	Hegedüs	A pályakonfigurációs fájl megtervezése, leírásának elkészítése (7.1.2.2 szakasz).
2019.03.29. 10:00	2,5 óra	Hegedüs	A prototípus interface-definíciójának elkészítése és formázása (7.1 szakasz) a 7.1.3.3 szakasz kivételével.
2019.03.29. 15:00	1 óra	Hegedüs	A játékmenet megértését, átlátását segítő üzenetek (7.1.3.3 szakasz) elkészítése és formázása.
2019.03.30 9:00	3 óra	Borsodi	Változások tervezése, diagramok, leírások elkészítése.
2019.03.30 20:00	2 óra	Bódi	7.3 szakasz elkészítése.
2019.03.31:19:00	1 óra	Nyári	Összes részletes use-case elkészítése (7.2 szakasz).

8. Részletes tervezés

8.1 Osztályok és metódusok tervezése.

8.1.1 Activateable

- **Felelősség**

Absztrakt (ős)osztály, amely biztosítja, hogy a belőle származó osztályok minden körben aktiválhatók. Az aktiválás bizonyos speciális képességek előhívását jelenti.

- **Ősosztályok**

FieldElement → Activateable

- **Attribútumok**

- - **id**: az objektum azonosítója

- **Metódusok**

- + **void activate()**: aktiválja az objektumot
- + **boolean random()**: egy véletlenszámot generál, és ettől függően true vagy false lesz a visszatérési értéke. Ez biztosítja, hogy véletlenszerűen aktiválódjon

8.1.2 ChocolateMachine

- **Felelősség**

Csokiautomatát reprezentáló osztály, amely bizonyos speciális képességekkel rendelkező Panda objektumokat ugraszt meg azzal, hogy néha aktiválódik és sípol.

- **Ősosztályok**

FieldElement → Activateable → ChocolateMachine

- **Metódusok**

- + **void activate()**: felülírja az örökölt metódust

8.1.3 CowardPanda

- **Felelősség**

Egy olyan pandát valósít meg, amelyik megijed a játékgép hangjától.

- **Ősosztályok**

FieldElement → Steppable → Panda → CowardPanda

- **Metódusok**

- + **gmActivated()**: azt jelenti, hogy egy játékgép aktiválódott egy szomszédos mezőn. Ettől a panda megijed.
- + **scared()**: Megijedt a játékgép hangjától, ezért ugrik egyet a mezőjén.

8.1.4 Exit

- **Felelősség**

Kijáratot reprezentáló osztály. Az orángutánok és a pandák ezen keresztül tudják elhagyni a játékteret, majd ezt követően a bejáraton tudnak visszatérni.

- **Ősosztályok**

FieldElement → Exit

- **Attribútumok**

- - **entrance:** Field: A kijárathoz tartozó bejárat(egy mező).

- **Metódusok**

- + **setEntrance(Field):** Beállítja a bejáratot a paraméterben kapott mezőre
- + **getEntrance(): Field:** Visszaadja a kijárathoz tartozó bejáratot.
- + **hitBy(Panda): boolean:** Közli a pandával, ami rá akarna lépni, hogy nem lehet ki, csak ha egy panda vezeti ki.
- + **hitBy(Orangutan): boolean:** A paraméterként kapott orángutánt a bejárathoz küldi, a követőivel közli, hogy kivezette őket egy orángután, illetve az orángutánnak beirja a pontokat.

8.1.5 Field

- **Felelősség**

Egy mezőt reprezentáló osztály. FieldElement-ek helyezkednek el rajta. Ezekből a mezőkből áll össze a pálya.

- **Attribútumok**

- - **fieldElement: FieldElement:** A mezőn álló elem
- - **durability: int:** A mező tartóssága
- - **fragile: boolean:** törékeny-e a mező
- - **neighbours: List<Field>:** A szomszédos mezők
- - **id:** az objektum azonosítója

- **Metódusok**

- + **setFieldElement(FieldElement fieldElement):** beállítja a mezőn álló elemet a paraméterben kapott elemre
- + **getFieldElement(): FieldElement:** visszaadja a mezőn álló elemet
- + **setDurability(int durability):** Beállítja a mező tartósságát a kapott értékre
- + **getDurability(): int:** Visszaadja a mező tartósságát
- + **setFragile(boolean fragile):** Beállítja, hogy a mező törékeny-e
- + **isFragile(): boolean:** visszaadja hogy a mező törékeny-e
- + **getNeighbour(int idx): Field:** Visszaadja az adott indexnek megfelelő szomszédot
- + **accept(Steppable st): boolean:** Visszaadja, hogy a paraméterben adott Steppable a mezőre léphet-e
- + **moveTo(Field f, Steppable st): boolean:** Megadja, hogy a paraméterben átadott Steppable az ugyancsak paraméterben adott mezőre léphet-e
- + **remove(FieldElement f):** Eltávolítja a paraméterben kapott elemet a mezőről

- + **scareNeighbours()**: Megijeszti a szomszédos mezőket(sípol)
- + **jumpNeighbours()**: Megijeszti a szomszédos mezőket(csilingel)
- + **getFieldElement()**: **FieldElement**: visszaadja a mezőn álló elemet
- + **sleepNeighbours(Sofa s)**: Jelzi a szomszédoknak, hogy van itt egy fotel, és ha egy sleepyPanda jár arrafelé, akkor bele kell ülnie
- + **pandaJumped(Panda p)**: Ugrott egy panda a mezőn
- + **decDurability()**: Csökkenti a mező tartósságát

8.1.6 FieldElement

- **Felelősség**

A mezőn lévő elemek absztrakt ősosztálya. Biztosítja, hogy minden mezőre helyezhető elem rendelkezzen a benne lévő metódusokkal.

- **Attribútumok**

- - **field: Field**: A mező, amin áll
- - **id**: az objektum azonosítója

- **Metódusok**

- + **setField(Field field)**: Beállítja a mezőt a kapott értékre
- + **getField(): Field**: Visszaadja a mezőt, amin áll
- + **hitBy(Panda p)**: **boolean**: Panda próbál ugyanarra a mezőre lépni, visszaadja, hogy léphet e
- + **hitBy(Orangutan o)**: **boolean**: Orangután próbál ugyanarra a mezőre lépni, visszaadja, hogy léphet e
- + **sofaActivated(Sofa s)**: **boolean**: Szomszédos mezőn Sofa van
- + **gmActivated()**: Játékgép aktiválódott egy szomszédos mezőn
- + **cmActivated()**: Csokiautomata aktiválódott egy szomszédos mezőn

8.1.7 Floor

- **Felelősség**

A játékmenet vezérlését végző osztály. Tárolja a mezőket, tárgyakat, az orángutánokat és a pandákat. Új kört indít, töröl, feldolgozza a felhasználó parancsát.

- **Attribútumok**

- - **fields: List<Field>**: A pálya mezői
- - **activateables: List<Activateable>**: A pálya aktiválható elemei
- - **orangutans: List<Orangutans>**: A pálya orángutánjai
- - **pandas: List<Panda>**: A pálya pandái
- - **id**: az objektum azonosítója

- **Metódusok**

- + **addField(Field f)**: Hozzáad egy mezőt a pályához
- + **add(Activateable a)**: Hozzáad egy aktiválható elemet
- + **add(Orangutan o)**: Hozzáad egy orángutánt
- + **add(Panda p)**: Hozzáad egy pandát

- + **remove(Orangutan o)**: A paraméterben kapott orángutánt törli a pályáról
- + **remove(Panda p)**: A paraméterben kapott pandát törli a pályáról
- + **newTurn()**: Új kör indul
- + **waitForStep(Orangutan o)**: **Field**: Vár, hogy a felhasználó léptesse az orángutánt

8.1.8 GamblingMachine

- **Felelősség**

Játékgépet reprezentáló osztály, amely bizonyos speciális képességű Panda objektumokat ijeszt meg azzal, hogy néha aktiválódik és csilingel.

- **Ősosztályok**

FieldElement → Activateable → GamblingMachine

- **Metódusok**

- + **activate()**: Aktiválja a játékgépet. Felülírja az örökölt függvényt.

8.1.9 Game

- **Felelősség**

A játékot irányító osztály, felelős a játék elindításáért és befejezéséért.

- **Attribútumok**

- - **id**: az objektum azonosítója

- **Metódusok**

- + **getInstance(): Game**: Visszaadja az osztály egyetlen példányát, vagy létrehozza, ha még nincs.
- + **newGame(String map)**: Elindít egy új játékot a paraméterben megadott pályán
- + **endGame()**: A játék vége

8.1.10 JumpyPanda

- **Felelősség**

Egy olyan pandát valósít meg, ami fél a csokiautomata sípolásától. Ha meghallja, akkor ugrik egyet.

- **Ősosztályok**

FieldElement → Steppable → Panda → JumpyPanda

- **Metódusok**

- + **cmActivated()**: Csokiautomata aktiválódott a szomszédos mezőn
- + **jump()**: A panda ugrik egyet

8.1.11 Orangutan

- **Felelősség**

Egy orángutánnak feleltethető meg, annak megvalósításáért felel.

- **Ősosztályok**

FieldElement → Steppable → Orangutan

- **Attribútumok**

- - **score: int**: Az orángután pontjai
- - **floor: Floor**: Az emelet, amin az orángután van

- **Metódusok**

- + **setScore(int score)**: Beállítja az orángután pontjait
- + **getScore(): int**: Visszaadja az orángután pontjait
- + **setFloor(Floor floor)**: Beállítja az emeletet
- + **getFloor(): Floor**: Visszaadja az emeletet, amin van
- + **collideWith(FieldElement fe): boolean**: Az orángután ütközik egy elemmel
- + **caught(Panda p)**: A paraméterben adott pandát elkapta, ezért beteszi a sorba
- + **incScore()**: Növeli az orángután pontjainak a számát
- + **die()**: Az orángután meghal

8.1.12 Panda

- **Felelősség**

Egy pandát reprezentáló osztály.

- **Ősosztályok**

FieldElement → Steppable → Panda

- **Attribútumok**

- - **floor: Floor**: Az emelet, amin van
- - **catcher: Orangutan**: Az elkapó orángután

- **Metódusok**

- + **setFloor(Floor floor)**: Beállítja az emeletet
- + **getFloor(): Floor**: Visszaadja az emeletet, amin van
- + **setCatcher(Orangutan catcher)**: Beállítja az elkapó orángutánt
- + **getCatcher(): Orangutan**: Visszaadja az elkapó orángutánt
- + **step()**: A panda lép
- + **exitReached()**: A vezető orángután kilépett a kijáraton, ezért őt is kivezette
- + **hitBy(Orangutan o): boolean**: Ütközik egy orángutánnal, ami el is kapja ezért.
- + **hitBy(Panda p): boolean**: Ütközik egy pandával, ezért nem tud oda lépni
- + **releaseBoth()**: Elengedi az őt vezető és követő panda kezét is(illettve orángutánét)
- + **collideWith(FieldElement fe): boolean**: Az panda ütközik egy elemmel
- + **die()**: Az orángután meghal

8.1.13 SleepyPanda

- **Felelősség**

Fáradékony pandát valósít meg. Ha fotel mellé ér, akkor egy körig beleül.

- **Ősosztályok**

FieldElement → Steppable → Panda → SleepyPanda

- **Metódusok**

- + **sofaActivate(Sofa s): boolean**: A szomszédos mezőn egy fotel van, visszaadja, hogy sikeresült-e belülni
- + **sleep(Sofa s)**: Beül egy fotelbe egy körre.

8.1.14 Sofa

- **Felelősség**

Egy fotel működését valósítja meg. Ha egy fáradékony panda elhalad a szomszédos mezőn, akkor beleül egy körre.

- **Ősosztályok**

FieldElement → Activateable → Sofa

- **Metódusok**

- + **setPanda(Panda p)**: Beállítja a pandát, ami a fotelben van
- + **getPanda(): Panda**: Visszaadja a pandát, ami a fotelben van
- + **sit(Panda p)**: A paraméterben adott panda beleül a fotelbe
- + **activate()**: Ha üres, minden körben szól a szomszédainak, hogy üres.

8.1.15 Steppable

- **Felelősség**

Egy absztrakt osztály, ami minden olyan tárgynak az ősosztálya, ami mozgatható a pályán.

- **Ősosztályok**

Field → FieldElement

- **Attribútumok**

- - **stepped: boolean**: Lépett-e az adott körben
- - **follower: Panda**: A sorban következő panda, ha van
- - **anterior: Steppable**: A sorban előtte haladó
- - **lastStepperOn: Field**: A mező, amire utoljára lépett

- **Metódusok**

- + **setStepped(boolean stepped)**: Beállítja, hogy a körben már lépett-e
- + **setAnterior(Steppable st)**: Beállítja a sorban előtte levőt
- + **getAnterior(): Steppable**: Visszaadja a sorban előtte levőt
- + **isStepped(): boolean**: Visszaadja, lépett-e a körben

- + **setFollower(Panda follower)**: Beállítja a sorban utána következőt
- + **getFollower(): Panda**: Visszaadja a sorban következő pandát
- + **setLastSteppenOn(Field field)**: Beállítja a mezőt, ahová utoljára lépett
- + **step(Field f): boolean**: A paraméterben adott Fieldre próbál lépni
- + **releaseFollower()**: Elengedi a követőjét
- + **collideWith(FieldElement fe): boolean**: Absztrakt függvény, ütközik egy elemmel
- + **die()**: Absztrakt függvény, meghal

8.1.16 Wardrobe

- **Felelősség**

Egy szekrényt valósít meg. Ha egy Steppable belelép egybe, akkor egy másik szekrényből bukkan elő, ez véletlenszerűen történik.

- **Ősosztályok**

FieldElement → Wardrobe

- **Attribútumok**

- - **target: Wardrobe**: A célszekrény
- - **targetField: Map<Orangutan, field>**: minden orangutához tárolja azt a mezőt, ahová utoljára rakta

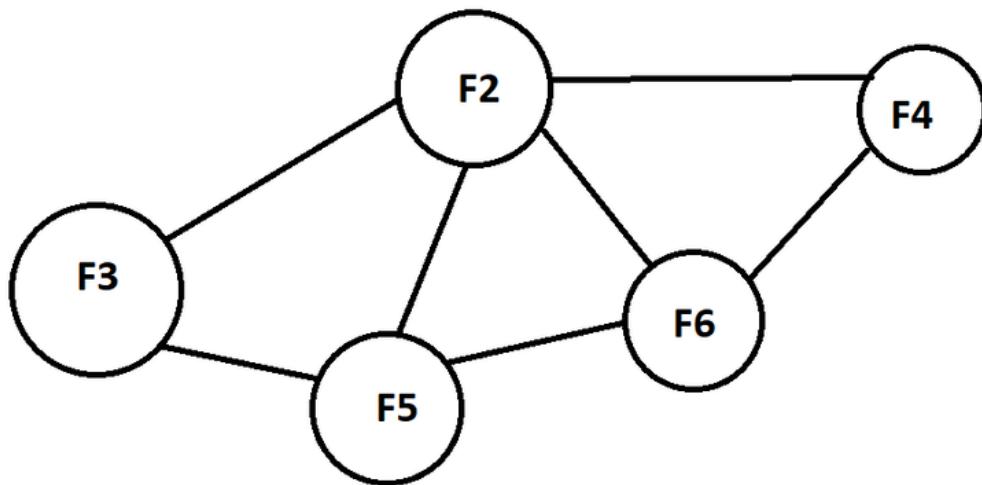
- **Metódusok**

- + **setTarget(Wardrobe target)**: Beállítja a célszekrényt
- + **getTarget(): Wardrobe**: Visszaadja a célmezőt
- + **setTargetField(Map<Orangutan, Field> targetField)**: Átállítja az orangutánokhoz tartozó utolsó helyeket, ahova rakta őket
- + **getTargetfield(): Map<Orangutan, Field>**: Visszaadja, melyik orangutánt hova rakta utoljára
- + **receive(Orangutan o): boolean**: Valamelyik szomszédos mezőre rakja a kapott orangutánt
- + **receive(Panda p): boolean**: Valamelyik szomszédos mezőre rakja a kapott pandát.
- + **hitBy(Panda p): boolean**: Egy panda megy be a szekrénybe
- + **hitBy(Orangutan o): boolean**: Egy orangután lép a szekrénybe

8.2 A tesztek részletes tervez, leírásuk a teszt nyelvén

A tesztesetek külön pályákat töltenek be, amik az alap tesztpálya mezőstruktúráját tartalmazzák, és még azon egyéb elemek lehetnek elhelyezve, amiket a teszt leírása tartalmaz. Pl: Orangutan(8,2) egy orangután, akinek az id értéke 8, a mezőjének id-je 2.

Az alap tesztpálya vizuálisan így néz ki: (Fi olyan mező, ahol az id=i)



A bemeneti nyelv változásai:

test - a parancs kiadása engedélyezi az alábbi parancsok kiadását, és determinisztikus módba kapcsol (azaz, ha például egy panda véletlenszerűen lépne, a lehető legkisebb id-jű mezőre próbál lépni).

step <Steppable id> <Field id>

activate <Activateable id>

8.2.1 Orangutan step

- Leírás

Az Orangutan lépésének a tesztelése.

A tesztpálya tartalma: Orangutan(8,2), a szomszédos mező id=3

- Ellenőrzött funkcionalitás, várható hibahelyek

Az Orangutan sikeresen lép a id=3 mezőre

- Bemenet

load map –orangutanstep

test

step 8 3

- Elvárt kimenet

MESSAGE: Map loading successful!

MESSAGE: Orangutan 8 stepped to Field 3.

8.2.2 Orangutan catches Panda

- Leírás

Az Orangutan egy szabad pandával való ütközésének tesztelése.

A tesztpálya tartalma: Orangutan(8, 2), Panda(7, 3).

- Ellenőrzött funkcionalitás, várható hibahelyek

Mivel id=7 panda szabad panda ezért sikeres elkapás üzenetet várunk.

- Bemenet

load map –orangutancatchespanda

test

step 8 3

- Elvárt kimenet

MESSAGE: Map loading successful!

MESSAGE: Orangutan 8 caught Panda 7.

MESSAGE: Panda 7 stepped to Field 2.

MESSAGE: Orangutan 8 stepped to Field 3.

8.2.3 Orangutan exit

- **Leírás**

Orángután kimegy a kijáraton.

A tesztpálya tartalma: Orangutan(8,2), Exit(7, 3), Panda(9, 4), Panda(10, 6), Entrance(11,5).

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Az Orángután kivezeti az elfogott pandákat és megkapja az utánuk járó pontokat

- **Bemenet**

load map –orangutanexit

test

step 8 3

- **Elvárt kimenet**

MESSAGE: Map loading successful!

MESSAGE: Panda 9 exited.

MESSAGE: Orangutan 8 Points increased by 1.

MESSAGE: Panda 10 exited.

MESSAGE: Orangutan 8 Points increased by 1.

MESSAGE: Panda 10 died.

MESSAGE: Panda 9 died.

MESSAGE: Orangutan 8 stepped to Field 5.

8.2.4 Orangutan goes Wardrobe

- **Leírás**

Az Orangutan és a szekrény interakciójának tesztelése.

A tesztpálya tartalma: Orangutan(8, 2), Panda(7, 6), Wardrobe(9, 4), Wardrobe(10, 3) .

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Az Orangutan átkerül egy másik szekrény melletti mezőre majd miután újból lépett, a pandája is.

- **Bemenet**

load map –orangutangoeswardrobe

test

step 8 4

step 8 6

- **Elvárt kimenet**

MESSAGE: Map loading successful!

MESSAGE: Orangutan 8 stepped into Wardrobe 9.

MESSAGE: Orangutan 8 stepped out from Wardrobe 10.

MESSAGE: Orangutan 8 stepped to Field 5.

MESSAGE: Panda 7 stepped to Field 2.

MESSAGE: Orangutan 8 stepped to Field 6.

MESSAGE: Panda 7 stepped into Wardrobe 9.

MESSAGE: Panda 7 stepped out from Wardrobe 10.

MESSAGE: Panda 7 stepped to Field 5.

8.2.5 Orangutan die

- **Leírás**

Az orangutan meghal. A pályán a 4-es mező törött.

A tesztpálya: Orangutan(7, 2), Panda(8, 3), Panda(9, 5).

- **Ellenőrzött funkcionális, várható hibahelyek**

Az Orangutan egy törött csempére lépett így meghal. Ő kikerül a játékból a követő pandák szabdon lesznek.

- **Bemenet**

load map –orangutandie

test

step 8 4

- **Elvárt kimenet**

MESSAGE: Map loading successful!

MESSAGE: Orangutan 8 stepped to Field 4.

MESSAGE: Orangutan 8 died.

MESSAGE: Orangutan 8 lost Panda 8.

MESSAGE: Orangutan 8 lost Panda 9.

8.2.6 Orangutan steal Pandas

- **Leírás**

Az orangutan ellopja egy másiktól az elfogott pandáit.

A tesztpálya: Orangutan(7, 2), Panda(8, 3), Panda(9,5), Orangutan(10,6).

- **Ellenőrzött funkcionális, várható hibahelyek**

Az orangutan sikeresen ellopja a pandákat és helyet cserél a másikkal.

- **Bemenet**

load map –orangutanstealpandas

test

step 10 6

- **Elvárt kimenet**

MESSAGE: Map loading successful!

MESSAGE: Orangutan 10 stole Pandas from Orangutan 7.

MESSAGE: Orangutan 7 stepped to Field 6.

MESSAGE: Orangutan 10 stepped to Field 2.

8.2.7 Orangutan set Pandas free

- **Leírás**

Az Orangutan engedi a pandáit.

A tesztpálya: Orangutan(7,2), Panda(8, 3), Panda(9, 5).

- **Ellenőrzött funkcionális, várható hibahelyek**

Az Orangutan sikeresen felszabadítja a pandákat.

- **Bemenet**

load map –orangutansetpandasfree

start

Y

stop

- **Elvárt kimenet**

MESSAGE: Map loading successful!

MESSAGE: Game started.

MESSAGE: Does Orangutan 7 set free his pandas?(Y/N)

MESSAGE: Orangutan 7 lost Panda 8.

MESSAGE: Orangutan 7 lost Panda 9.

MESSAGE: Orangutan 7 has to step. Give a field ID! (ID)

MESSAGE: Game stopped.

8.2.8 Panda step

- Leírás

Az elfogott, és a szabad pandák lépését teszteli. A tesztpálya tartalma: Orangutan(7,2), az öt követő Panda(8,3), és egy szabad Panda(9,6). A szabad panda és az orángután lépnek a teszt során.

- Ellenőrzött funkcionális, várható hibahelyek

Az orángután lépése után a pandája is lép.

- Bemenet

load map -pandastep

test

step 7 6

step 7 4

step 9 5

- Elvárt kimenet

MESSAGE: Map loading successful!

MESSAGE: Orangutan 7 could not step to Field 6.

MESSAGE: Orangutan 7 stepped to Field 4.

MESSAGE: Panda 8 stepped to Field 2.

MESSAGE: Panda 9 stepped to Field 5.

8.2.9 Panda exit

- Leírás

A Panda kijáratra lépését teszteli. A tesztpálya tartalma : Panda(7,2), Exit(8,5).

- Ellenőrzött funkcionális, várható hibahelyek

A panda nem léphet ki a kijáraton, azaz nem szabad olyan üzenetet kapnunk, hogy kilépett.

- Bemenet

load map -pandaexit

test

step 7 5

- Elvárt kimenet

MESSAGE: Map loading successful!

8.2.10 Panda enters Wardrobe

- Leírás

A panda belép egy szekrénybe, ami egy másik szekrénynek küldi át. A tesztpálya tartalma: Panda(7,2), Wardrobe(8,3), Wardrobe(9,4), és a két szekrény egymással párban áll.

- Ellenőrzött funkcionális, várható hibahelyek

A panda nem tud beleférni a szekrénybe, vagy a fogadó szekrény nem a megfelelő mezőre teszi.

- Bemenet

load map -pandawardrobe

test

step 7 3

- **Elvárt kimenet**

MESSAGE: Map loading successful!

MESSAGE: Panda 7 stepped into Wardrobe 8.

MESSAGE: Panda 7 stepped out from Wardrobe 9.

MESSAGE: Panda 7 stepped to Field 6.

8.2.11 Panda die (+ Field breaks)

- **Leírás**

Az orángutánt követ 2 panda, és egy törékeny, 1-es tartósságú mezőn áll. Az orángután lép egyet, majd a pandák lépnek utána.

A tesztpálya tartalma: Orangutan(7,2),Panda(8,3),Panda(9,5).

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A mezőnek el kell törnie, a rálépő panda le kell, hogy essen, a sor pedig fel kell hogy bomoljon.

- **Bemenet**

load map -pandadie

test

step 7 4

- **Elvárt kimenet**

MESSAGE: Map loading successful!

MESSAGE: Orangutan 7 stepped to Field 4.

MESSAGE: Durability decrementation of Field 2.

MESSAGE: Field 2 has broken.

MESSAGE: Panda 8 died.

MESSAGE: Orangutan 7 lost Panda 8.

MESSAGE: Orangutan 7 lost Panda 9.

8.2.12 Sofa activate

- **Leírás**

A pályán álló fotel aktiválódik, és egy panda beleül, majd a fotel következő aktiválódásánál feláll onnan.

A tesztpálya tartalma: SleepyPanda(7,2),Sofa(8,5).

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Az álmos panda bele kell, hogy üljön a fotelbe, mivel az üres. A második aktiválódásnál nem maradhat benne, és nem is ülhet vissza rögtön, mert most állt föl.

- **Bemenet**

load map -sofaactivate

test

activate 8

activate 8

- **Elvárt kimenet**

MESSAGE: Map loading successful!

MESSAGE: Activateable 8(Sofa) was activated.

MESSAGE: Panda 7 sat down on Field 5.

MESSAGE: Activateable 8(Sofa) was activated.

MESSAGE: Panda 7 stepped to Field 2.

8.2.13 GamblingMachine activate

- **Leírás**

A pályán álló játékgép aktiválódik, miközben mellette egy elfogott panda áll, akit követ mégegy panda a sorban.

A tesztpálya tartalma: Orangutan(7,4), CowardPanda(8,2), SleepyPanda(9,3), GamblingMachine(10,5)

- **Ellenőrzött funkcionális, várható hibahelyek**

Az ijesztő pandának meg kell ijesztenie az aktiválódás hatására, és el kell engednie a mögötte álló panda kezét.

- **Bemenet**

```
load map -gmactivate
```

```
test
```

```
activate 10
```

- **Elvárt kimenet**

MESSAGE: Map loading successful!

MESSAGE: Activateable 10(GamblingMachine) was activated.

MESSAGE: Panda 8 got scared on Field 2.

MESSAGE: Orangutan 7 lost Panda 9.

8.2.14 ChocolateMachine activate

- **Leírás**

A pályán álló csokiautomata aktiválódik, miközben mellette egy ugrós panda áll, aki törékeny mezőn áll.

A tesztpálya tartalma: JumpyPanda(7,2), ChocolateMachine(8,5).

- **Ellenőrzött funkcionális, várható hibahelyek**

A pandának ugrania kell az aktiválódás hatására, és el kell engednie a mögötte álló panda kezét.

- **Bemenet**

```
load map -cmactivate
```

```
test
```

```
activate 8
```

- **Elvárt kimenet**

MESSAGE: Map loading successful!

MESSAGE: Activateable 8(ChocolateMachine) was activated.

MESSAGE: Panda 7 jumped on Field 2.

MESSAGE: Durability decrementation of Field 2.

8.2.15 New turn

- **Leírás**

A pályán új kör kezdődik. A játékos lép egy orángutánnal majd a pályán álló szabad panda lép magától, és aktiválódik egy fotel is a körben. Ezután a játékot leállítjuk.

A tesztpálya tartalma: Orangutan(7,4), ScaryPanda(8,2), Sofa(9,5).

- **Ellenőrzött funkcionális, várható hibahelyek**

A játék elindulása után a játékos kiadhatja, hogy az orángután melyik mezőre lépjen, majd a szabad panda, és a fotel maguktól cselekszenek.

- **Bemenet**

```
load map -newturn
```

```
test
```

```
start
```

6

stop

- **Elvárt kimenet**

MESSAGE: Map loading successful

MESSAGE: Game started.

MESSAGE: Round of orangutans.

MESSAGE: Does Orangutan 7 set free his pandas? (Y/N)

MESSAGE: Orangutan 7 has to step. Give a field ID! (ID)

MESSAGE: Orangutan 7 stepped to Field 6.

MESSAGE: Round of pandas.

MESSAGE: Panda 8 stepped to Field 3.

MESSAGE: Activateable 9(Sofa) was activated.

MESSAGE: Round of orangutans.

MESSAGE: Does Orangutan 7 set free his pandas? (Y/N)

MESSAGE: Game stopped.

A tesztelést támogató programok tervei

A tesztelést támogató program előre definiált teszteseteket tartalmaz, minden tesztesethez két szöveges állomány tartozik. Az egyik fájl felhasználói utasításokat tartalmazza, a másik pedig az ezekre az utasításokra várt válaszokat. A segédprogram elindítja a főprogramot, az előbb említett felhasználói utasításokat a fájlból a program standard inputjára irányítja, a standard outputon megjelenő eredményt pedig összehasonlítja az elvárt válaszokat tartalmazó fájl tartalmával. Ha megegyezik a főprogram által generált kimenet az elvárttal, akkor a teszt sikeresen lefutott. Ha valahol eltérést talál a kimenet és az elvárt üzeneteket tartalmazó fájl közt, akkor kiírja az első nem egyező sor sorszámát és tartalmát.

8.3 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2019.04.06. 10:00	2 óra	Borsodi	Alap tesztpálya, nyelv módosítása, Panda tesztek.
2019.04.06. 14:00	2 óra	Borsodi	Furniture + Floor tesztek.
2019.04.06. 16:00	1 óra	Borsodi	Panda,furniture,floor tesztek pályafájljai.
2019.04.06. 18:00	3 óra	Nyári	Osztályok és metódusok tervezet
2019.04.07. 6:30	1 óra	Nyári	Osztályok és metódusok tervezet
2019.04.07. 14:00	2 óra	Bódi	Orangutan tesztek
2019.04.07. 18:30	0,5 óra	Borsodi	Egységesítés, formázás (5.2).
2019. április 8.	0,5 óra	Hegedüs	A tesztelést támogató program terveinek elkészítése.

10. Prototípus beadása

10.1 Fordítási és futtatási útmutató

10.1.1 Fájllista

10.1.1.1 businesslogic

Fájl neve	Méret	Keletkezés ideje	Tartalom
Floor.java	8 KB	2019.04.23. 23:10	Floor osztály
Orangutan.java	5 KB	2019.04.23. 23:10	Orángután osztály
ChocolateMachine.java	1 KB	2019.04.23. 23:02	Csokiautomata osztály
CowardPanda.java	1 KB	2019.04.23. 23:02	CowardPanda osztály
GamblingMachine.java	1 KB	2019.04.23. 23:02	Játékgép osztály
Panda.java	5 KB	2019.04.23. 23:02	Panda osztály
SleepyPanda.java	2 KB	2019.04.23. 23:02	SleepyPanda osztály
Field.java	7 KB	2019.04.23. 21:58	Mező osztály
Wardrobe.java	6 KB	2019.04.23. 21:58	Szekrény osztály
MapParser.java	15 KB	2019.04.23. 19:57	Pályafájl értelmező
Sofa.java	2 KB	2019.04.23. 19:57	Sofa osztály
Steppable.java	3 KB	2019.04.23. 19:57	Steppable osztály
Activateable.java	1 KB	2019.04.23. 19:57	Activateable osztály
Exit.java	2 KB	2019.04.23. 19:57	Kilépési pont osztály
FieldElement.java	2 KB	2019.04.23. 19:57	FieldElement osztály
Game.java	3 KB	2019.04.23. 19:57	Game osztály
JumpyPanda.java	1 KB	2019.04.23. 19:57	JumpyPanda osztály

10.1.1.2 root

Fájl neve	Méret	Keletkezés ideje	Tartalom
Main.java	8 KB	2019.04.23. 23:10	Main osztály
cmactivate.map	1 KB	2019.04.23. 21:57	cmactivate pályafájl
gmactivate.map	2 KB	2019.04.23. 21:57	gmactivate pályafájl
newturn.map	1 KB	2019.04.23. 21:57	newturn pályafájl
orangutancatchespanda.map	1 KB	2019.04.23. 21:57	orangutan catches panda pályafájl
orangutandie.map	2 KB	2019.04.23. 21:57	orangutan die pályafájl
orangutanexit.map	2 KB	2019.04.23. 21:57	orangutan exit pályafájl
orangutangoeswardrobe.map	2 KB	2019.04.23. 21:57	orangutan goes wardrobe pályafájl
orangutansetpandasfree.map	2 KB	2019.04.23. 21:57	orangutan set pandas free pályafájl
orangutanstealpandas.map	2 KB	2019.04.23. 21:57	orangutan steal pandas pályafájl
orangutanstep.map	1 KB	2019.04.23. 21:57	orangutanstep pályafájl
pandidie.map	2 KB	2019.04.23. 21:57	panda die pályafájl
pandaexit.map	1 KB	2019.04.23. 21:57	panda exit pályafájl
pandastep.map	2 KB	2019.04.23. 21:57	panda step pályafájl
pandawardrobe.map	1 KB	2019.04.23. 21:57	panda wardrobe pályafájl

sofaactivate.map	1 KB	2019.04.23. 21:57	sofa activate pályafájl
runTest.bat	1 KB	2019.04.23. 22:47	tesztprogramot futtató batch file

10.1.2 Fordítás

A tesztprogram és a főprogram egy batch fájl futtatásával automatikusan fordul, ld.: Futtatás.

10.1.3 Futtatás

A letöltött test.zip fájl kicsomagolása után a gyökérkönyvárban lévő **runTest.bat** fájl a tesztprogramot automatikusan fordítja, majd futtatja. A tesztprogram futása során egy újabb batch fájl jön létre az src könyvtárban, ami a főprogram futtatásáért felelős. Utóbbival nem kell foglalkozni, annak indítását a tesztprogram a szükséges pillanatokban elvégzi. Ha a JAVA környezeti változói nincsenek beállítva az adott számítógépen, akkor a runTest.bat fájlra jobbegér gombbal kell kattintani és a lenyíló menüben ki kell választani az **Edit/Szerkesztés** lehetőséget. A megnyíló szövegszerkesztőben látható két kikommentezett sor (**::PATH**, **::JAVA_HOME**), melyek helyére beírható a környezeti változók beállításának parancsa. Az alábbiakban leírt két parancs használható (természetesen a megfelelő verziószámokkal) a beállítások elvégzésére.

::PATH helyére:	set PATH=%PATH%;c:\Program Files\Java\jdk1.6.0_07\bin
::JAVA_HOME helyére:	set JAVA_HOME=c:\Program Files\Java\jdk1.6.0_07

Jelen esetben a tesztprogram és a főprogram azonos verziójú JDK-val fordítható, de előfordulhat, hogy a két programot eltérő JDK-val kell fordítani, ezért a tesztprogram az indítás után egyszer bekéri a környezeti változók beállításához tartozó parancsokat, melyeket a főprogram batch fájljába fog beszűrni. A fent leírt parancsok itt is használhatók. Természetesen az ENTER billentyű leütésével a parancs megadása kihagyható, ha a számítógépen a környezeti változók jól vannak beállítva.

10.2 Tesztek jegyzőkönyvei

10.2.1 Orangutan Step

Tesztelő neve	Bódi
Teszt időpontja	2019. 04. 23. 22:00

10.2.2 Orangutan catches Panda

Tesztelő neve	Borsodi
Teszt időpontja	2019. 04. 23. 21:00

10.2.3 Orangutan exit

Tesztelő neve	Hegedüs
Teszt időpontja	2019. 04. 23. 23:00

Tesztelő neve	Borsodi
Teszt időpontja	2019.04.23. 21:00
Teszt eredménye	3 sor különbözik az elvárt kimenettől
Lehetséges hibaok	hibás teszteset definíció
Változtatások	teszteset elvárt kimenet módosítása

10.2.4 Orangutan goes Wardrobe

Tesztelő neve	Borsodi
Teszt időpontja	2019. 04. 23. 21:00

10.2.5 Orangutan die

Tesztelő neve	Borsodi
Teszt időpontja	2019. 04. 23. 21:30

10.2.6 Orangutan steal Pandas

Tesztelő neve	Hegedüs
Teszt időpontja	2019. 04. 23. 23:00

Tesztelő neve	Borsodi
Teszt időpontja	2019.04.23. 21:30
Teszt eredménye	nem jelenik meg semmi a kimeneten
Lehetséges hibaok	hibás teszteset bemenet
Változtatások	teszteset bemenetének javítása : step 10 6 helyett step 10 2

10.2.7 Orangutan set pandas free

Tesztelő neve	Hegedüs
Teszt időpontja	2019. 04. 23. 23:00

Tesztelő neve	Borsodi
Teszt időpontja	2019.04.23. 21:30
Teszt eredménye	nem az elvárt kimenet érkezett
Lehetséges hibaok	módosítás a program bemenetében a definiálthoz képest
Változtatások	teszeset bemenetének , és kimenetének módosítása

10.2.8 Panda step

Tesztelő neve	Hegedüs
Teszt időpontja	2019. 04. 23. 23:00

Tesztelő neve	Bódi
Teszt időpontja	2019. 04. 23. 20:00
Teszt eredménye	elvárt kimenet 2. sora hiányzik
Lehetséges hibaok	a programban egy output hibásan van megírva
Változtatások	program javítása

10.2.9 Panda Exit

Tesztelő neve	Bódi
Teszt időpontja	2019. 04. 23. 20:30

10.2.10 Panda enters Wardrobe

Tesztelő neve	Bódi
Teszt időpontja	2019. 04. 23. 20:30

10.2.11 Panda die

Tesztelő neve	Hegedüs
Teszt időpontja	2019. 04. 23. 23:00

Tesztelő neve	Bódi
Teszt időpontja	2019. 04. 23. 20:30
Teszt eredménye	hibás kimenet
Lehetséges hibaok	teszeset elvárt kimenetében 1 sor hiányzik
Változtatások	teszeset módosítása

10.2.12 Sofa activate

Tesztelő neve	Bódi
Teszt időpontja	2019. 04. 23. 21:00

10.2.13 Gambling Machine activate

Tesztelő neve	Hegedüs
Teszt időpontja	2019. 04. 23. 23:00

Tesztelő neve	Bódi
Teszt időpontja	2019. 04. 23. 21:00
Teszt eredménye	hibás kimenet
Lehetséges hibaok	teszteset elvárt kimenetében 1 sor hiányzik
Változtatások	teszteset javítása

10.2.14 ChocolateMachine activate

Tesztelő neve	Bódi
Teszt időpontja	2019. 04. 23. 21:00

10.2.15 New Turn

Tesztelő neve	Hegedüs
Teszt időpontja	2019. 04. 23. 23:00

Tesztelő neve	Borsodi
Teszt időpontja	2019. 04. 23. 22:30
Teszt eredménye	hibás kimenet
Lehetséges hibaok	módosítás a kimeneti nyelvben a tervezetthez képest.
Változtatások	Teszteset módosítása

10.3 Értékelés

Tag neve	Tag neptun	Munka százalékban
Borsodi Regő	HH5GMS	28 %
Bódi Dániel	KM3B5S	25 %
Hegedüs Amadé	P53ZLW	26 %
Nyári Tamás	X49AI8	21 %

10.4 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2019.04.10.	3,5 óra	Hegedüs	Tesztprogram prototípusának elkészítése, a bemenetek és elvárt kimenetek kezdetleges változatának elkészítése.
2019.04.19. 8:30	2,5 óra	Nyári	Osztályok kódjának írása
2019.04.19. 9:00	3 óra	Borsodi	MapParser + Game
2019.04.20. 16:00	1,5 óra	Bódi	Orangutan tesztekhez tartozó mapok létrehozása
2019.04.20. 16:00	4 óra	Borsodi	Main input + Floor + alap outputok.
2019.04.21 14:00	2 óra	Bódi	Orangutan változások + kimenetek
2019.04.23. 20:00	3 óra	Bódi	8 – 14. tesztek elvégzése , dokumentáció
2019. 04. 23. 20:30	3 óra	Borsodi	1 – 7. tesztek , tesztesetek/ program javítása.
2019. 04. 23. 20:30	3 óra	Hegedüs	Tesztprogram kiegészítése, hibás tesztek ellenőrzése.

11. Grafikus felület specifikációja

11.1 A grafikus interfész

11.1.1 A játékban előforduló objektumok

Panda		Fotel	
Orángután		Szekrény	
Játékgép		Bejárat	
Csokiautomata		Kijárat	

11.1.2 A csempeszínek jelentése

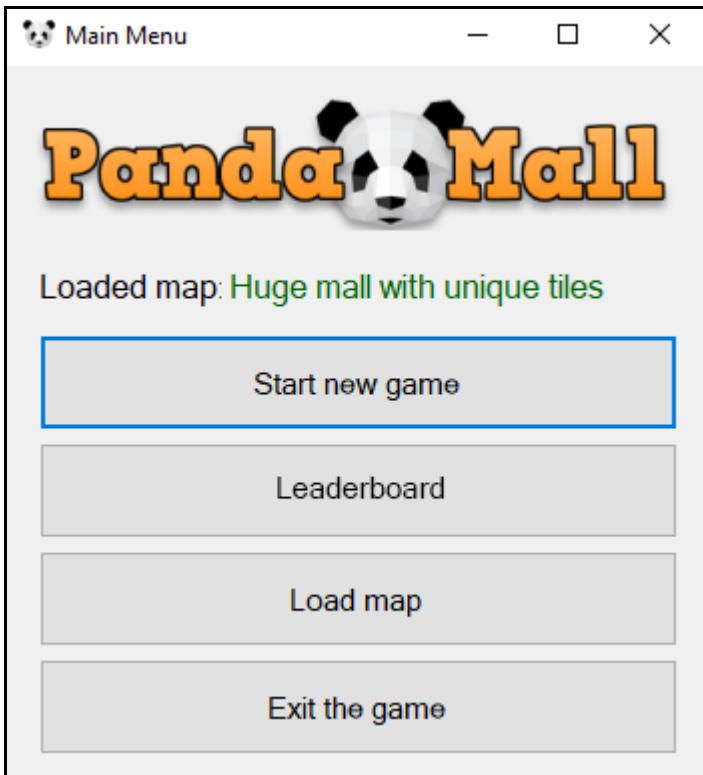
	Törékeny csempe
	Nem törékeny csempe
	Beszakadt csempe
	Lépni kívánó panda vagy orángután
	Lépni kívánó orángután követői

11.1.3 Egy pálya felépítése



11.1.4 A játék ablakai

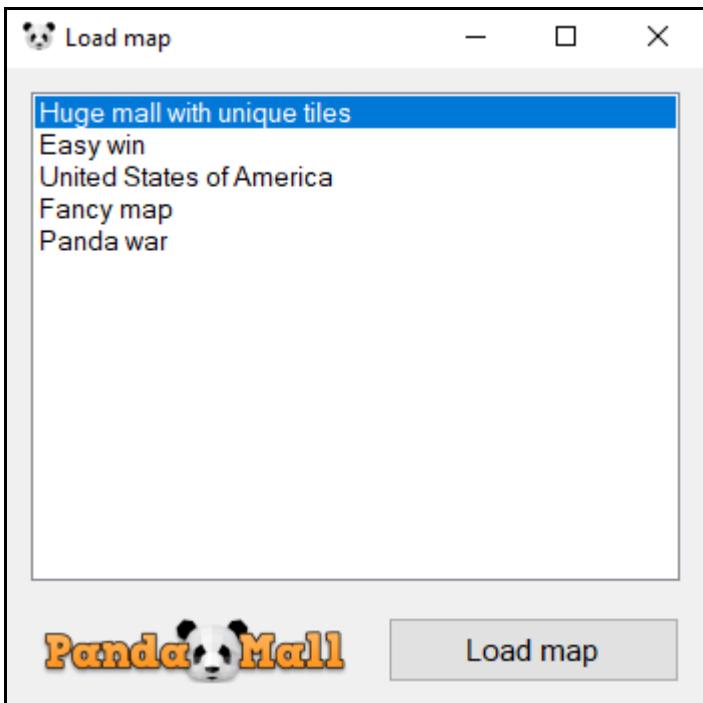
11.1.4.1 Indítási ablak/Főmenü



Vezérlőelemek:

- **Loaded map:** Az éppen betöltött pálya neve.
- **Start new game:** Új játék indítása a betöltött pályának megfelelően.
- **Leaderboard:** Ranglista (dicsőségtábla) megnyitása
- **Load map:** Betöltendő pálya kiválasztása.
- **Exit the game:** Kilépés a játékból.

11.1.4.2 Pályabetöltő ablak



Vezérlőelemek:

- **Load map:** A listából kiválasztott pálya betöltése.

11.1.4.3 Ranglista ablak



11.1.4.4 Játékablak



Vezérlőelemek:

- Loaded map:** Az éppen betöltött pálya neve.
- Round of:** Kiírja, hogy épp a pandák vagy az orangutánok lépnek.
- Pandas:** A pályán lévő pandák száma.
- Orangutans:** A pályán lévő orangutánok száma.
- Highlight followers:** Az éppen lépni kívánó orangutánt követő pandák alatti csempék kiemelése.
- Save status:** Aktuális állapot mentése.
- Exit the game:** Kilépés a játékból.

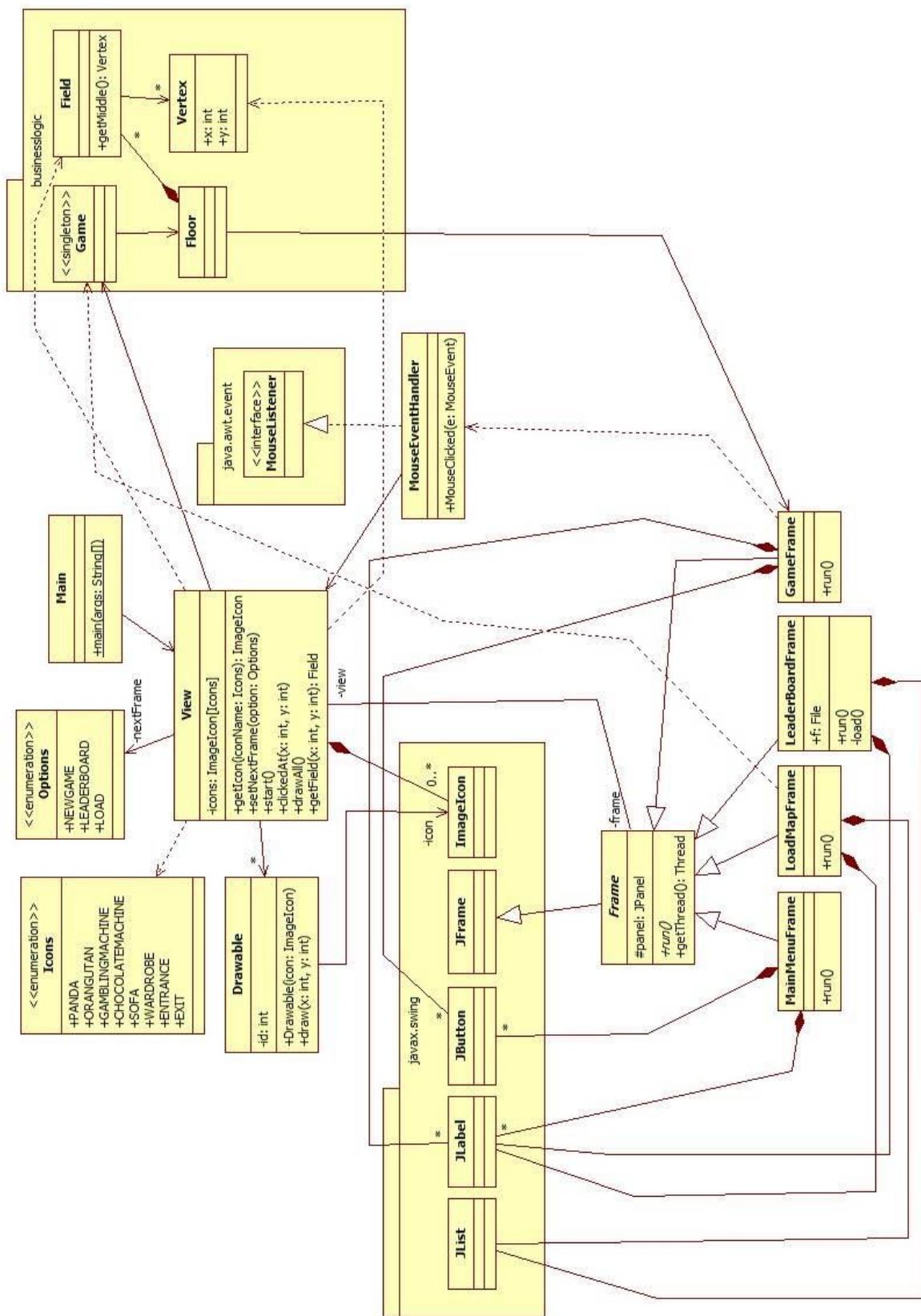
11.2 A grafikus rendszer architektúrája

11.2.1 A felület működési elve

A grafikus felület megfelel az MVC (Model - View - Controller) architektúrának, minden (képernyőn megjelenő) objektumnak van egy Model-beli és egy View-beli változata. A View-beli objektumok Drawable típusúak, mindegyik rendelkezik draw(x,y) függvényel, mely az adott objektum kirajzolását végzi a kapott képernyő-koordinátekra. A kirajzolás első lépése, hogy a View lekérdezi az adott Drawable Model-beli megfelelőjének állapotát, a második lépés pedig az állapot képernyőre való kirajzolása. minden változás során az egész képet újra kell rajzolni a drawAll() függvényel (nem tehető az meg, hogy egy adott objektumot eltávolítunk a képről, majd más pozícióba visszahelyezzük).

A játék indítása után a vezérlés a View-hez adódik, ami mindig egy Frame-t jelenít meg a képernyőn. Az aktív Frame eseménykezelői kommunikálnak a View-vel. A játék a GameFrame-ben valósul meg. Ennek a felületén történő kattintásokat a MouseEventHandler továbbítja a View-nek, ami kiszámítja, hogy melyik mezőn történt a kattintás, szól a Controllernek, és a pályát is újrarendezza.

11.2.2 A felület osztály-struktúrája



11.3 A grafikus objektumok felsorolása

11.3.1 Drawable

- **Felelősség**

A pálya kirajzolható elemei. Tárolják, hogy melyik ikont kell kirajzolni, és a draw függvényben megkapják, hogy a képernyőn milyen pozícióba kerüljön. Ezen keresztül lehet kirajzolni az adott ikonokat az adott koordinátákra.

- **Attribútumok**

- **-id: int:** Az kirajzolandó objektum azonosítója
- **-icon: ImageIcon:** Az adott objektumhoz tartozó ikon, amit kirajzol

- **Metódusok**

- **+Drawable(ImageIcon icon):** konstruktor, a paraméterben kapott ikont beállítja a saját ikonjának
- **+draw(int x, int y):** A paraméterben adott koordinátákra rajzolja az ikont

11.3.2 Field

- **Felelősség**

Egy mezőt reprezentáló osztály. FieldElement-ek helyezkednek el rajta. Ezekből a mezőkből áll össze a pálya.

- **Metódusok**

- **+getMiddle(): Vertex:** A mező közepének a koordinátáját adja meg.

11.3.3 Frame

- **Felelősség**

Az osztály a különböző ablakok ősosztálya. Ezek valósítják meg a játék grafikus interfészét.

- **Ősosztályok**

JFrame → Frame

- **Attribútumok**

- **-id: int:** Az objektum azonosítója
- **#panel: JPanel:** A keret fő panelje.
- **-view: View:** A nézet, amihez tartozik a keret. Értesíteni kell bizonyos események bekövetkeztekor.

- **Metódusok**

- **+run():** Inicializálja az ablakot.
- **+getThread(): Thread:** Visszaadja a szálat, ami a framet futtatja.

11.3.4 GameFrame

- **Felelősség**

A Frame osztályból leszármazó osztály. A játék felületét megjelenítő ablakot valósítja meg.

- **Ősosztályok**

JFrame → Frame → GameFrame

- **Attribútumok**

- **+run():** Felülírja az örökolt metódust, de ugyanaz a feladata.

11.3.5 Icons

- **Felelősség**

Enumeráció, ami a különböző játékbeli elemek megjelenítéséhez szükséges ikonokat sorolja fel.

- **Attribútumok**

- **+PANDA:** A panda ikonja
- **+ORANGUTAN:** Az orangután ikonja
- **+GAMBLINGMACHINE:** A játékgép ikonja
- **+CHOCOLATEMACHINE:** A csokiautomata ikonja
- **+SOFA:** A fotel ikonja
- **+WARDROBE:** A szekrény ikonja
- **+ENTRANCE:** A bejárat ikonja
- **+EXIT:** A kijárat ikonja

11.3.6 LeaderBoardFrame

- **Felelősség**

A Frame osztályból leszármazó osztály. A játékosok sorrendjét írja ki a pontjaik alapján fájlba mentett dicsőséglistáról.

- **Ősosztályok**

JFrame → Frame → LeaderBoardFrame

- **Attribútumok**

- **+f: File:** Azt a filet tartalmazza, amiben a játékosok pontjai vannak tárolva.

- **Metódusok**

- **+run():** Felülírja az örökolt metódust, de ugyanaz a feladata.
- **-load():** A metódus betölti az adatokat a fileból, hogy aztán ki tudja írni az osztály

11.3.7 Main

- **Felelősség**

Elindítja a játékot.

- **Metódusok .**

- **+main(args: String[]):void :** A program elindulási pontja.

11.3.8 LoadMapFrame

- **Felelősség**

A Frame osztályból leszármazó osztály. Ebben a keretben lehet kiválasztani a betöltendő térképet. Kiírja az összes lehetőséget, és azok közül lehet választani.

- **Ósosztályok**

JFrame → Frame → LoadMapFrame

- **Metódusok**

- **+run():** Felülírja az örökolt metódust, de ugyanaz a feladata.

11.3.9 View

- **Felelősség**

A játék kirajzolása. A „Game osztály állapotának” megjelenítése.

- **Ósosztályok**

Nincs

- **Attribútumok**

- **-frame: Frame:** Aktuális ablak, ami a képernyőn megjelenik.
- **-nextFrame: Options:** A megjelenítendő ablakot jelöli.
- **-icons: ImageIcon[Icons]:** A Játékhoz tartozó ikonok gyűjteménye.

- **Metódusok**

- **getIcon(iconName: Icons): ImageIcon:** Paraméterként kapott ikon azonosítóhoz tartozó ikon képet adja vissza.
- **+setNextFrame(option: Options):** A paraméterként kapott enumhoz tartozó ablakot állítja be megjelenítendő ablakként.
- **+start(): void:** Elindítja a felületet, megjelenik a Főmenü Ablak (MainMenuFrame).
- **+clickedAt(x: int, y: int): void :** Megkapja egy kattintás koordinátáit, és továbbadja a kontrollernek a kattintott mezőt.
- **+drawAll(): void:** A játék aktuális állapotát rajzolja ki.
- **+getField(x: int, y:int): Field :** A játék pályának x,y koordináta által leírt Field objektumát adja vissza.

11.3.10 Vertex

- **Felelősség**

Az ablak x,y koordinátáját írja le.

- **Ősosztályok Nincs**

- **Attribútumok**

- **+x: int:** Az ablak x-koordinátája
- **+y: int:** Az ablak y-koordinátája

11.3.11 MouseEventHandler

- **Felelősség**

Az egér műveleteket kezeli, amelyek a GameFrame-n történnek,

- **Ősosztályok**

- **MouseEventHandler → MouseListener interface**

- **Metódusok**

- **+MouseClicked(e: MouseEvent): void :** A paraméterként kapott eseményt kezeli.

11.3.12 MainMenuFrame

- **Felelősség**

A játék indításakor megjelenítendő ablak.

- **Ősosztályok MainMenuFrame → Frame → JFrame**

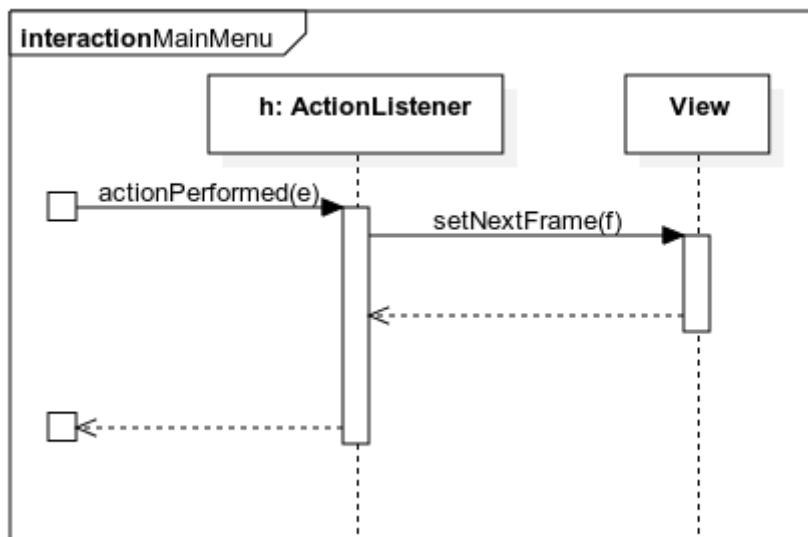
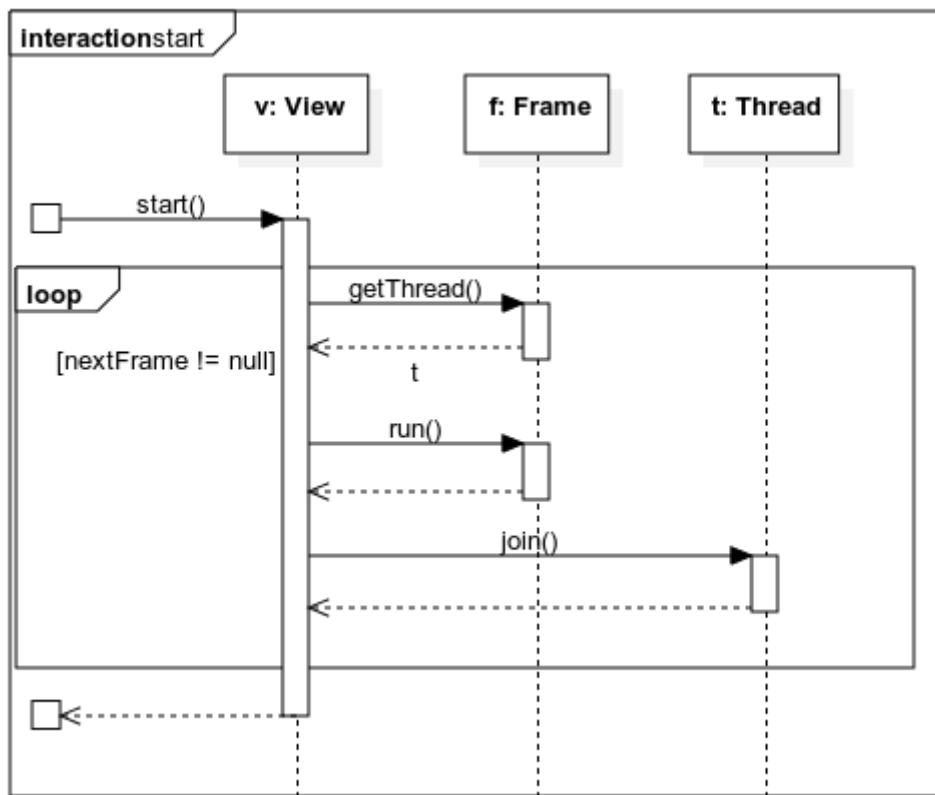
- **Attribútumok**

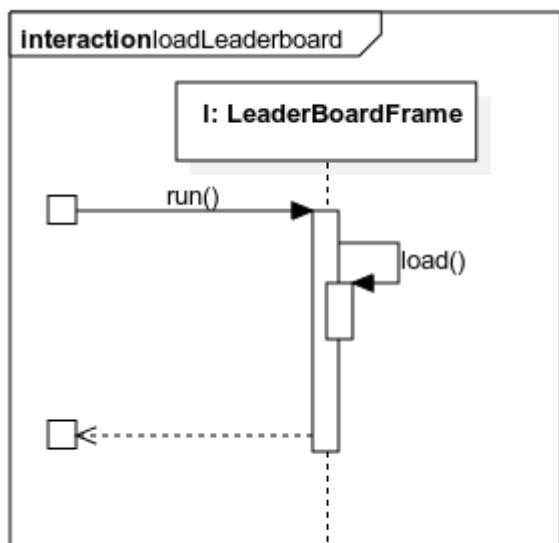
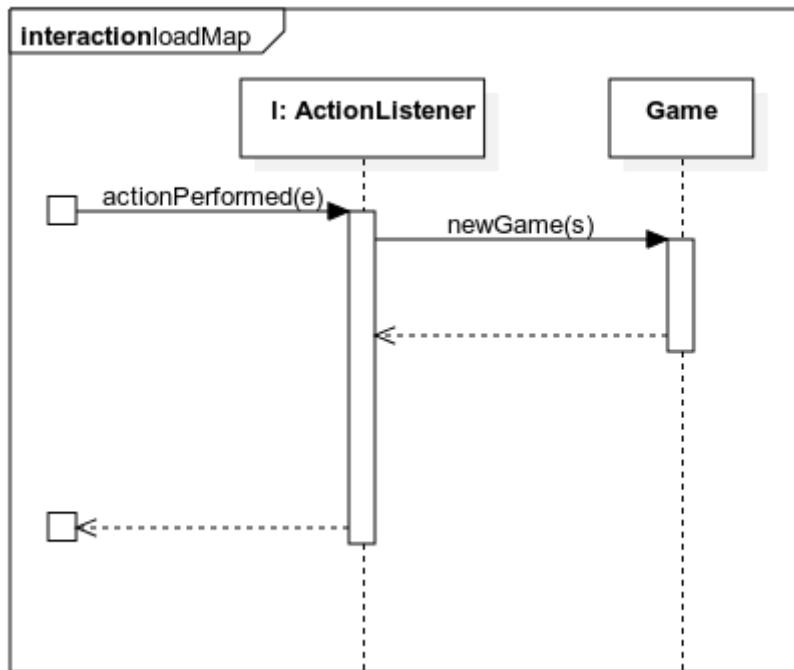
- **-jlabel[]: JLabel:** Az ablakon található címkék.
- **-jbutton[]: JButton:** Az ablakon található gombok.

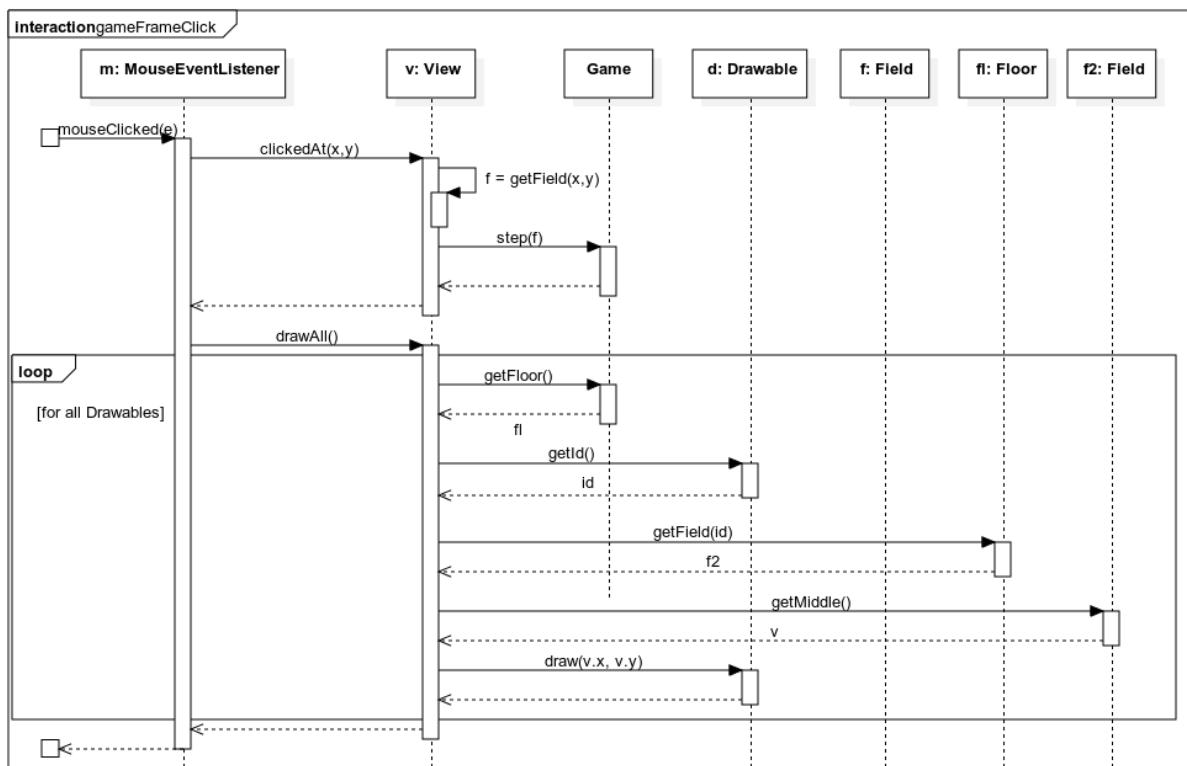
- **Metódusok .**

- **+run():void:** Felülírja az örökölt metódust, de ugyanaz a feladata.

11.4 Kapcsolat az alkalmazói rendszerrel







11.5 Napló

Kezdet	Időtartam	Résztvevők	Leírás
2019.04.28. 7:00	2 óra	Hegedüs	A játékban lévő objektumok grafikus formájának elkészítése, továbbá egy grafikus mintapálya elkészítése.
2019.04.28. 14:00	1,5 óra	Hegedüs	A screenshot-szerű képek elkészítése.
2019.04.28. 19:30	0,5 óra	Hegedüs	A 11.1 szakasz szövegeinek megírása, táblázatok elkészítése, a szakasz formázása.
2019.04.29. 12:30	1 óra	Borsodi	A struktúradiagram első felének megtervezése (11.2.2 szakasz).
2019.04.29. 12:30	1 óra	Hegedüs	A struktúradiagram második felének megtervezése (11.2.2 szakasz).
2019.04.29. 18:00	1 óra	Hegedüs	A struktúradiagram javítása, kiegészítése.
2019.04.29. 19:00	2 óra	Borsodi	Szekvenciadiagramok elkészítése, a struktúradiagram kiegészítése.
2019.04.29. 21:00	2 óra	Bódi	Osztályok feléhez tartozó leírások elkészítése
2019.04.29. 23:00	2 óra	Nyári	Osztályok leírásának elkészítése

13. Grafikus változat beadása

13.1 Fordítási és futtatási útmutató

13.1.1 businesslogic

Fájl neve	Méret	Keletkezés ideje	Tartalom
Activateable.java	1KB	2019.05.09. 18:27	Activateable osztály
ChocolateMachine.java	1KB	2019.04.23. 22:35	ChocolateMachine osztály
CowardPanda.java	1KB	2019.04.23. 22:36	CowardPanda osztály
Exit.java	2KB	2019.05.09. 21:03	Exit osztály
Field.java	7KB	2019.05.12. 8:42	Field osztály
FieldElement.java	2KB	2019.05.10. 13:49	FieldElement osztály
Floor.java	4KB	2019.05.12. 23:59	Floor osztály
GamblingMachine.java	1KB	2019.04.23. 22:35	GamblingMachine osztály
Game.java	9KB	2019.05.13. 0:08	Game osztály
JumpyPanda.java	1KB	2019.04.22. 20:20	JumpyPanda osztály
MapParser.java	16KB	2019.05.12. 17:24	Pálya fájl értelmező
Orangutan.java	5KB	2019.05.12. 23:59	Orangutan osztály
Panda.java	5KB	2019.05.11. 20:44	Panda osztály
SleepyPanda.java	2KB	2019.05.10. 22:46	SleepyPanda osztály
Sofa.java	2KB	2019.05.10. 22:46	Sofa osztály
Steppable.java	2KB	2019.05.10. 13:56	Steppable osztály
Wardrobe.java	5KB	2019.05.12. 23:59	Wardrobe osztály

13.1.2 Graphics

Fájl neve	Méret	Keletkezés ideje	Tartalom
ButtonListener.java	1KB	2019.05.12. 23:36	ButtonListener osztály
Drawable.java	1KB	2019.05.12. 23:36	Drawable osztály
FieldView.java	4KB	2019.05.12. 23:36	FieldView osztály
Frame.java	2KB	2019.05.12. 23:36	Frame osztály
GameCanvas.java	1KB	2019.05.12. 23:36	GameCanvas osztály
GameEndDialog.java	2KB	2019.05.12. 23:36	GameEndDialog osztály
GameFrame.java	6KB	2019.05.12. 23:59	GameFrame osztály
Icons.java	1KB	2019.05.12. 23:36	Icons enum osztály
LeaderBoardFrame.java	2KB	2019.05.12. 23:59	LeaderBoardFrame osztály
LoadMapFrame.java	3KB	2019.05.12. 23:36	LoadMapFrame osztály
MainMenuFrame.java	4KB	2019.05.13. 0:08	MainMenuFrame osztály
MouseEventHandler.java	1KB	2019.05.12. 23:36	MouseEventHandler osztály
Options.java	1KB	2019.05.12. 23:36	Options enum osztály
Vertex.java	1KB	2019.05.09. 21:03	Vertex osztály
View.java	8KB	2019.05.12. 23:59	View osztály

13.1.3 images

Fájl neve	Méret	Keletkezés ideje	Tartalom

chocolate_machine.png	486KB	2019.05.08. 8:09	Csoki automata ikon
entrance.png	24KB	2019.05.08. 8:09	Bejárat ikon
exit.png	1896KB	2019.05.08. 8:09	Kijárat ikon
gambling_machine.png	166KB	2019.05.08. 8:09	Játék gép ikon
icon.png	40KB	2019.05.11. 10:49	Menüöz tartozó ikon
logo.png	26KB	2019.05.11. 10:52	Menüöz tartozó logo ikon
orangutan.png	26KB	2019.05.08. 8:09	Orangután ikon
panda.png	174KB	2019.05.10. 8:35	Panda ikon
sofa.png	4KB	2019.05.08. 8:09	Fotel ikon
sofapanda.png	326KB	2019.05.10. 8:52	Fotelben ülő Panda ikon
wardrobe.png	20KB	2019.05.08. 8:09	Szekrény ikon

13.1.4 root

Fájl neve	Méret	Keletkezés ideje	Tartalom
Main.java	1KB	2019.05.07. 21:12	Main osztály
firstmap.map	21KB	2019.05.12. 20:19	Egyéni pálya fájl
Sample_Map.map	15KB	2019.05.12. 22:41	Megrendelő által előírt pálya fájl
leaderboard.txt	1KB	2019.05.13. 0:08	Ranglista adatai
pandastep.map	2KB	2019.05.12. 22:53	kis méretű tesztpálya

13.1.5 Fordítás és telepítés

1. lépés: A letöltött tömörített állomány (zip fájl) kibontása egy tetszőleges helyre.
2. lépés: Windows parancssor megnyitása. (**Win+R** billentyű megnyomása, majd a **cmd** parancs beírása, végül az **ENTER** billentyű leütése.)
3. lépés: Navigálás a kibontás helyére, az src mappába a parancssorral. (A **cd <mappa_neve>** parancs használatával lehet a mappát váltani, a **cd..** parancs segítségével lehet a könyvtárhierarchiában egy szinttel feljebb lépni, a **dir** parancccsal pedig az aktuális helyen lévő fájlok és könyvtárak írathatók ki.)
4. lépés: A következő parancs futtatása: **javac -d bin Main.java businesslogic/*.java Graphics/*.java**

13.1.6 Futtatás

1. lépés: Navigálás a kibontás helyére a parancssorral. (A **cd <mappa_neve>** parancs használatával lehet a mappát váltani, a **cd..** parancs segítségével lehet a könyvtárhierarchiában egy szinttel feljebb lépni, a **dir** parancccsal pedig az aktuális helyen lévő fájlok és könyvtárak írathatók ki.) A fordítás után az éppen aktuális hely ez lesz a parancssorban.
2. lépés: A következő parancs kiadása: **cd bin**
3. lépés: A következő parancs futtatása: **java Main**

13.2 Értékelés

Tag neve	Tag neptun	Munka százalékban
Borsodi Regő	HH5GMS	32,26
Bódi Dániel	KM3B5S	23,04
Hegedüs Amadé	P53ZLW	24,65
Nyári Tamás	X49AI8	20,05

13.3 Napló

Kezdet	Időtartam	Résznevők	Leírás
2019.05.07. 17:00	3 óra	Borsodi	Alap grafikus megjelenítés, grafikus osztályok
2019.05.10. 18:00	3 óra	Borsodi	pálya megjelenítése, javítások a modellben.
2019.05.11. 15:00	4,5 óra	Bódi	Saját pálya lemodellezése + a hozzá tartozó map fájl megírása
2019.05.12. 18:00	2 óra	Bódi	A kiadott pálya lemodellezése
2019.05.12. 21:00	2 óra	Borsodi	A kiadott pálya pályafájlba írása, GUI funkciók megvalósítása
2019.05.12. 19:00	3 óra	Nyári	Leaderboard ablak, és fájlműveletek
2019.05.13. 0:30	0,5 óra	Bódi	Dokumentum 13.1.1, 13.1.2, 13.1.3, 13.1.4 részének elkészítése
2019.05.13. 8:00	1 óra	Nyári	Dokumentum 13.1.5, 13.1.6 részének elkészítése
2019.05.13. 8:30	1 óra	Nyári	Dokumentum 13.2 részének elkészítése
2019.05.13. 10:00	1 óra	Hegedüs	A fordítást és futtatást végző batch fájlt elkészítése.

14. Összefoglalás

14.1 A projektre fordított összes munkaidő

Tag neve	Munkaidő (óra)
Borsodi	51,5
Bódi	39,5
Hegedüs	42,5
Nyári	36
Összesen	169,5

14.2 •A feltöltött programok forrássorainak száma

Fázis	Kódsorok száma
Szkeleton	1425
Prototípus	1648
Grafikus változat	2183
Összesen	5256

Megjegyzés: A forrássorok számoltatásánál az üres sorokat nem vettük figyelembe.

14.3•Projekt összegzés

14.3.1 Mit tanultak a projektből konkrétan és általában?

Egy szoftver fejlesztésének teljes folyamatával megismerkedtünk, ami magába foglalta a modellezésen túl a tesztelést és a dokumentálást is, továbbá megtanultunk csapatban dolgozni. Jobban megismerkedtünk a Java API néhány részével, és a Java nyelv erősségeivel, és gyengeségeivel. Megtanultuk használni a GIT-et és rájöttünk arra, hogy Windowsos környezetben batch fájlok készítésével jelentősen meggyorsíthatjuk egy program fordítását és futtatását. Megtanultuk beosztani az időket és felismertük, hogy a határidőknek súlya van.

14.3.2 Mi volt a legnehezebb és a legkönnyebb?

A legnehezebb a grafikus felület specifikációjának az elkészítése volt. Sajnos Programozás alapjai 3-ból nem mentünk bele a grafikus felületek részleteibe, bár az megfelelő alapot adott ahhoz, hogy tudásunkat saját magunk továbbfejlesszük. A legkönnyebb a skeleton elkészítése volt, hiszen az azt megelőző modellt a konzulens több alkalommal is átnézte, ezért mire kódolásra került sor, addigra már a hibák nagy része javítva volt. Munkánkat jelentősen megkönnyítette az, hogy volt egy hozzáértő személy, aki heti szinten értékelte és javította az általunk készített dokumentációt és kódot.

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

Többségében igen.

14.3.4 Ha nem, akkor hol okozott ez nehézséget?

A grafikus felület specifikációjánál kevés idő állt rendelkezésre a részletekbe menő tervezésre.

14.3.5 Milyen változtatási javaslatuk van?

Az anyagokat lehetne elektronikus formában leadni, hogy ne kelljen a papírmunkával foglalkozni.

14.3.6 Milyen feladatot ajánlanának a projektre?

Jó ötletnek tartanánk, ha az előző félévben Szoftvertechnológiából kiadott házi feladatot kapná minden csoport ebből a tárgyból, így gyakorlatilag a korábban félre elkészített megoldásokat tovább lehetne fejleszteni és az ott elkészített modellt működés közben is látná a hallgatók. A tárgy egyébként összeköthető a Számítógépes grafika nevű tantárggyal, a grafikus felület elkészíthető lenne pluszfeladat formájában a feladathoz.

14.3.7 Egyéb kritika és javaslat

Az értékelések során sajnos számtalan szor találkoztunk olyan esettel, hogy egy másik csapat, akik sokkal kevesebb időt, energiát és tudást fektettek a munkájukba, sokkal magasabb pontszámot kaptak azért, mert a konzulensük nem nézte át megfelelő mértékben a leadott munkát. A grafikus felület beadásakor erre természetesen fény derült, mert rengeteg hibát kellett javítaniuk az utolsó utáni pillanatban, azonban az mégis rosszul értékelte minket, hogy nagyon jó pontszámokat szereztek a hibás munkáikkal.