

Continuity projekt

38 - mazli

Konzulens:

Dr. László Zoltán

Csapattagok:

Szabó Csaba	BABFRK	sz.csabi.3@gmail.com
Halász Antal István	F6ID7E	halasz.antal.istvan@gmail.com
Kiss Beáta	GT2NUN	lotrisz@gmail.com
Gerlei Balázs	NIEIIB	gbalazs6@gmail.com
Berki Ramóna	G8FPAJ	b.ramcsi@gmail.com

2012. május 9.

2. Követelmény, projekt, funkcionalitás

2.1 Követelmény definíció

A program célja, alapvető feladata: a Continuity játék alapján egy grafikus játék elkészítése csapatmunkában egy egyetemi tantárgy (Szoftver laboratórium 4) keretein belül.

Rendszerkövetelmények:

- a program futtatásához szükséges környezet:
 - a program Java nyelven készül el, ezért Java 1.5 VM-nal rendelkező számítógép szükséges a fordításához
 - IB407-es teremben lévő számítógépeken fordíthatónak és futtathatónak kell lennie a programnak
- a rendszernek lehetővé kell tennie azt, hogy a program új fájlt hozzon létre és hogy ez ne vesszen el
- funkcionális követelmények:
 - a programnak a bemeneti adatokra (billentyűzet leütése) azonnal reagálnia kell
- felhasználói felület:
 - grafikus felület
 - menüvel rendelkezik a játék
 - billentyűzetről irányítható
- a szoftver elkészítésének a menete időrendben
 - analitikus modell
 - szkeleton (programváz)
 - prototípus
 - grafikus felület
- szoftver fejlesztésével kapcsolatos alapkövetelmények
 - szoftverfejlesztési napló, melyben személy szerint követhető minden elvégzett feladat és a munkaórák száma
 - programok verziószám nyilvántartása és kezelése
 - major, minor, build és revision number használata
 - csapatmunkában történő fejlesztés, dokumentálás
 - hétről-hétre pontos határidőre elkészített feladatok leadása (dokumentáció, programkód)
- szoftver fejlesztésével kapcsolatos célok
 - csapattagok egyéni fejlődése dokumentálás, tervezés, tesztelés és implementálás terén
 - csapatban egy közös projektben való munka megismerése

Felhasználói követelmények:

- a felhasználónak egy grafikus felületű játékot kell irányítania
- a felhasználó számítógépén telepítve kell, hogy legyen a Java Runtime Environment 1.5, így lehet elindítani a játékot
- billentyűzetről lehet irányítani a játékot, így billentyűzet szükséges a játékhoz
- előzetes információ vagy tudás nem elvárt a felhasználó részéről, mivel ez egy interaktív, könnyen megtanulható játék

2.2 Projekt terv

Felhasznált fejlesztőeszközök:

- Eclipse Indigo, Papyrus (fejlesztés, tesztelés)
- Oracle JDeveloper és OpenAmeos (tervezés, fejlesztés)
- Microsoft Word, Google Docs (.doc, docx formátum)
- Microsoft Excel, Google Docs (.xls formátum)
- GIT Google Code tárhellyel (fejlesztés)
- Javadoc (dokumentáció)
- JUnit (tesztelés)
- Dropbox (közös fájl elérés)

Fejlesztői csoport tagjai, azok feladatai:

- csapatvezető: Szabó Csaba
 - értekezletek lebonyolítása, előkészület értekezletre
 - projekt felügyelete
 - tervezés
 - munkafelügyelet
 - konzultációk látogatása
- Berki Ramóna
 - tervezés
 - fejlesztési napló összeállítása
 - értekezlet dokumentálása
- Gerlei Balázs
 - implementálás
 - tesztelés
 - feladatbeadás
- Halász Antal István
 - implementálás
 - tesztelés
- Kiss Beáta
 - fejlesztési napló összeállítása
 - értekezlet dokumentálása
 - tervezés

Aki az aktuális részfeladaton nem dolgozik, az ellenőrzi a többiek munkáját, ellenőrzi a projekt logikai helyességét.

Kommunikáció a csapattagok között:

- legalább heti rendszerességgű, 5 fős gyűlés
- skype értekezletek
- google eszközök (pl.: gtalk), email, telefon

Fejlesztési mérföldkövek:

3 fő lépcső, ezeket a feladatokat a Hecules feladatbeadó rendszeren kell beadni.

- szkeleton
 - feltöltés legkorábbi időpontja: március 12. 18:00
 - feltöltés határideje: március 19. 14:00
 - feltöltés végső határideje: május 18. 14:00
- prototípus

- feltöltés legkorábbi időpontja: április 10. 18:00
- feltöltés határideje: április 16. 14:00
- feltöltés végső határideje: május 18. 14:00
- grafikus felület
 - feltöltés legkorábbi időpontja: május 2. 18:00
 - feltöltés határideje: május 7. 14:00
 - feltöltés végső határideje: május 18. 14:00

Ütemterv: heti rendszerességű feladatbeadás és konzultáció

- február 8. – Csapatok regisztrációja
- február 15. – Követelmények, projekt, funkcionalitás + konzultáció
- február 22. – Analízis modell kidolgozása 1. + konzultáció
- február 29. – Analízis modell kidolgozása 2. + konzultáció
- március 7. – Szkeleton tervezése + konzultáció
- március 14. – Szkeleton beadása + konzultáció (szkeleton bemutató)
- március 21. – Prototípus koncepciója + konzultáció
- március 28. – Részletes tervek + konzultáció
- április 4. – Prototípus készítése, tesztelése + konzultáció
- április 11. – Prototípus beadása + konzultáció (prototípus bemutató)
- április 18. – Grafikus felület specifikálása + konzultáció
- április 25. – Grafikus változat készítése + konzultáció
- május 2. – Grafikus változás beadása + konzultáció
- május 9. – Összefoglalás + konzultáció (grafikus bemutató)

Határidők:

- A nyomtatott anyagokat hétfőnként 14:00 és 14:15 között az I épület földszintjén kell átadni az IIT munkatársának
- Késedelem esetén az anyagot közvetlenül a konzulensnek kell átadni vagy leadható az IIT oktatási adminisztrációjában is (IB316)
- Pontos határidők:
 - február 10. – csapatok regisztrációja
 - február 20. – Követelmény, projekt, funkcionalitás (beadás)
 - február 27. – Analízis modell kidolgozás 1. (beadás)
 - március 5. – Analízis modell kidolgozása 2 (beadás)
 - március 12. – Szkeleton tervezése (beadás)
 - március 19. – Szkeleton (beadás)
 - március 26. – Prototípus koncepciója (beadás)
 - április 2. – Részletes tervek (beadás)
 - április 16. – Prototípus (beadás)
 - április 23. – Grafikus felület specifikációja (beadás)
 - május 7. – Grafikus változás (beadás)
 - május 11. – Összefoglalás (beadás)

Kockázatelemzés:

- IL407-es gépen nem fordul a kód, megoldások:
 - figyelni kell rá, többször beadás előtt tesztelni a laborban
 - jelen kell lennie beadáskor a kód készítőjének
- csapaton belül konfliktusok, megoldások:
 - csapatvezető feladata ezeket a konfliktusokat feloldani
- egészségügyi állapotok, megoldások:
 - csapatvezető kijelöl mást a csapattól aki megoldja a beteg személy feladatát

- elromlik egy csapattag számítógépe, megoldások:
 - csapattagok segítenek megoldani, például számítógép kölcsönadásával
 - HSZK gépeinek használata
 - IB407 labor gépeinek használata

2.3 Feladateleírás

Bevezető:

A megrendelő egy olyan játékprogramot kér, amelynek az IL407-es labor gépein kell futnia. Ezekben a gépeken rendelkezésre áll a Java JRE 1.5-ös változata, a rendszerkövetelmények ennek megfelelőek. Erre kell tehát fejleszteni.

Pontos feladat leírás helyett kaptunk egy hasonló, Continuity nevű játékot, amely alapján kell elkészítenünk a miénket. Az eredeti a <http://continuitygame.com/playcontinuity.html> URL címen megtalálható. A konkrét megvalósításban többé-kevésbé eltérhetünk, saját ötleteket belevihetünk.

Rövid leírás a játékról:

A program egy grafikus felületű, logikai/ügyességi játék. Van egy bábu, amelyet több különböző szinten végig kell vinni. A játékot az teszi egyedivé, hogy a felhasználó nem csak a bábút irányíthatja, hanem a program felületét is; vagyis nem egy fix pálya van, melyen lehet mozogni, hanem több kisebb csempére van felosztva és ezeknek a csempéknek a elhelyezését tudja változtatni a felhasználó. Ezáltal egy-egy szinten nem csak egyféleképp lehet eljutni el a kijáratig illetve a kijáratig, csak a felhasználón múlik, hogyan át a pályát.

A játék célja:

A játék célja a szórakoztatás és a logika fejlesztése. Konkrétabban: az összes szinten sikeresen végigmenni. Ehhez van egy bábu, amely a játékost reprezentálja, amivel az átrendezhető pályán fel kell szedni az elhelyezett kulcsot (vagy kulcsokat) és el kell jutni a kijáratig. Ehhez gyakran a pálya többszöri átrendezése van szükség. Ha a bábuval elérjük a kijáratot, a következő szintre lépünk.

A játék főbb elemei:

Csempe: A pálya csempékből áll, ezek a mozgatható részei. Egy csempében szerepelhet terep, levegő, kulcs, kijárat, bábu.

Az ablakban $x \cdot y$ darab csempe van, ahol az x és y egynél nagyobb, és nem feltétlenül egyenlő egész szám. A méretük állandó és darabszámtól függetlenül mindig azonos, akár $4 \cdot 4$ darab van, akár $10 \cdot 10$. A csempéket a billentyűzet gombjai segítségével tudjuk mozgatni függőleges és vízszintes irányban az üres helyekre.

Minden csempének van kerete. A bábu csak akkor mehet át két csempe között, ha az a két csempe illeszkedik. Ez az átjárás történhet vízszintes és függőleges irányban is. Az üres helyre egy alatta, fölötte vagy mellette lévő csempét tehetünk.

A bábu csak a csempéken belül tud mozogni. A csempében van minden, ami nem terep, kulcs, kijárat és bábu, az levegő. Ez tölti ki a teret. A csempéket úgy kell elrendezni a játék során, hogy a bábu meg tudja szerezni a kulcsokat, majd el tudjon jutni a kijáratig.

A csempék irányítása a billentyűzetről történik.

Bábu: A bábu lényegi része a játéknak, ez reprezentálja a játékost. A bábu csak csempékben belül tud mozogni, csak a terepen állhat, a levegőben és a csempe szélén nem, viszont mozogni csak a levegőben mint közegben tud. Irányítani csak belső nézetben lehet. A bábu meghal, ha ráesik valamelyik csempe keretére, ami alatt nincs másik csempe. Ilyenkor visszatér a kiindulási pontjára. Terep a levegőben is lehet, ezekre rá tud ugrani a bábu. A bábu nem lóghat ki a csempéből, teljes egészében a csempén belül kell lennie: vagy a tereppel érintkezve, vagy ugrás közben a levegőben. Ha eléri a csempe valamelyik oldalsó szélét, akkor megakad és vissza kell fordulni. A függőlegesen előtte álló terepre fel tud ugrani, ha az

nem haladja meg a maximális ugrási magasságát. Ha nem sikerül felugrania, akkor is vissza kell fordulnia. A bábu vezérlése a billentyűzetről történik.

Kulcs, kijárat:

A kulcs egy tárgy, melyet a felhasználónak meg kell szereznie a bábuval, hogy a következő szintre léphessen. A kulcsot nem tudjuk mozgatni, viszont ha a bábu egyszer eléri, akkor nála marad addig, amíg a kijáratot eléri, nem tudja letenni.

A kijárat elérésével teljesíthetjük a szintet. Ha úgy érjük el a kijáratot, hogy nincsenek nálunk a szükséges kulcsok, akkor semmi sem történik.

Külső nézetben úgy kell elrendezni a csempéket, hogy utat tudjunk a bábunak biztosítani a kulcshoz, ezután pedig a kijárhoz. Ezt többszöri váltást eredményezhet a külső és a belső nézetek között.

Szintek:

A játékban több szint van és közöttük meghatározott a sorrend. Ha a legelsőt teljesítettük, akkor lépünk tovább a másodikra, majd a harmadikra és így tovább, amíg végig nem játszottuk az összes szintet.

Az aktuális szintet (ahol vagyunk játék közben) el tudjuk menteni, így a játék legközelebb annak a szintnek az elejéről folytatódhat, de akár újra is kezdhető az elejéről. A program tehát csak egy mentést képes kezelni. Ha már van egy mentett szintünk és valaki elkezd egy új játékot, majd elmenti, ahol tart, akkor az első mentés felülíródik.

A játék indítása után választhatunk, hogy új játékot szeretnénk kezdeni, vagy betöltjük a mentettet.

2.4 Szótár

játekos: A program felhasználója, aki játszik.

bábu: A játékoszt reprezentáló objektum, képes ugrani, futni, zuhanni, a kulcsot megszerezni, elérni a kijáratot.

ugrik: A bábu függőlegesen felfelé elmozdul, ugrás közben vízszintesen is irányítható. Ha ugrással elérte az ugrás tetőpontját és nincs közvetlenül alatta terep, akkor elkezd zuhanni.

fut: A bábu vízszintes mozogása a terepen. Ha futás közben lefut a terepről, akkor zuhan.

elakad: Elakadni a falban, keret felső vagy oldalsó részében lehet. Ilyenkor a bábu abban az irányban nem tud tovább mozogni (ugrani vagy futni).

zuhanás: Függőleges lefelé irányú mozgás, zuhanni csak levegőben tud a bábu. Zuhanás közben vízszintesen is irányítható.

megszerez: A bábu eléri a kulcsot a szint teljesítése érdekében.

kiindulási pont: Itt jelenik meg a szint elején a bábu, ide kerül vissza halál után is.

halál: A bábu hozzáér a csempe alsó keretéhez, ilyenkor visszatér a kiindulási pontra, (ha a keret fölött közvetlenül terep van, a bábu nem tudja elérni, így nem hal meg).

szint: Logikai fejtörő, ilyen szinteket kell teljesíteni a játékban.

pálya: A szintek grafikus interpretációja, csempékre van darabolva.

csempe: Az egy pályán lévő alakzatok összessége, ezen belül ugrik, fut, zuhan és szerzi meg a kulcsot a bábu, a csempék képesek illeszkedni, a bábu képes átmenni csempék között, a csempék tudnak függőleges és vízszintes irányban mozogni, a csempék sor-oszlop elrendezésűek, a számozás balról jobbra, fentről lefelé megy, tehát a bal felső sarokban a 0-ás számozással kezdődik.

levegő: A csempe része, ebben a közegben tud mozogni a bábu.

terep: A pálya azon része, amibe nem lóghat bele a bábu.

keret: A csempét, mint geometriai alakzatot veszi körül, tereppel és a levegővel szomszédos, azok nem lóghatnak bele.

alakzat: Bábu, terep, levegő és minden egyéb tárgy ami a csempén belül és kívül van.

kulcs: Olyan alakzat, amit meg kell szerezni a pálya teljesítéséhez, minden szinten legalább egy van belőle, de több is lehet.

kijárat: Ezt az alakzatot kell elérni a játékosnak a szint teljesítéséhez.

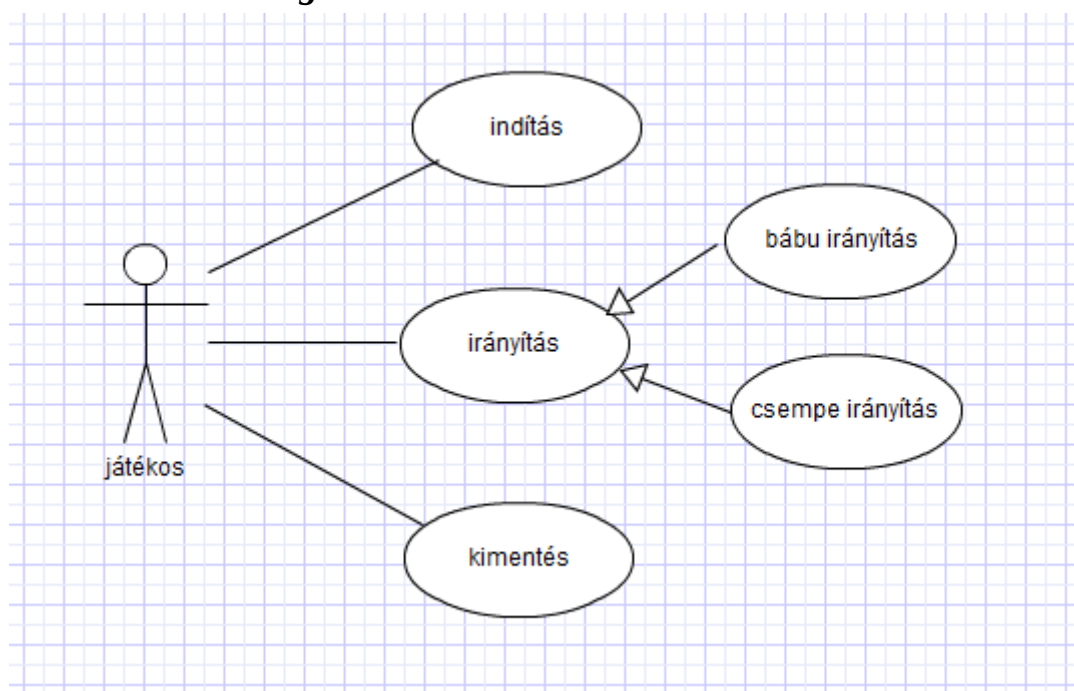
illeszkedés: Két csempe szomszédos, a kereten belüli szélein terep és a levegő mintázata pontosan megegyezik.

átmegy: Ha két csempe pontosan illeszkedik azon az oldalán, ahol át akar menni egyikből a másikba a bábu, akkor át tud menni.

szint teljesítése: Egy szintet akkor teljesít a játékos, ha a kulcsot (vagy kulcsokat) megszerezte a bábuval és elérte a kijáratot. Ekkor megjelenik egy gratuláló képernyő, majd betöltődik a következő szint.

2.5 Essential use-case-ek

2.5.1 Use-case diagram



2.5.2 Use-case leírások

Use-case neve	Indítás
Rövid leírás	A játékos kiválasztja a menüben, hogy mentett, vagy új játékot szeretne-e kezdeni, a rendszer betölti a megfelelő szintet.
Aktorok	Játékos
Forgatókönyv	meghívja a megfelelő függvényeket, betölti a szintet
Use-case neve	Kimentés
Rövid leírás	Kimenti a játék aktuális állapotát visszatölthető formában
Aktorok	Játékos
Forgatókönyv	kimenti a játék állását
Use-case neve	Irányítás
Rövid leírás	A játékos mozgatja a rendszer egyes elemeit
Aktorok	Játékos
Forgatókönyv	Kiszámolja a rendszer az irányított elem új koordinátáját.
Kapcsolatok	belőle származik a bábu és a csempék irányítása

Use-case neve	Bábu irányítás
Rövid leírás	A játékos mozgatja a bábút
Aktorok	Játékos
Forgatókönyv	Kiszámolja a rendszer az irányított bábu új koordinátáját.
Use-case neve	Csempe irányítás
Rövid leírás	A játékos mozgatja a csempéket
Aktorok	Játékos
Forgatókönyv	Kiszámolja a rendszer az irányított csempe új koordinátáját.

3. Analízis modell kidolgozása

3.1 Objektum katalógus

3.1.1. *Application*

Ezt hozza létre először a modellek közül a Controller, felelős a szintek betöltéséért, a játékállás mentéséért, egy korábban mentett állás betöltéséért, a program futásának befejezéséért. Mozgásra utasítja a csempéket vagy a figurát.

3.1.2. *Controller*

Ez jön létre először a program indításakor. Felelős az időkezelésért és a csempe vagy figura mozgásáért.

3.1.3. *EmptyTile*

Olyan mozgásra képes csempe, mely nem tartalmaz terepet. Benne bábu nem lehet.

3.1.4. *Exit*

Ezt kell elérni a megfelelő kulcsokkal a birtokunkban a szint teljesítéséhez. Tudja, hogy hány kulcsra van szüksége.

3.1.5. *Field*

A terepet reprezentálja. Meg tudja mondani azt, hogy egy adott pontban van-e terep.

3.1.6. *Figure*

A felhasználó által irányított bábút valósítja meg. Tárolja azt hogy ugrás közben van-e és ha igen akkor az mennyi ideig fog tartani még. Tud mozogni. Vissza tudja adni a zsebét.

3.1.7. *FullTile*

Olyan csempe, amely tartalmaz terepet és amiben a figura tud mozogni. Meg tudja kérdezni a terepet, hogy adott pontban terep van-e. Át tudja adni másnak a benne lévő terepet. Meg tudja vizsgálni, hogy illeszkedik-e egy másik csempével, az alapján, hogy a bábu honnan hova akar elmozdulni.

3.1.8. *Key*

A szint teljesítéséhez szükséges. Amikor a figura eléri a kulcsot, akkor a kulcs beteszi önmagát a figura zsebébe. Meg tudja mondani, hogy felvették-e.

3.1.9. *Level*

Ez kezeli a mozgást. Ez dönti el, hogy csempe, vagy bábu mozogjon, és milyen irányban. Ez gondoskodik a folyamatban lévő zuhanásról (gravitáció) és ugrásról. A bábu tőle kérdezi meg, hogy ha el akar mozdulni valamelyik irányban, akkor hova kerül.

3.2 Osztályok leírása

3.2.1. *Application*

- *Felelősség*

- kezelje az idő múlását
- betölt egy szintet
- kezeli a szinten lévő mozgásokat
- menti a játékállást
- kilép az alkalmazásból
- **Ősosztályok**

Object

- **Interfészek**

nincs

- **Attribútumok**

- **String[] levelsPath:** A szintek elérési útját tárolja.
- **String savedLevelPath:** Az elmentett szint elérési útja. Null, ha nincs mentett szint.
- **Level actualLevel:** Az éppen betöltött szint.

- **Metódusok**

- **void simulateWorld(int time):** Az idő múlását szimulálja, a betöltött szint gravitációt kezelő függvényét hívja, ha van betöltött szint.
- **bool loadLevel(String path):** Betölti az átadott elérési útnak megfelelő szintet. Visszaadja, hogy sikerült-e a betöltés.
- **void nextMove(Type what, Direction dir):** Ha van betöltött szint, akkor továbbítja a szintnek, hogy mi mozogjon (what paraméter) és milyen irányba (dir paraméter).
- **void save():** Külső fájlba menti az aktuális szintet és beállítja a savedLevelPath változót.
- **void exit():** Kilép a programból.

3.2.2. Controller

- **Felelősség**

- időkezelés
- mozgás kezelés

- **Ősosztályok**

Object

- **Interfészek**

még nem definiált

- **Attribútumok**

még nem definiált

- **Metódusok**

még nem definiált

3.2.3. EmptyTile

- **Felelősség**

- mozogni tud

- **Ősosztályok**

Tile

- **Interfészek**

IMovable

- **Attribútumok**

nincs

- **Metódusok**

nincs

3.2.4. Exit

- **Felelősség**
 - *meg tudja mondani, hány kulcs szükséges a szint teljesítéséhez*
 - *be tudja állítani a szükséges kulcsok számát*
 - *megvizsgálja a kapott kulcsokat*
- **Ősosztályok**
GameObject
- **Interfészek**
nincs
- **Attribútumok**
 - **int numberOfKeys:** A szint teljesítéséhez szükséges kulcsok száma.
- **Metódusok**
 - **int getNumberOfKeys():** Visszaadja a szint teljesítéséhez szükséges kulcsok számát.
 - **void setNumberOfKeys(int number):** Beállítja a szint teljesítéséhez szükséges kulcsok számát.
 - **bool interact(Key[] pocket):** Megvizsgálja a kapott zsebben lévő kulcsokat. Ha megfelelő számú van, akkor true-val tér vissza, ha nem akkor false-szal.

3.2.5. Field

- **Felelősség**
 - *megmondja, hogy az adott koordinátán terep van-e*
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - **Point[][] points:** A terepet reprezentáló pontok. A terepen egyenesek vannak. Egy egyenes két végpontját egy Point tömb tárolja index alapján sorban. Több Point tömb (egyenes) reprezentálja a terepet.
- **Metódusok**
 - **bool isThereField(Point point):** Megmondja, hogy a paraméterként kapott pontban terep van-e.

3.2.6. Figure

- **Felelősség**
 - *meg tudja mondani a koordinátáját*
 - *meg tudja mondani, hogy mennyi ideig ugrik még*
 - *át tudja adni a zsebét*
- **Ősosztályok**
Object
- **Interfészek**
IMoveable
- **Attribútumok**
 - **Point coord:** A bábu koordinátája.
 - **int jump:** Tárolja mennyi ideig van még ugrásban a bábu
 - **Level level:** Referencia arra a szintre amiben éppen van a bábu.

- **Key[] pocket:** A bábu zsebe, melyben kulcsokat tárolunk.
 - **Metódusok**
- **Point getCoord():** Visszaadja a bábu koordinátáját.
- **int getJump():** Visszaadja azt hogy mennyi ideig ugrik még a bábu.
- **Key[] getPocker:** Visszaadja a bábu zsebet.
- **void setCoord(point:Point):** Beállítja a bábu koordinátáját.

3.2.7. FullTile

- **Felelősség**
 - *illeszkedést kezel a csempék között*
 - *vizsgálja a benne lévő terepet*
- **Ősosztályok**
Tile
- **Interfészek**
nincs
- **Attribútumok**
 - **Field field:** A benne lévő terep.
 - **bool isActive:** Megmondja, hogy ebben a csempében van-e a bábu.
- **Metódusok**
 - **bool match(Point oldCoord, Point newCoord):** Illeszkedést vizsgál a csempék között. A paraméterként kapott két pontból számolja ki azt, hogy a csempék melyik oldalának illeszkedését kell vizsgálni.
 - **Field getField():** Visszaadja a csempében található terepet.

3.2.8. GameObject

- **Felelősség**
 - *ez az absztrakt osztály kezeli a bábu zsebet*
 - *megmondja, hogy a vele történő érintkezés hatására a pálya teljesített lett-e*
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - **Point coord:** Megmondja, hol helyezkedik el a GameObject.
- **Metódusok**
 - **Point getCoord():** Visszaadja a koordinátáját.
 - **void setCoord(Point point):** Beállítja a koordinátáját.
 - **bool interact (Key[] pocket):** A paraméterként kapott zsebet olvassa vagy módosítja és ezután visszaadja, hogy teljesítve van-e a pálya.

3.2.9. IMoveable

- **Felelősség**
 - *ez az interfész valósítja meg a mozgást*
- **Ősosztályok**
nincs
- **Metódusok**
 - **void move(Direction dir):** a paraméterben kapott irány alapján mozgatja az ezt az interfészt megvalósító objektumok (annak koordinátáját változtatja meg).

3.2.10. Key

- **Felelősség**
 - *tudja, hogy felvették-e*
 - *beleteszi magát a figura zsebébe*
- **Ősosztályok**
GameObject
- **Interfészek**
nincs
- **Attribútumok**
 - **bool pickedUp:** Felvették-e már.
- **Metódusok**
 - **bool getPickedUp():** Visszaadja, hogy felvették-e már.
 - **bool interact(Key[] pocket):** Ha még nem vették fel, akkor beleteszi önmagát a paraméterként kapott zsebbe. Mindig false-szal tér vissza.

3.2.11. Level

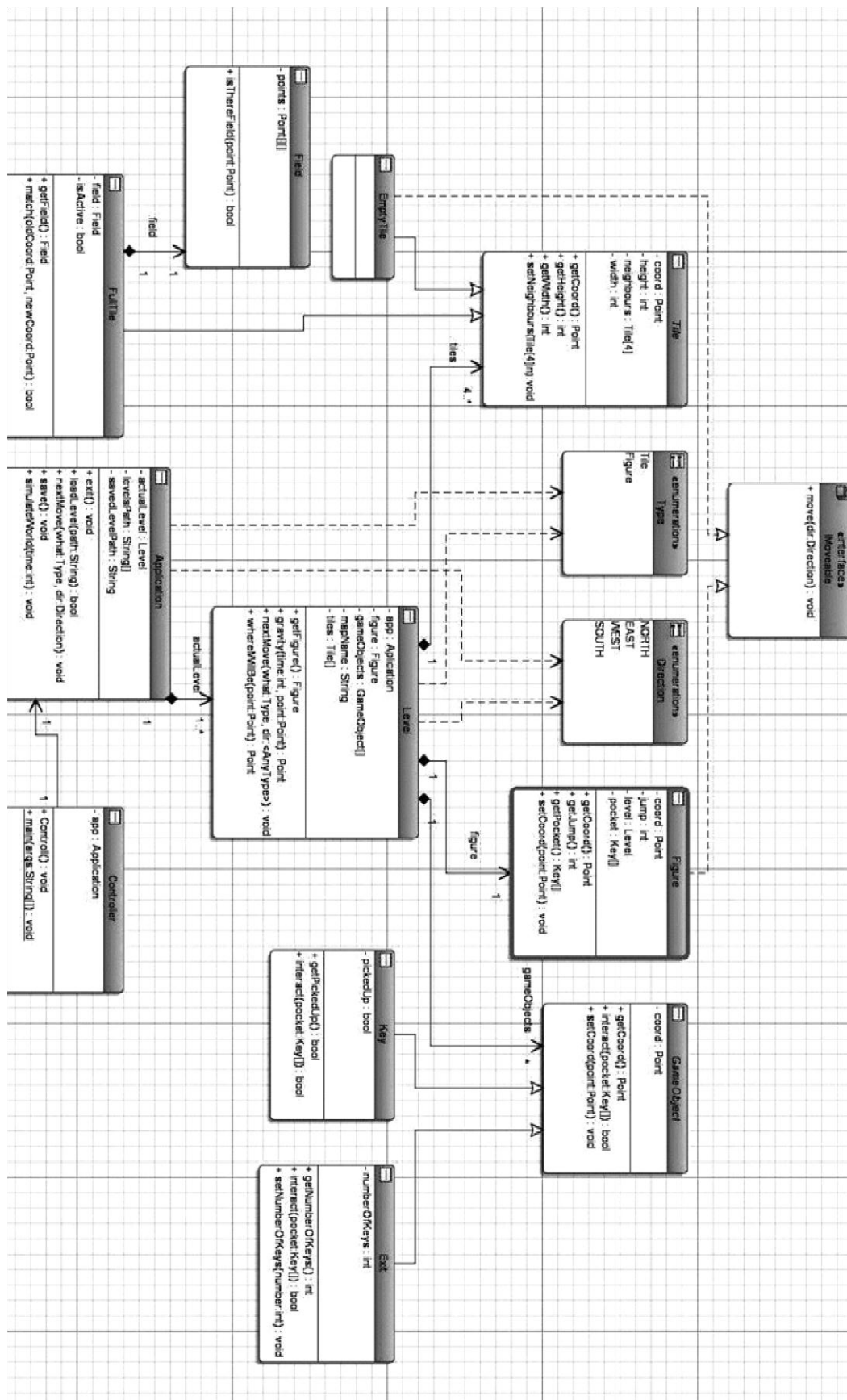
- **Felelősség**
 - *A szinten lévő elemek mozgását irányítja*
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - **Application app:** Referencia az alkalmazásra, ami létrehozta. A következő szint betöltésekor használjuk.
 - **Tile[] tiles:** A szinten lévő csempék. A 0. indexű elem az egyetlen üres csempe (EmptyTile).
 - **Figure figure:** A szinten lévő bábu.
 - **String mapName:** A szint elérési útvonalát tartalmazza.
 - **GameObjects[] gameObjects:** A szinten lévő alakzatok.
- **Metódusok**
 - **Figure getFigure():** Referenciát ad vissza a szinten található figurára.
 - **Point whereIWillBe(Point point):** A bábu tervezett pozícióját korrigálja a gravitációt és a folyamatban lévő ugrást figyelembe véve, és az újonnan kiszámított pozíciót visszaadja.
 - **void nextMove(Type what, Direction dir):** Megkapja, hogy mi mozogjon, és ennek hívja meg a mozgató metódusát, aminek átadja a kapott irányt (dir).
 - **Point gravity(int time, Point point):** A paraméterként kapott pont helyzetét korrigálja az eltelt idő alapján függőleges irányban, attól függően, hogy a bábu ugrik-e (ezt a bábutól kérdezi meg).

3.2.12. Tile

- **Felelősség**
 - *ez az absztrakt osztály nyilvántartja az aktuális szomszédjait*
- **Ősosztályok**
nincs
- **Interfészek**
nincs
- **Attribútumok**

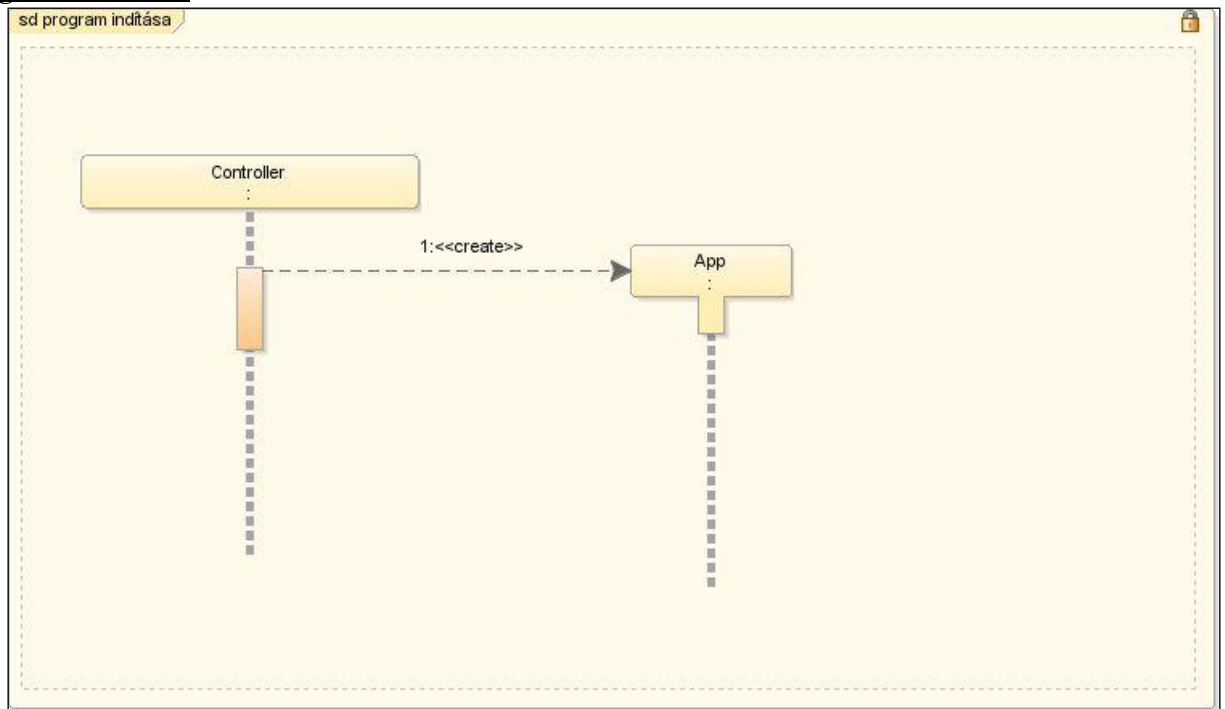
- **Point coord:** A csempe koordinátája.
- **Tile [4] neighbours:** A csempe szomszédjai.
- **int height:** A csempe magassága.
- **int width:** A csempe szélessége.
- **Metódusok**
 - **void setNeighbours(Tile [4] n):** Beállítja a csempe szomszédjait.
 - **int getHight():** Megadja a csempe magasságát.
 - **int getWidth():** Megadja a csempe szélességét.
 - **Point getCoord() :** Megadja a csempe koordinátáját.

3.3 Statikus struktúra diagramok

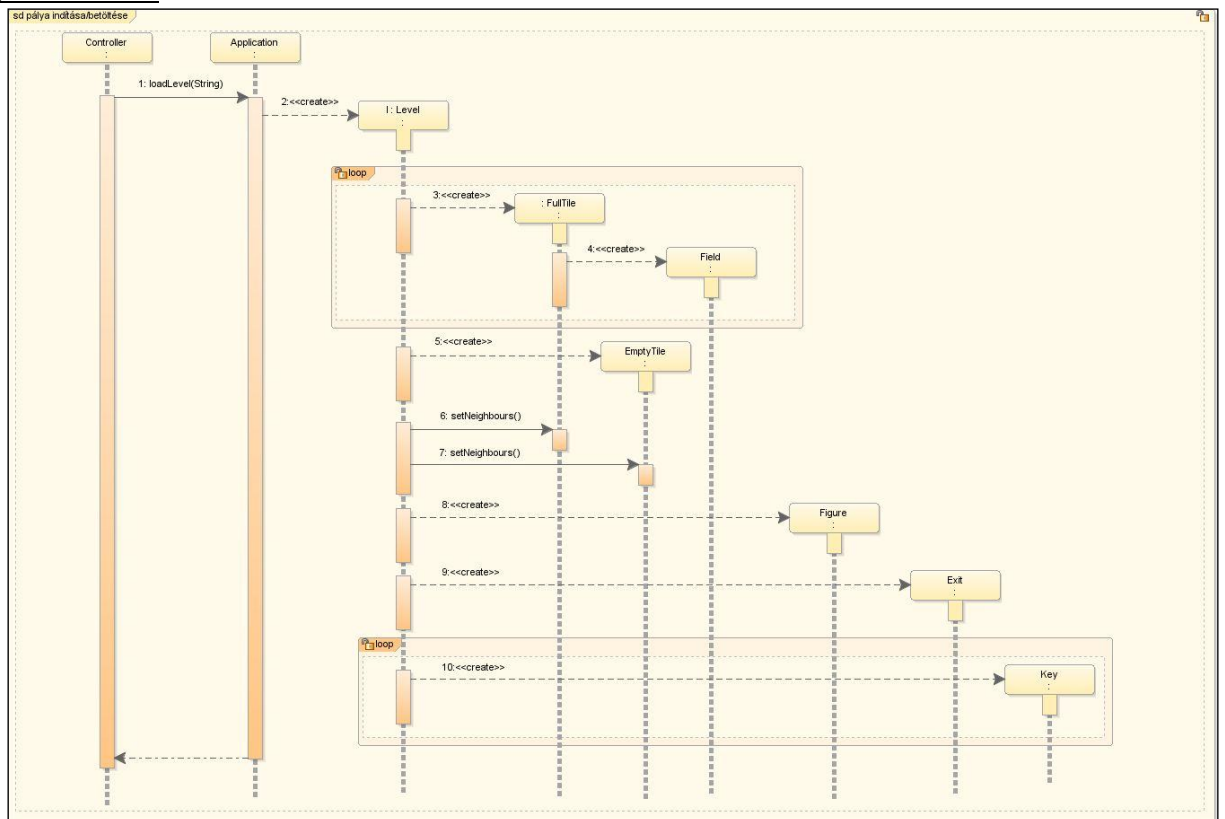


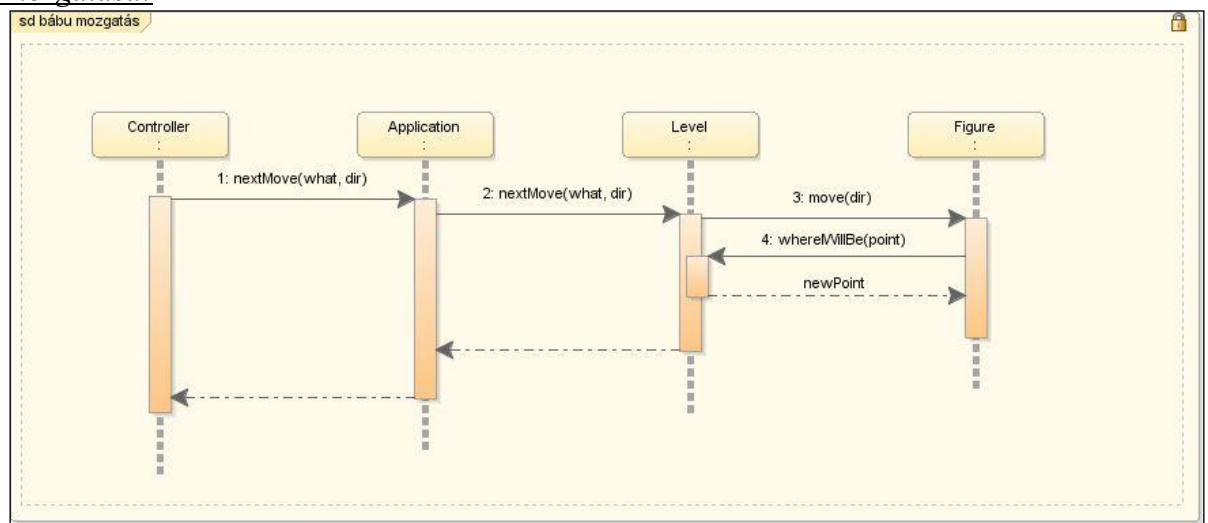
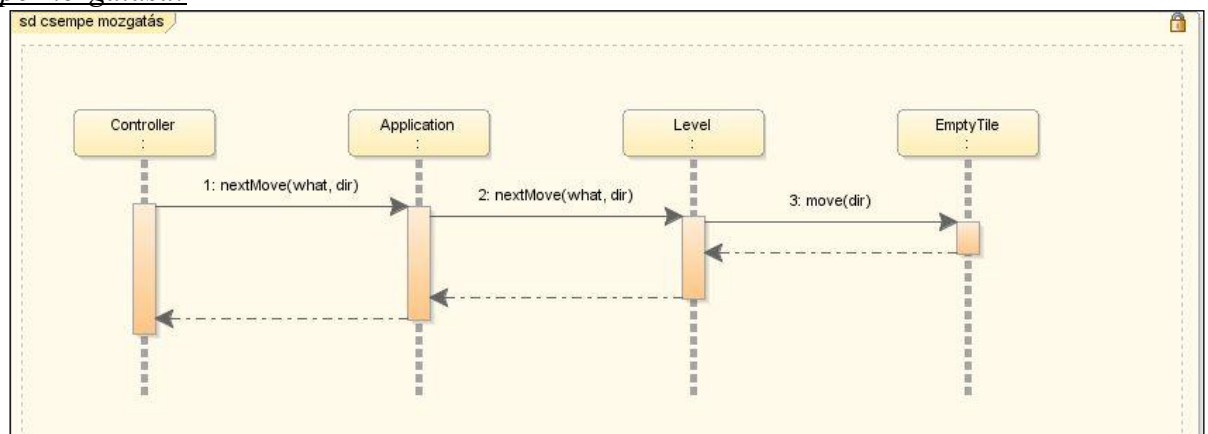
3.4 Szekvencia diagramok

A program indítása:

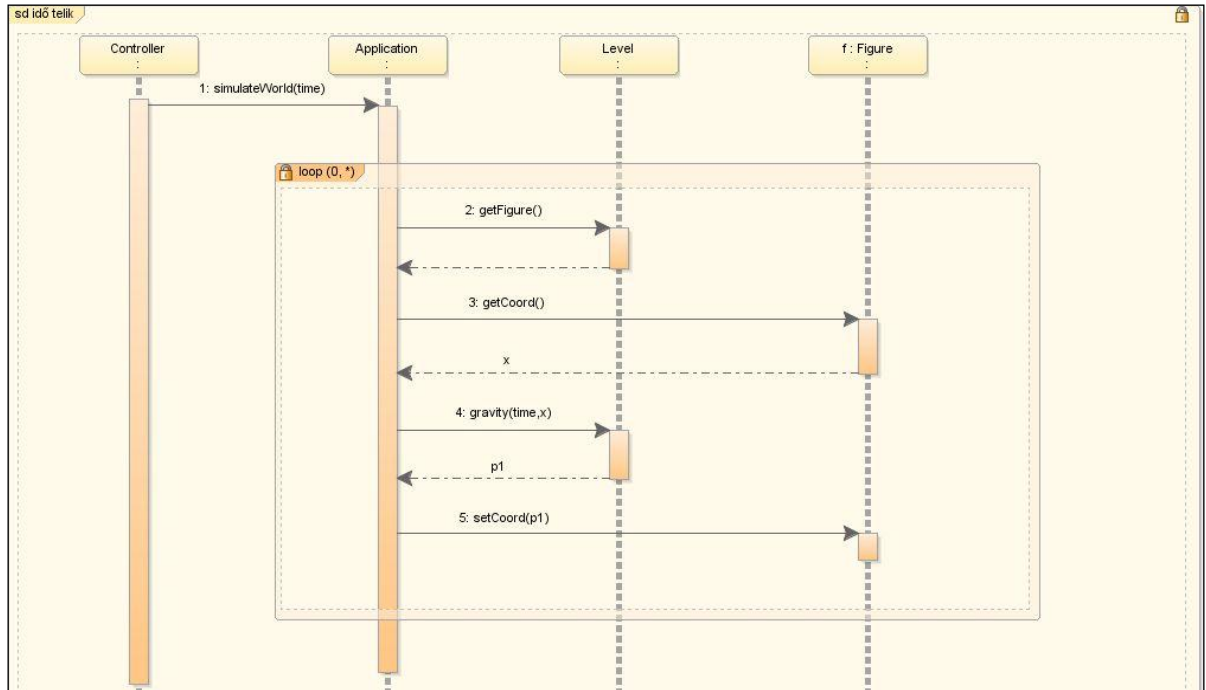


A pálya betöltése:

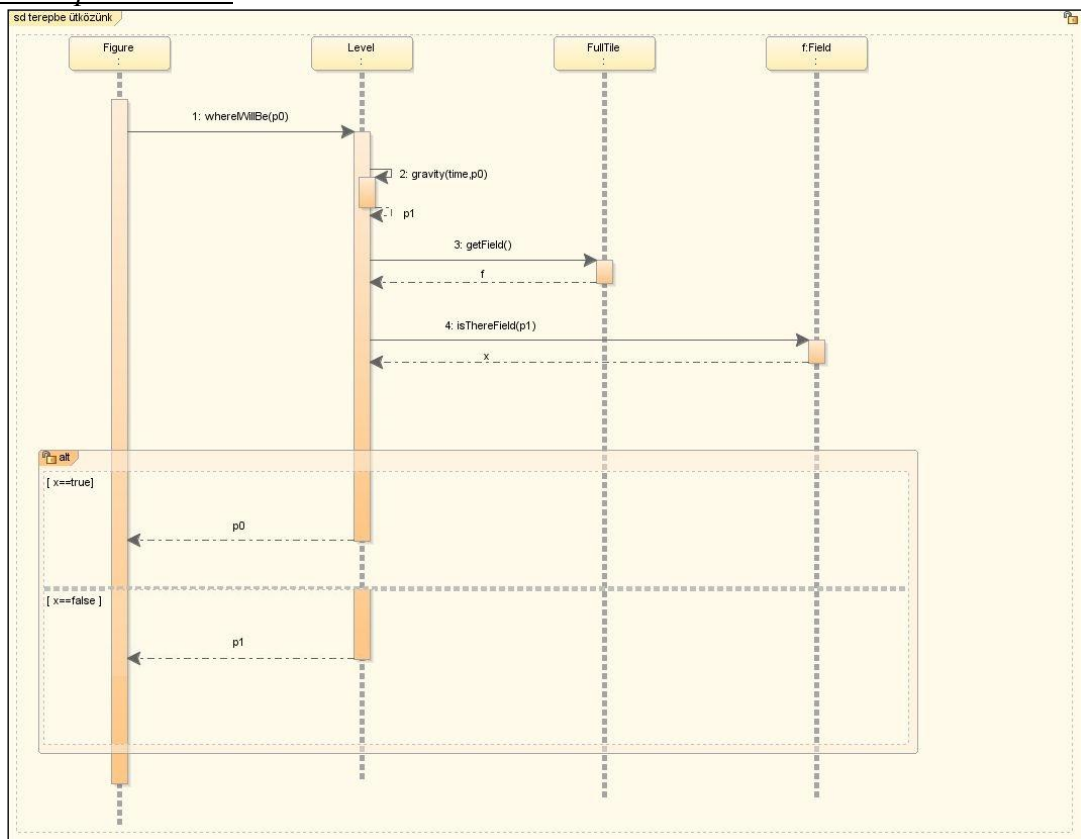


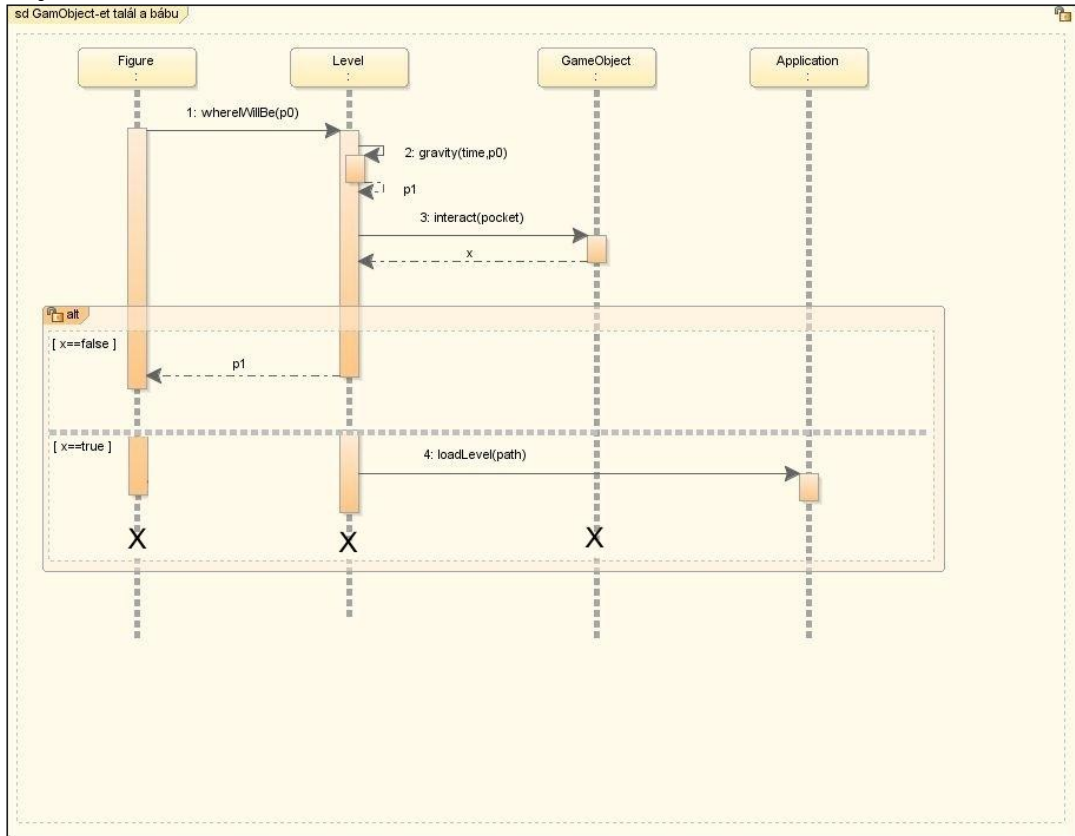
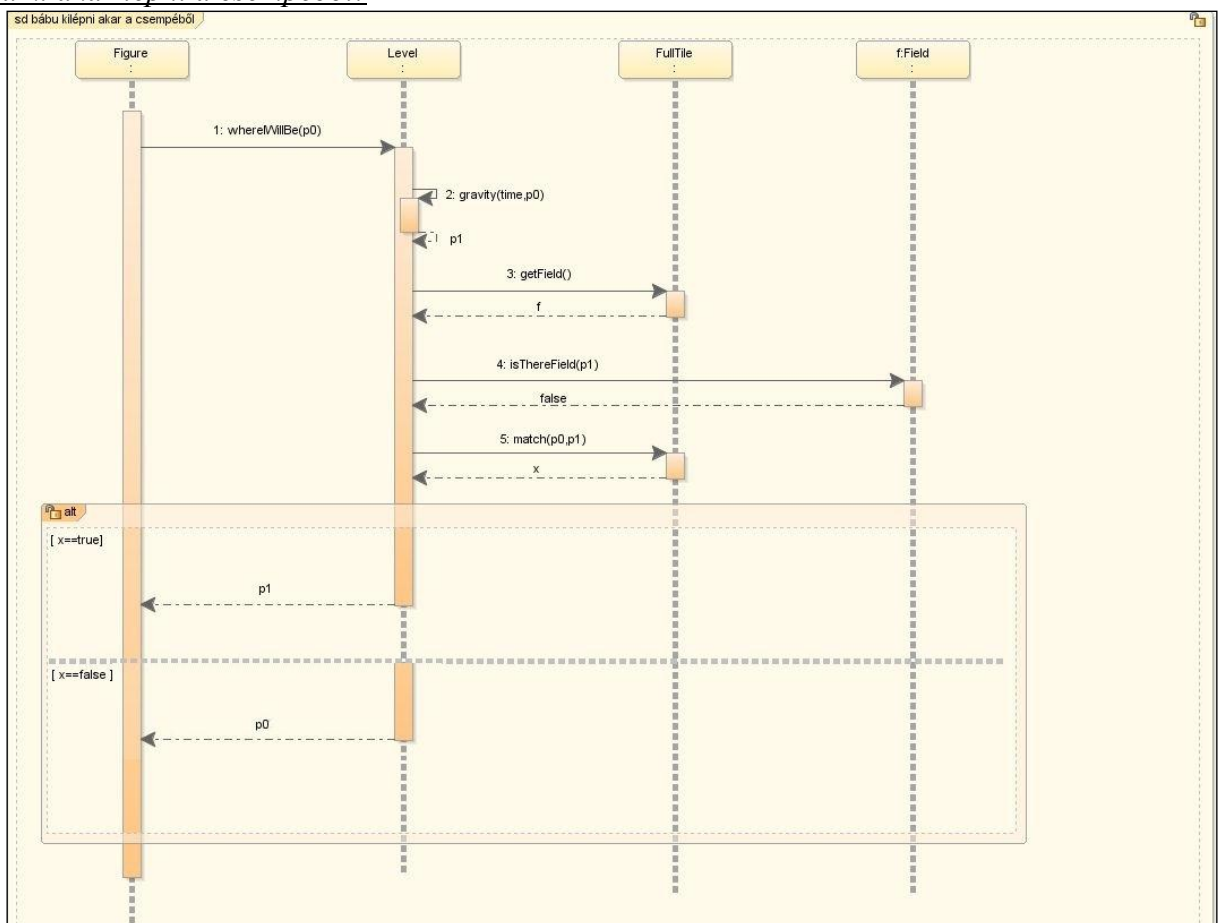
Bábu mozgatósa:Csempé mozgatósa:

Telik az idő:



A bábu terepbe ütközik:



GameObject-tel találkozik a bábu:A bábu ki akar lépni a csempéből:

3.5 State-chartok

A programban nincs állapotfüggő viselkedés, azaz nincs benne állapotgép.

4. Analízis modell kidolgozása II.

4.1 Objektum katalógus

4.1.1 Application

Ezt hozza létre először a modellek közül a Controller, felelős a szintek betöltéséért, a játékaállás mentéséért, egy korábban mentett állás betöltéséért, a program futásának befejezéséért. Mozgásra utasítja a csempéket vagy a figurát.

4.1.2 Controller

Ez jön létre először a program indításakor. Felelős az időkezelésért és a csempe vagy bábu mozgásáért.

4.1.3 Direction

Segédosztály, mely az irányvektort reprezentálja.

4.1.4 EmptyTile

Olyan mozgásra képes csempe, mely nem tartalmaz terepet. Benne bábu nem lehet.

4.1.5 Exit

Ezt kell elérni a megfelelő kulcsokkal a birtokunkban a szint teljesítéséhez. Tudja, hogy hány kulcsra van szüksége. Ha nincs elég kulcs a bábunál, akkor megy tovább a bábu.

4.1.6 Field

A terepet reprezentálja. Meg tudja mondani azt, hogy egy adott pontban van-e terep.

4.1.7 Figure

A felhasználó által irányított bábút valósítja meg. Tud mozogni. Képes megadni a szükséges adatokat a pontos elmozdulás-számításhoz, valamint képes visszaadni a zsebét.

4.1.8 Frame

A csempék keretét reprezentáló osztály. Meg tudja mondani, hogy két csempe terepe megegyezik-e a megadott szélen. A koordináták transzformálását végzi.

4.1.9 FullTile

Olyan csempe, amely tartalmaz terepet és olyan alakzatokat mellyel interakcióba tud lépni a bábu. Ebben a bábu tud mozogni. Kiszámolja a bábu pontos elmozdulásának irányát és nagyságát a bábút körülvevő tereptől függően. Meg tudja kérdezni a terepet, hogy adott pontban terep van-e. Meg tudja vizsgálni, hogy illeszkedik-e egy másik csempével, az alapján, hogy a bábu honnan hova akar elmozdulni.

4.1.10 GameObject

Ez az absztrakt osztály kezeli a bábu zsebét. Megmondja, hogy a vele történő érintkezés hatására a pálya teljesített lett-e

4.1.11 Key

A szint teljesítéséhez szükséges. Amikor a bábu eléri a kulcsot, akkor a kulcs beteszi önmagát a bábu zsebébe. Meg tudja mondani, hogy felvették-e.

4.1.12 Level

Felelős a csempék elrendezéséért, és a bábu kívánt irányú mozgásának beállításáért. Meg tudja mondani az adott szint kezdőcsempéjét és a bábu azon belüli kezdőpontját.

4.1.13 Tile

Ezzel az absztrakt osztállyal lehetséges a FullTile-ok és EmptyTile-ok egy tömbben tárolása.

4.1.14 Type

Olyan segédosztály, enum típus, melyet a mozgatóskor használunk arra a célra, hogy eldöntsük, csempét vagy bábút mozgatunk.

4.2 Osztályok leírása

4.2.1. Application

- **Felelősség**
 - kezeli az idő múlását
 - betölt egy szintet
 - kezeli a szinten lévő mozgásokat
 - menti a játékállást
 - kilép az alkalmazásból
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - **int levelsPathIndex:** értéke az aktuális szint indexe a levelsPath tömbben.
 - **String[] levelsPath:** A szintek elérési útját tárolja.
 - **String savedLevelPath:** Az elmentett szint elérési útja. Null, ha nincs mentett szint.
 - **Level actualLevel:** Az éppen betöltött szint.
 - **int previousTime:** Az előző simulateWorld futási ideje.
 - **String savedLevelPath:** Az elmentett szint elérési útja.
- **Metódusok**
 - **void simulateWorld(int time):** Az idő múlását szimulálja, a betöltött szint gravitációt kezelő függvényét hívja, ha van betöltött szint.
 - **void loadLevel():** Betölti az átadott elérési útnak megfelelő szintet. Visszaadja, hogy sikerült-e a betöltés.
 - **void move(Type what, Direction dir):** Ha van betöltött szint, akkor továbbítja a szintnek, hogy mi mozogjon (what paraméter) és milyen irányba (dir paraméter).
 - **void clearActualLevel():** Kinullázza a pillanatnyi pályát.
 - **void exit():** Kilép a programból és automatikusan menti a játék állását.

4.2.2. Controller

- **Felelősség**
 - időkezelés
 - mozgatósk kezelés
- **Ősosztályok**
Object
- **Interfészek**

nincs

- **Attribútumok**
 - **Application app:** A futtatandó program.
- **Metódusok**
 - **static void main(String[] args):** A program main függvénye.

4.2.3. Direction

- **Felelősség**
 - segédosztály, mely az irányvektort reprezentálja
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - **double angle:** irányvektor szöge
 - **double length:** irányvektor hossza
- **Metódusok**
 - **double getAngle:** Visszaadja az irányvektor szögét.
 - **double getLength:** Visszaadja az irányvektor hosszát.

4.2.4. EmptyTile

- **Felelősség**
 - mozogni tud
- **Ősosztályok**
Tile
- **Interfészek**
nincs
- **Attribútumok**
nincs
- **Metódusok**
 - **void move(Direction dir)** - *Az üres csempe mozgatását végző függvény. Ha van abba az irányba másik csempe, amerre mozgatni akarjuk, akkor megcseréli a két csempét*

4.2.5. Exit

- **Felelősség**
 - meg tudja mondani, hány kulcs szükséges a szint teljesítéséhez
 - be tudja állítani a szükséges kulcsok számát
 - megvizsgálja a kapott kulcsokat
- **Ősosztályok**
GameObject
- **Interfészek**
nincs
- **Attribútumok**
 - **int numberOfKeys:** A szint teljesítéséhez szükséges kulcsok száma.
- **Metódusok**
 - **int getNumberOfKeys():** Visszaadja a szint teljesítéséhez szükséges kulcsok számát.
 - **boolean interact(Figure figure):** Megvizsgálja a kapott figura zsebében lévő kulcsokat. Ha megfelelő számú van, akkor true-val tér vissza, ha nem akkor false-szal.

- **void setNumberOfKeys(int number):** Beállítja a szint teljesítéséhez szükséges kulcsok számát.

4.2.6. Field

- **Felelősség**
 - megmondja, hogy az adott koordinátán terep van-e
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - **ArrayList<Point> points:** A terepet reprezentáló pontok. A terepen egyenesek vannak. Egy egyenes két végpontját egy Point collection tárolja index alapján sorban. Több Point tömb (egyenes) reprezentálja a terepet.
- **Metódusok**
 - **boolean isThereField(Point point):** Megmondja, hogy a paraméterként kapott pontban terep van-e.

4.2.7. Figure

- **Felelősség**
 - meg tudja mondani az elmozduláshoz szükséges információkat
 - át tudja adni a zsebét
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - **Point coord:** A bábu koordinátája.
 - **Direction dir:** a felhasználói input által beállított irányvektor, ebbe az irányba akar elmozdulni a bábu.
 - **FullTile myTile:** Referencia arra a csempére, amiben éppen van a bábu.
 - **Direction oldDirection:** az előző lépésben kiszámított elmozdulás vektor
 - **ArrayList<Key> pocket:** A bábu zsebe, melyben kulcsokat tárolunk.
- **Metódusok**
 - **Point getCoord():** Visszaadja a bábu koordinátáját.
 - **Direction getDirection():** Visszaadja az input alapján beállított irányvektort.
 - **Direction getOldDirection():** Visszaadja az előző lépésben számított elmozdulásvektort.
 - **ArrayList<Key> getPocket():** Visszaadja a bábu zsebét.
 - **boolean interact(GameObject g):** GameObject interact függvény. Paraméterként kapja azt a GameObject-et, amivel interactolni akarunk.
 - **void move(int dt):** Mozgatja a bábút.
 - **void setCoord(point:Point):** Beállítja a bábu koordinátáját.
 - **void setDirection(Direction dir):** Beállítja az input alapján a kívánt elmozdulás irányvektorát.
 - **void setOldDirection(Direction dir):** Beállítja a kiszámított elmozdulás vektort.
 - **void setMyTile(FullTile myTile):** Beállítja a csempére mutató referenciát.

4.2.8. Frame

- **Felelősség**

- meg tudja mondani, hogy egyezik-e a két csempe terepe a megadott szélén (önmagát hasonlítja össze egy másik kerettel)
- le tudja ellenőrizni, hogy az adott koordinátán keret van-e
- belső koordinátát közös koordinátává tud átváltani
- közös koordinátát belső koordinátává tud átváltani
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - **ArrayList<Point> framePoints:** A terepet szélét reprezentáló pontok. Ebből áll a teli csempe kerete.
- **Metódusok**
 - **boolean checkFrame(Point coord):** Megnézi, hogy a paraméterként kapott koordinátán keret van-e.
 - **Point deTransformCoord(Point coord):** Közös koordinátát belső koordinátává vált át.
 - **boolean match (Frame frame):** Megvizsgálja, hogy egyezik-e a paraméterként kapott kerettel.
 - **Point transformCoord(Point coord):** Belső koordinátát közös koordinátává vált át.

4.2.9. FullTile

- **Felelősség**
 - illeszkedést kezel a csempék között
 - vizsgálja a benne lévő terepet
 - kiszámítja a bábu pontos elmozdulását
- **Ősosztályok**
Tile
- **Interfészek**
nincs
- **Attribútumok**
 - **Field field:** A benne lévő terep.
 - **ArrayList<Frame> frames:** Csempe kereteit tároló kollekció.
 - **Figure figure:** Referencia a bábura, értéke null, ha a bábu nincs a csempében.
 - **ArrayList<GameObject> gameObjects:** A csempében lévő kulcs(ok)at és/vagy kijáratot tároló tömb.
- **Metódusok**
 - **Point calculateNewCoord():** Visszaadja a bábu pontos elmozdulását a bábura ható pillanatnyi erőhatások figyelembevételével.
 - **void changeDirection(Direction dir):** A bábu kívánt elmozdulásának beállításáért felelős függvény.
 - **Point changePosition(Point coord):** Megváltoztatja a csempe helyét, visszaadja a csempe régi helyét.
 - **GameObject findGameObject():** Visszaadja azt a GameObject objektumot, ami a kiszámított elmozdulás helyén található. Ha nincs ott objektum, vagy a csempe nem tartalmaz GameObjecteket, visszatérési értéke null.
 - **Frame getFrame(Direction dir):** Visszaadja a bekért oldalon lévő keretet.
 - **boolean isThereFigure():** Megmondja, hogy van-e figura a csempében.
 - **Point moveFigure():** Kiszámítja a bábu pontos elmozdulását.

- **void setFigure(Figure f):** Beállítja a bábu referenciáját.
- **void setFigureCoord(Point coord):** Beállítja a benne lévő bábu koordinátáját.
- **void simulateWorld(int dt):** Időkezelés továbbadása a bábunak.

4.2.10. *GameObject*

- **Felelősség**
 - ez az absztrakt osztály kezeli a bábu zsebét
 - megmondja, hogy a vele történő érintkezés hatására a pálya teljesített lett-e
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - **Point coord:** Megmondja, hol helyezkedik el a GameObject.
- **Metódusok**
 - **Point getCoord():** Visszaadja a koordinátáját.
 - **boolean interact (Figure f):** Interactol a kapott figurával.
 - **void setCoord(Point point):** Beállítja a koordinátáját.

4.2.11. *Key*

- **Felelősség**
 - tudja, hogy felvették-e
 - beleteszi magát a bábu zsebébe
- **Ősosztályok**
GameObject
- **Interfészek**
nincs
- **Attribútumok**
 - **boolean pickedUp:** Megmondja, hogy felvették-e már.
- **Metódusok**
 - **boolean isPickedUp():** Visszaadja, hogy felvették-e már.
 - **boolean interact(Figure figure):** Interactol a paraméterként kapott figura zsebével (beteszi magát).

4.2.12. *Level*

- **Felelősség**
 - A szinten lévő csempék elrendezése
 - Meg tudja adni egy csempe adott oldali szomszédját
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - **Application app:** Referencia az alkalmazásra, ami létrehozta. A következő szint betöltésekor használjuk.
 - **Collection tiles:** A szinten lévő csempék. A 0. indexű elem az egyetlen üres csempe (EmptyTile).
 - **Point figureAStartCoord:** A bábu kezdőcsempén belüli kezdőkoordinátái
 - **Point figureBStartCoord:** B bábu kezdőcsempén belüli kezdőkoordinátái
 - **int figureAStartTile:** A kezdőcsempe referenciája

- **int figureBStartTile:** B kezdőcsempe referenciája
- **Metódusok**
 - **void finished():** jelzi az Application felé, hogy a szint teljesítve lett.
 - **Point getFigureAStartCoord():** Referenciát ad vissza A kezdő koordinátákra.
 - **Point getFigureBStartCoord():** Referenciát ad vissza B kezdő koordinátákra.
 - **int getFigureStartTileIndex():** Visszaadja a kezdőcsempe indexét.
 - **FullTile getNeighbour(Direction dir, Point pos):** Referenciát ad az argumentumok alapján keresett csempére. Egy csempe adott oldali szomszédjának kiderítésére használjuk.
 - **FullTile getFigureAStartTile():** Referenciát ad vissza A bábu kezdő csempéjére.
 - **FullTile getFigureBStartTile():** Referenciát ad vissza B bábu kezdő csempéjére.
 - **void move(Type what, Direction dir):** Megkapja, hogy mi mozogjon, és ennek hívja meg a mozgató metódusát, aminek átadja a kapott irányt (dir).
 - **void simulateWorld(int dt):** Továbbadja az időt annak a FullTile-nak, amelyikben a bábu van.

4.2.13. *Tile*

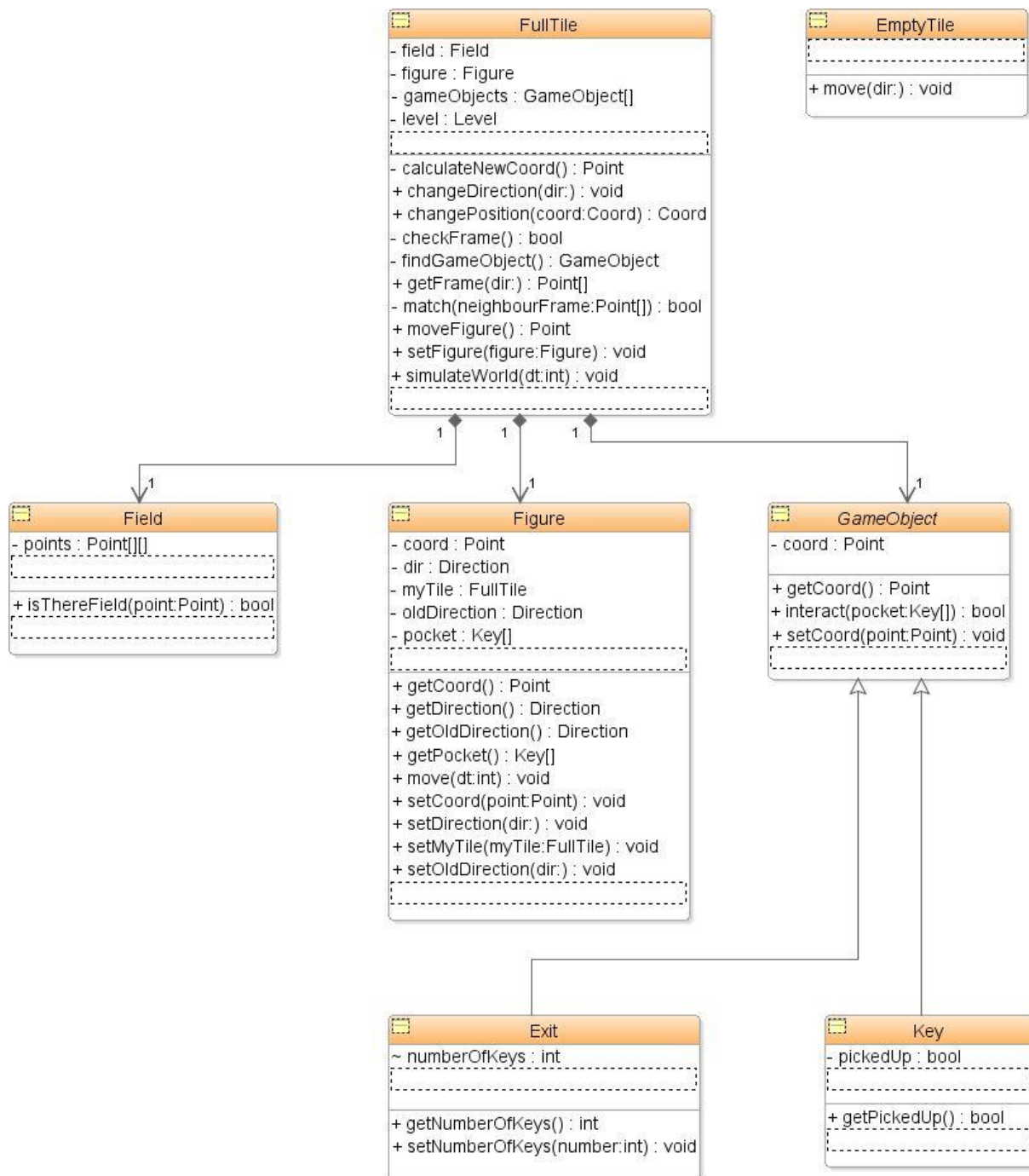
- **Felelősség**
 - ezzel az absztrakt osztállyal lehetséges a FullTile-ok és EmptyTile-ok egy tömbben tárolása
- **Ősosztályok**
nincs
- **Interfészek**
nincs
- **Attribútumok**
 - **Point coord:** A csempe koordinátája.
 - **int height:** A csempe magassága.
 - **Level myLevel:** A csempe saját szintje.
 - **int width:** A csempe szélessége.
- **Metódusok**
 - **Point getCoord() :** Megadja a csempe koordinátáját.
 - **int getHight():** Megadja a csempe magasságát.
 - **int getWidth():** Megadja a csempe szélességét.
 - **void setCoord():** Beállítja a csempe koordinátáját.

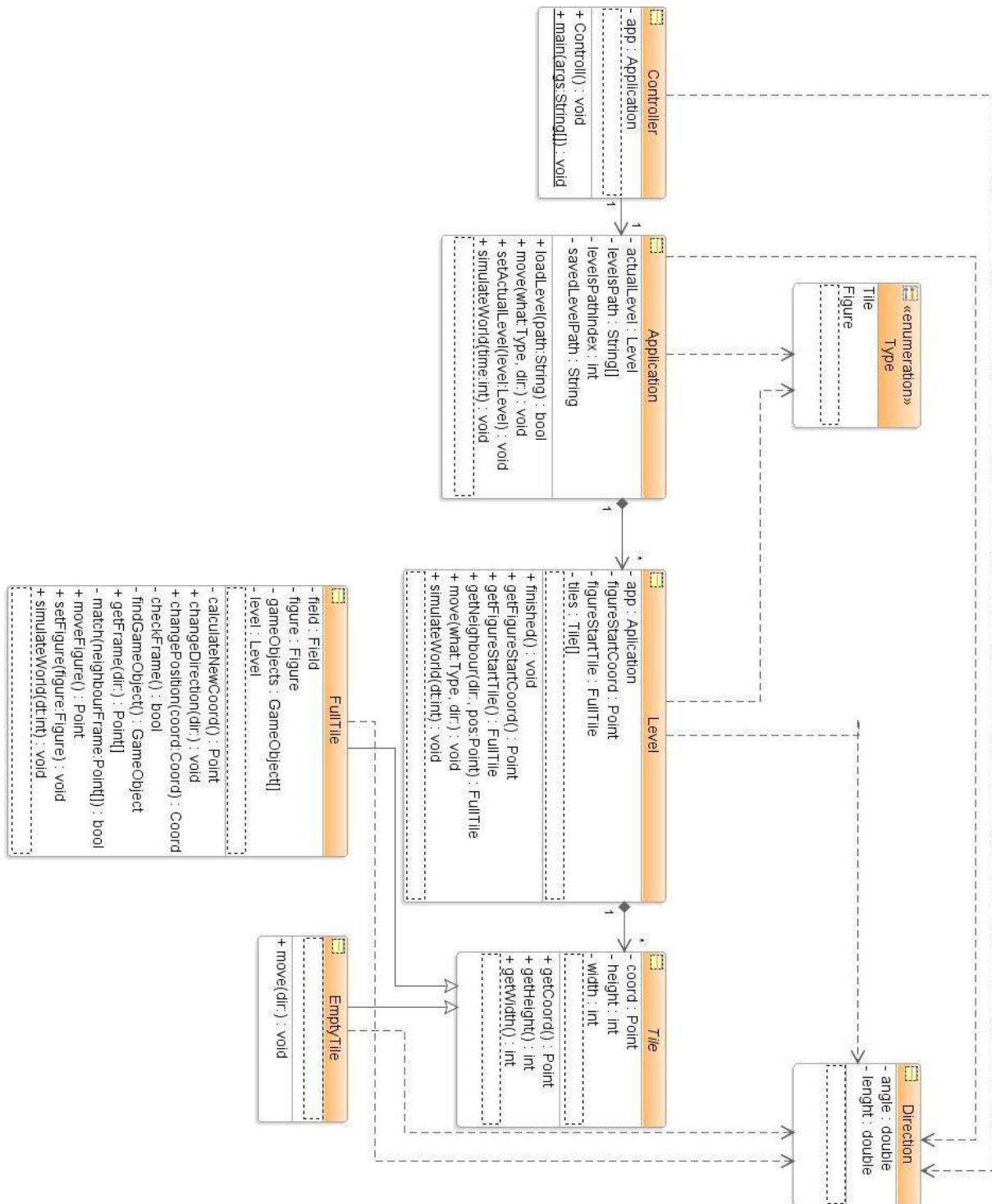
4.2.14. *Type*

- **Felelősség**
 - segédosztály, enum típus mely a mozgatáskor használunk arra hogy eldöntsük hogy csempét vagy bábut mozgatunk
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Enum állapotok:**
 - Tile
 - FigureA
 - FigureB

4.3 Statikus struktúra diagramok

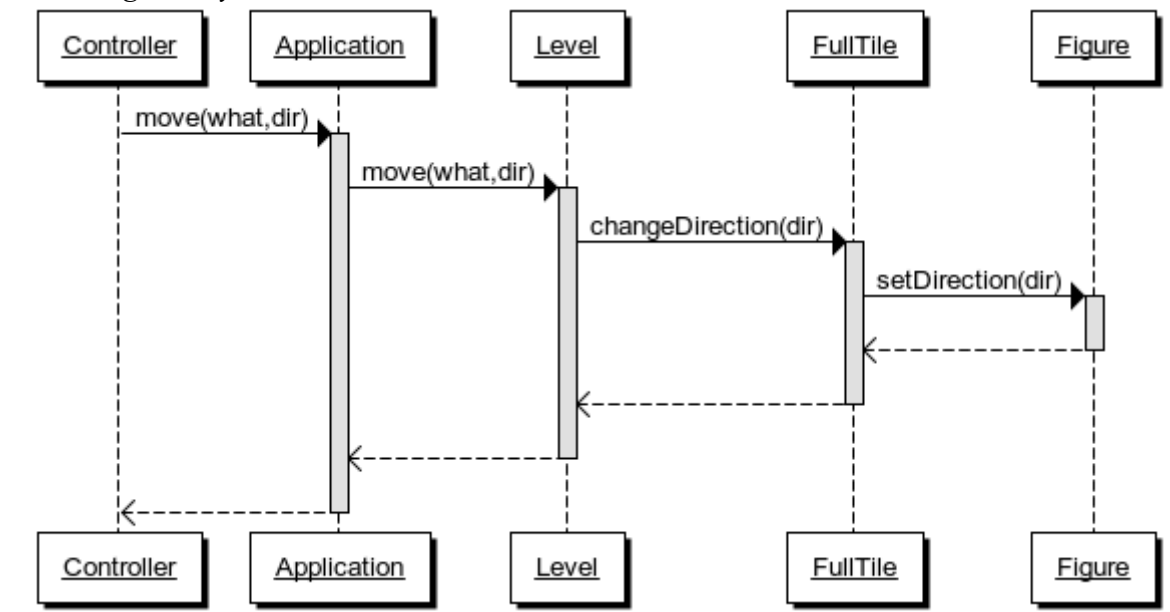
1. rész:



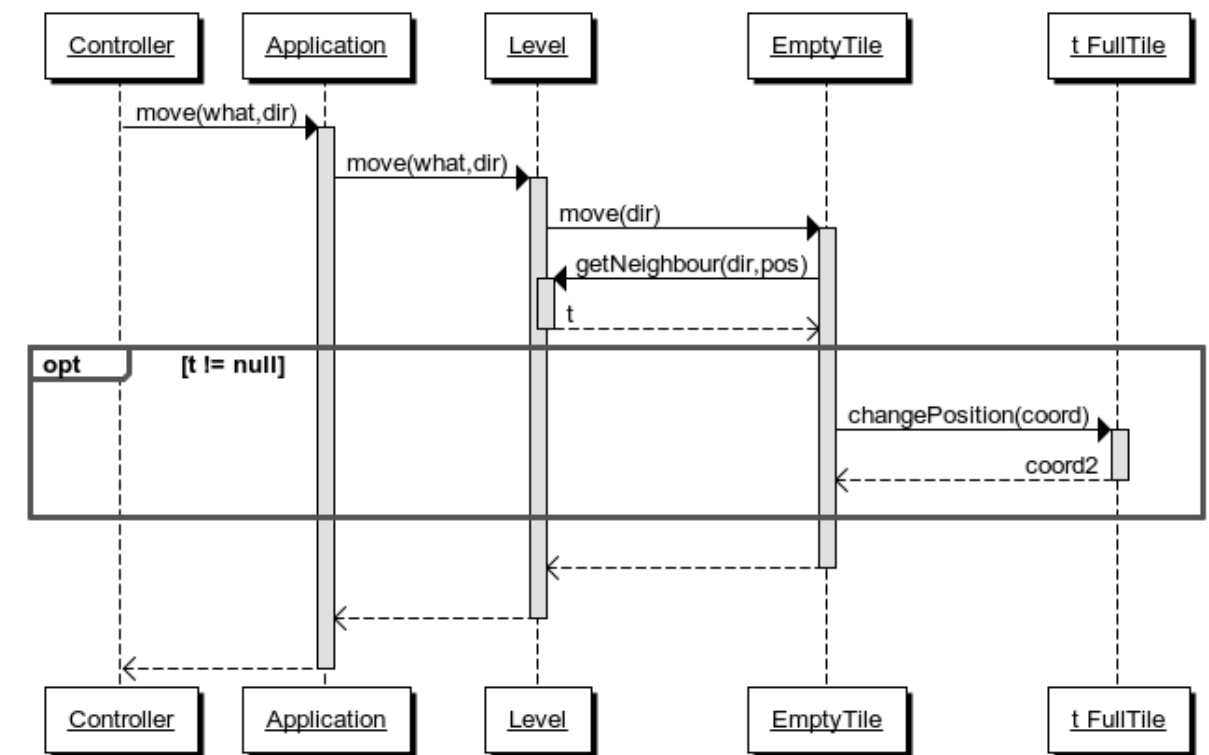
2. rész:

4.4 Szekvencia diagramok

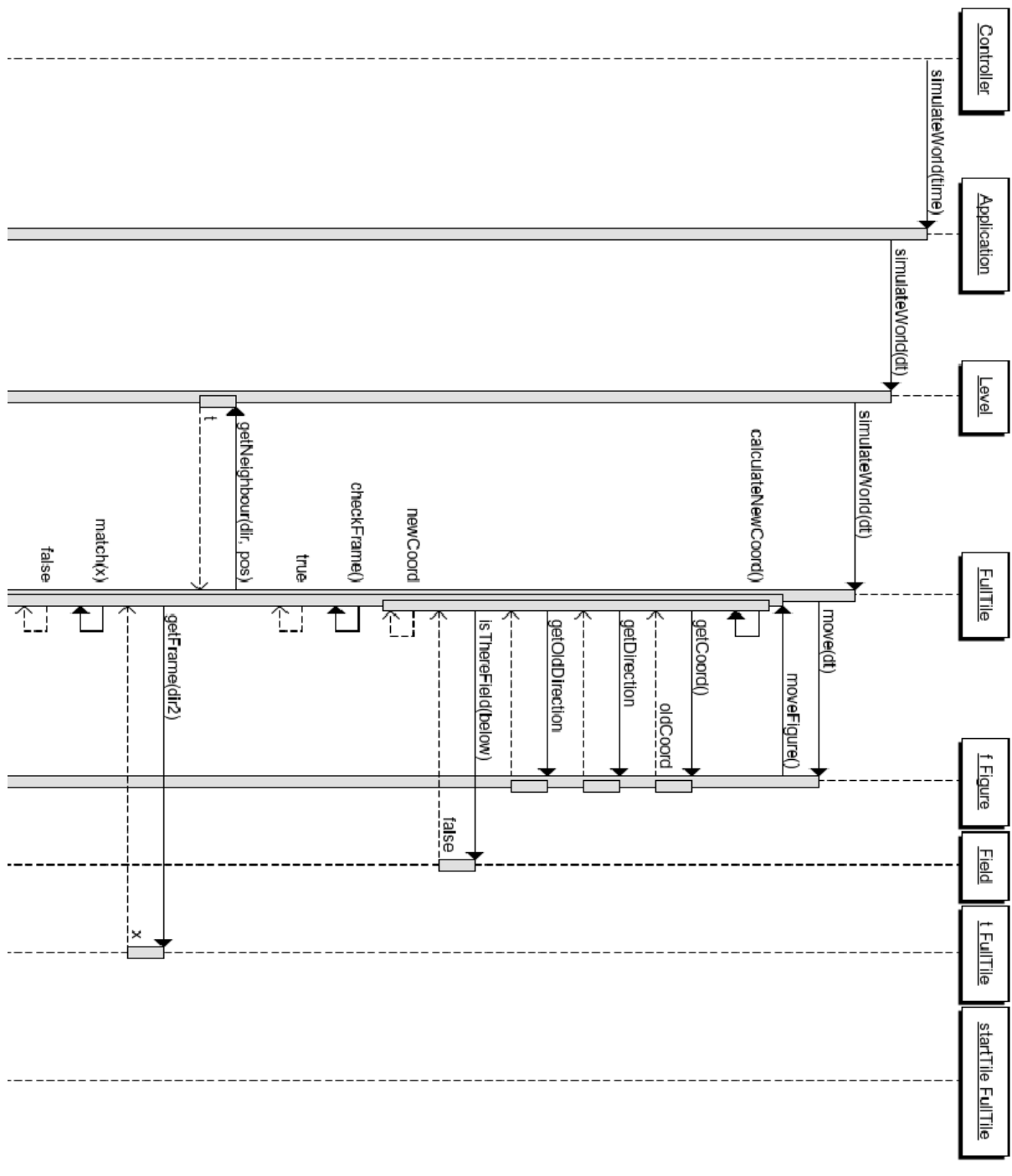
Bábu mozgásirányának beállítása:



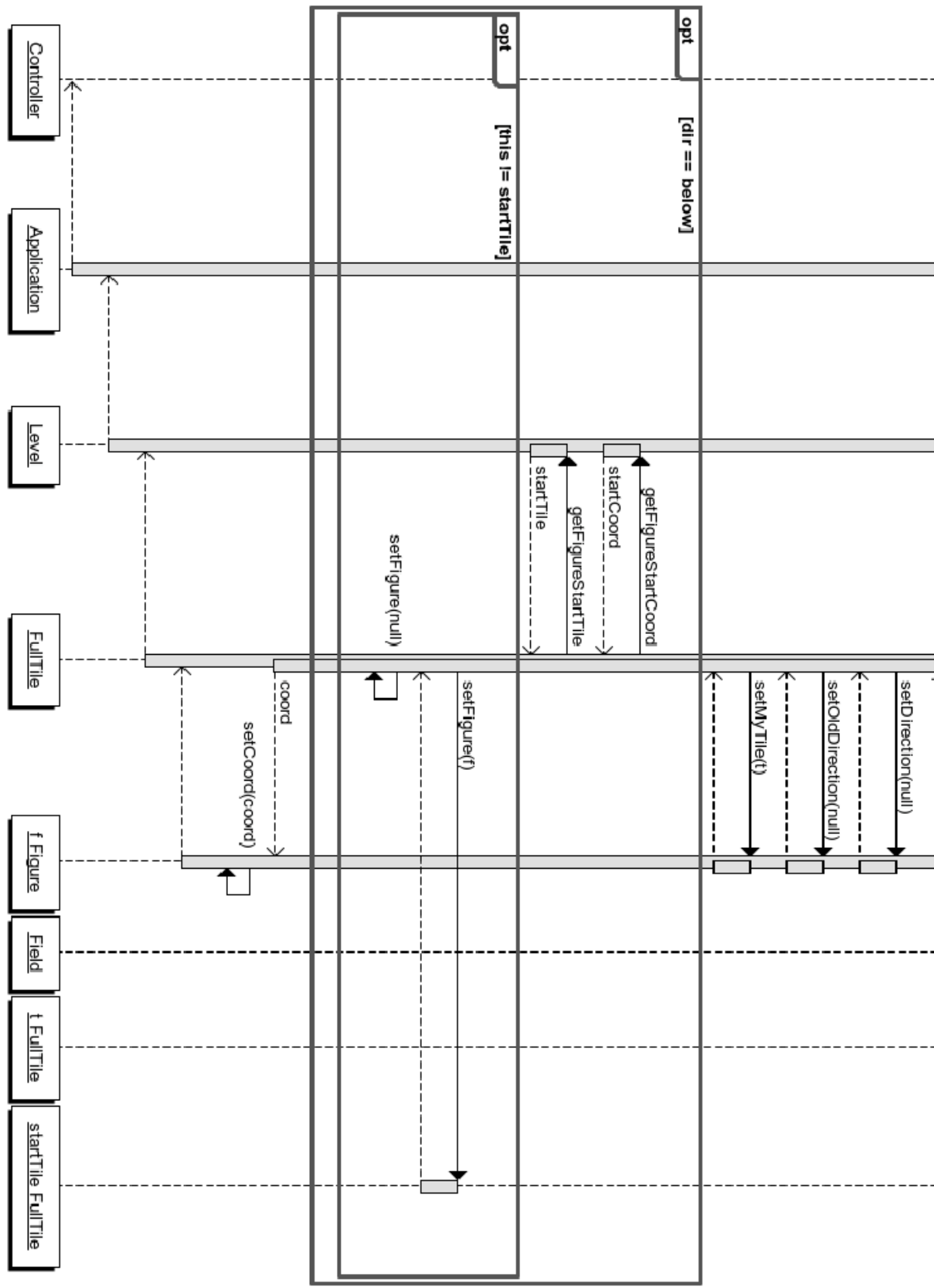
Csempé tolása sikeres vagy sikertelen:



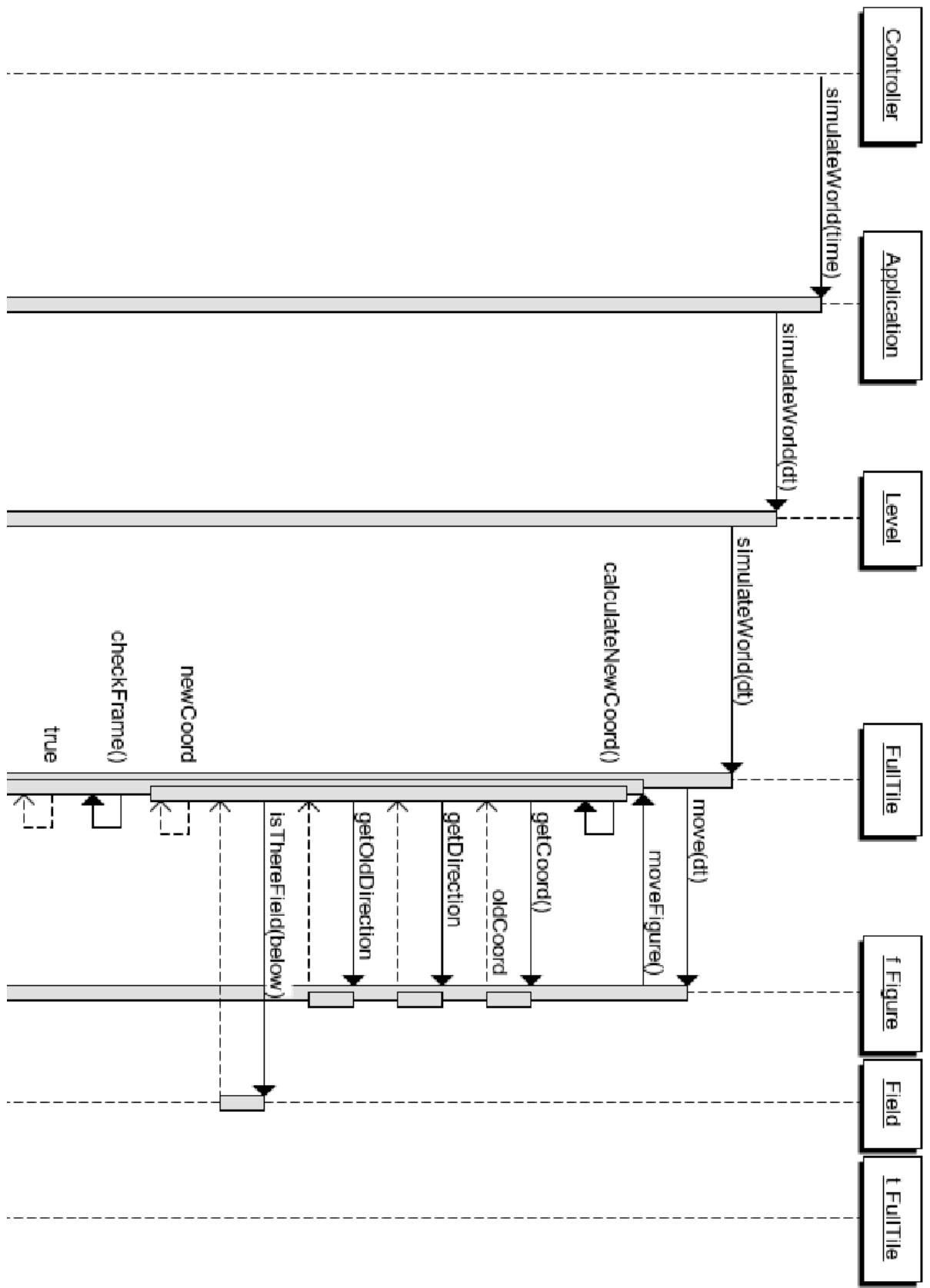
Csempén át akar menni a bábu, nem illeszkedik, meghal – 1. rész:



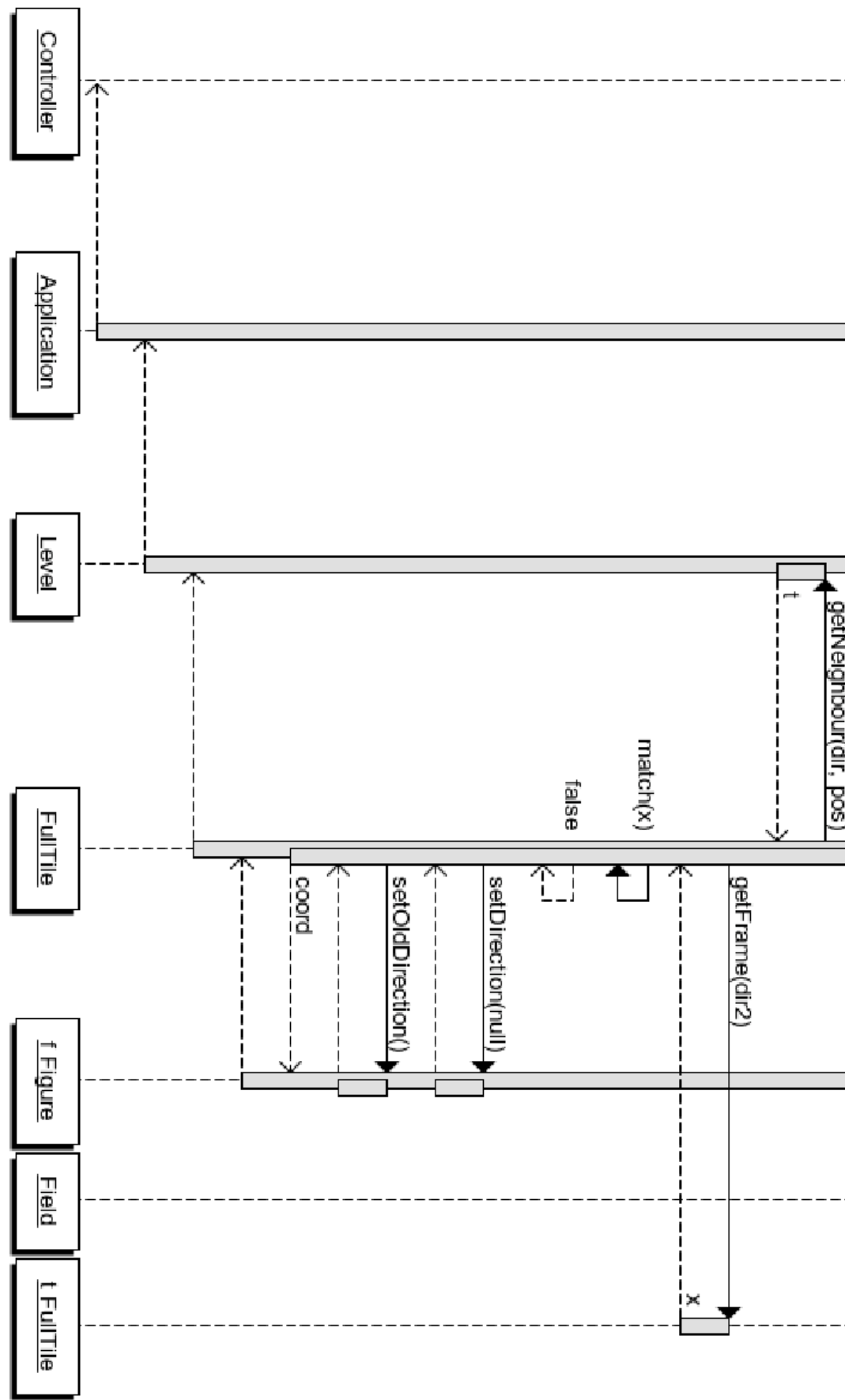
Csempén át akar menni a bábu, nem illeszkedik, meghal – 2. rész:



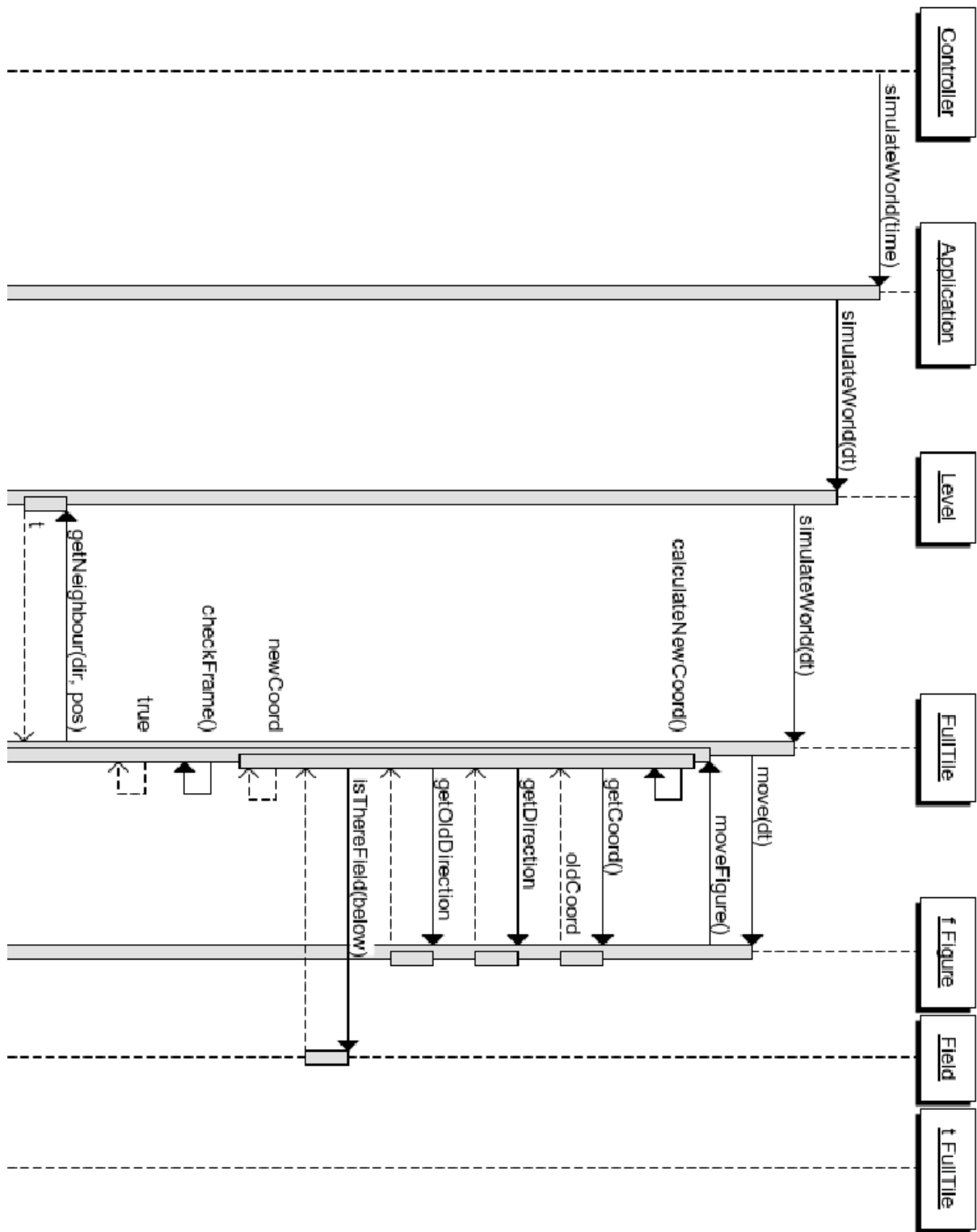
Csempén át akar menni a bábu alul, nem illeszkedik, ugyanott marad – 1. rész:



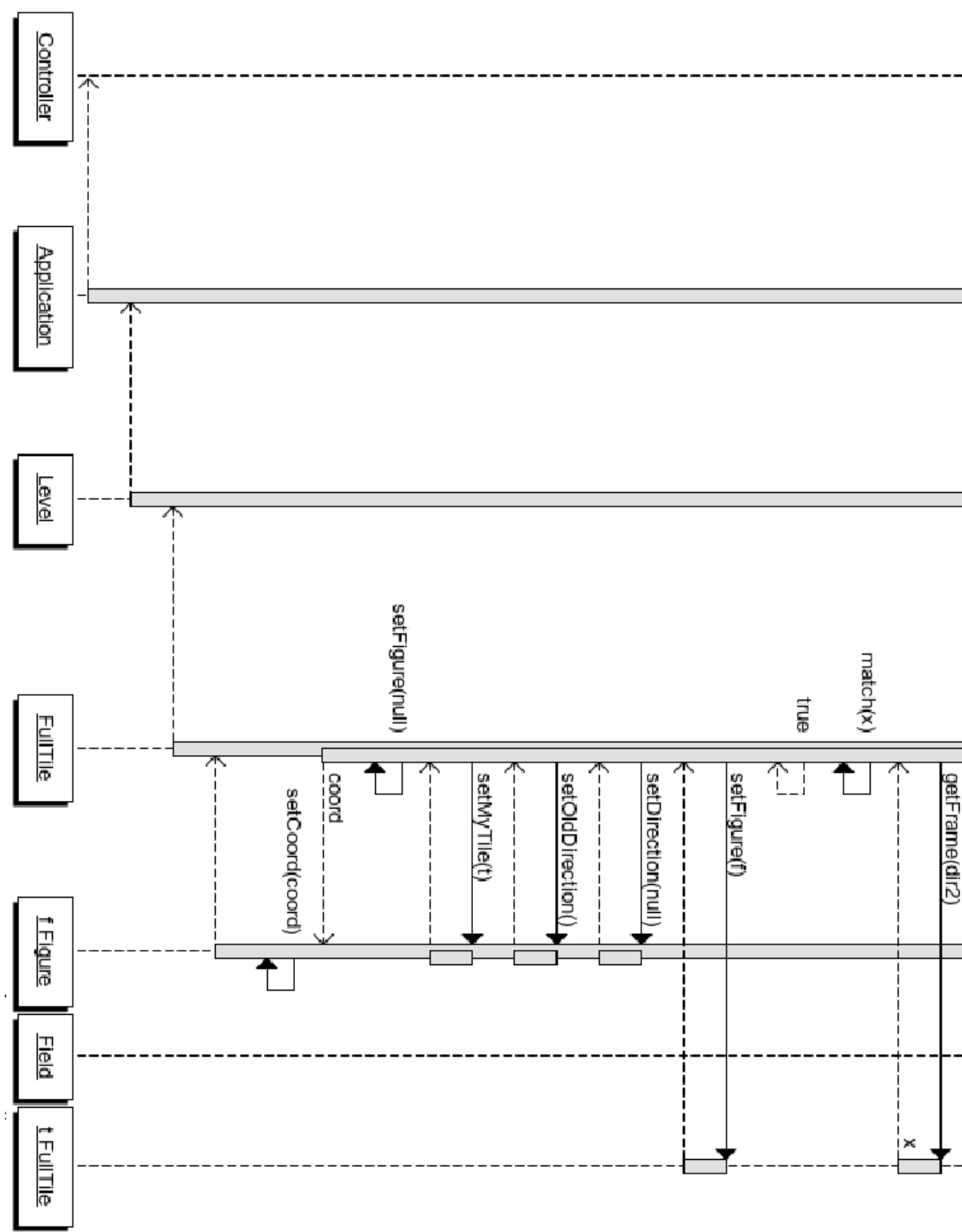
Csempén át akar menni a bábu alul, nem illeszkedik, ugyanott marad – 2. rész:



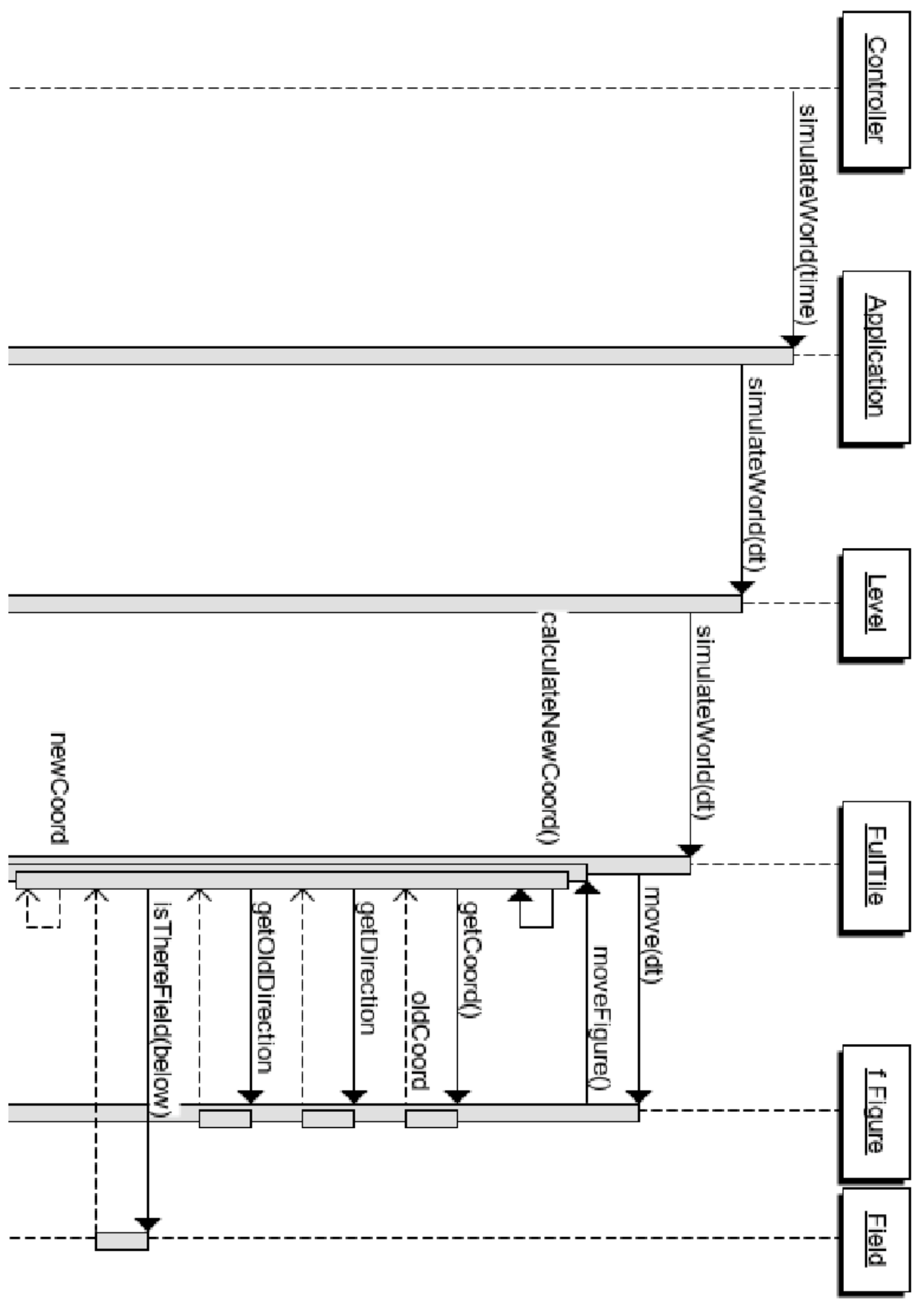
Csempén átmegy a bábu, illeszkednek a csempék – 1. rész:



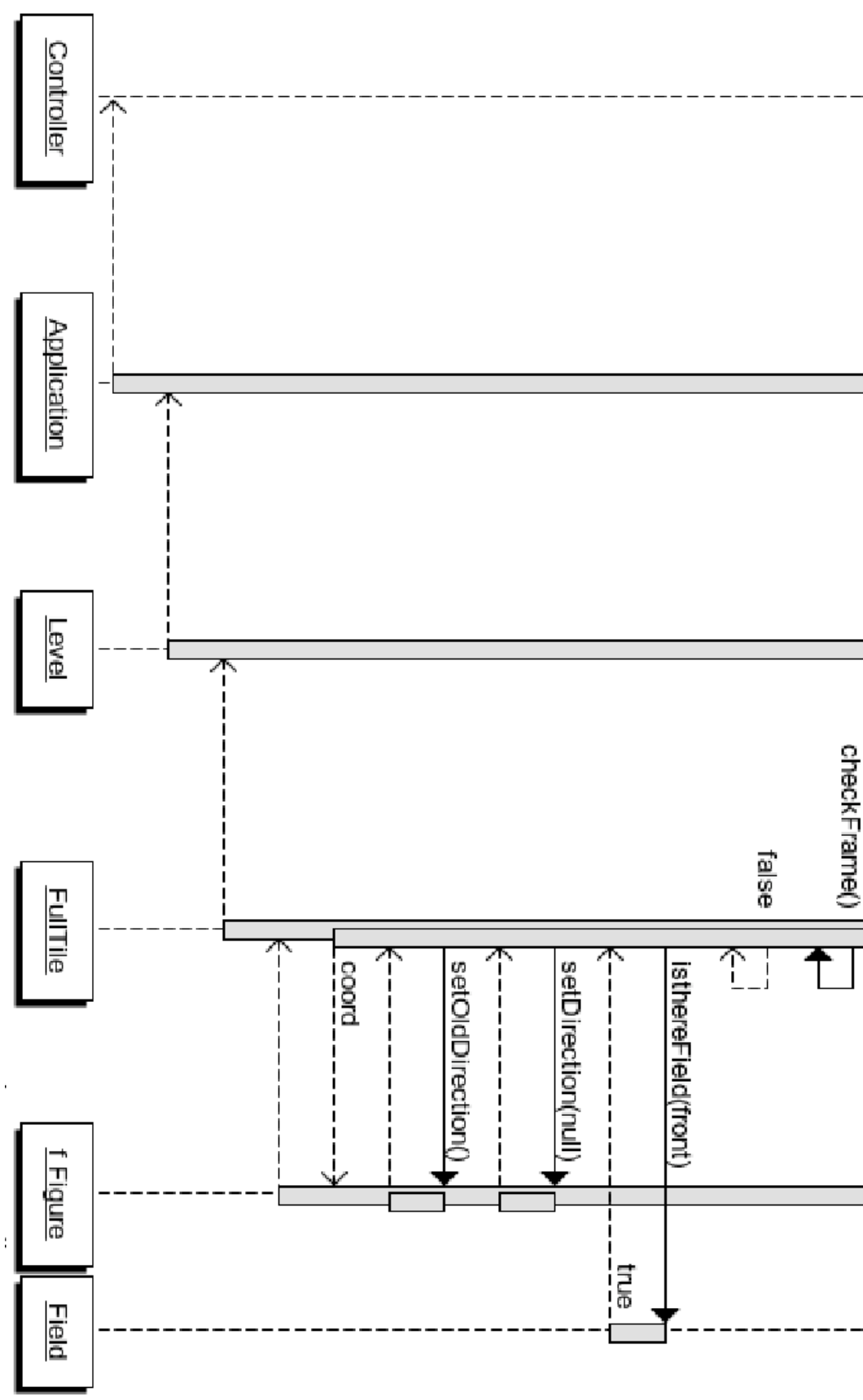
Csempén átmegy a bábu, illeszkednek a csempék – 2. rész:

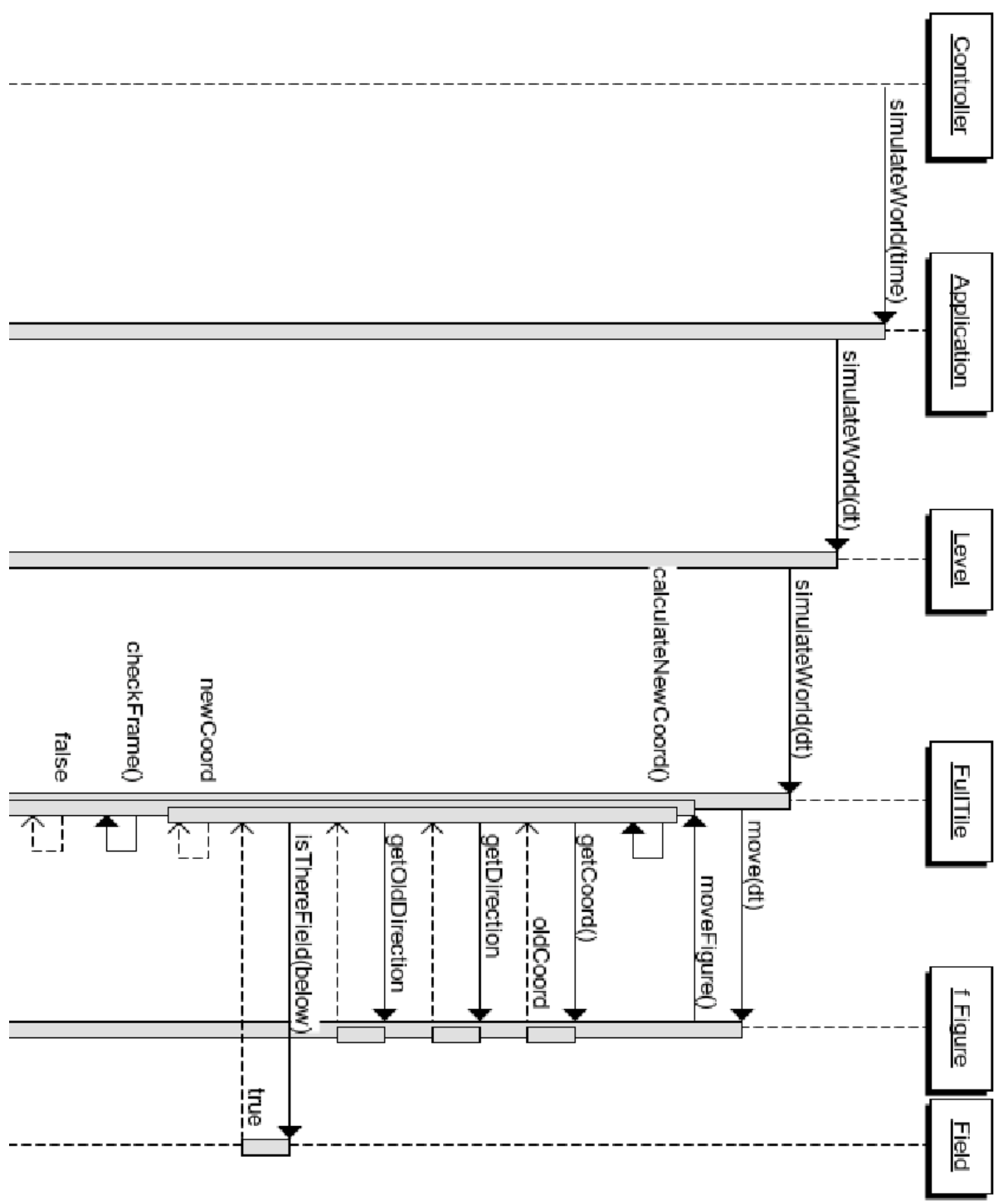


Falnak megy a bábu, nem tud arrébb menni – 1. rész:

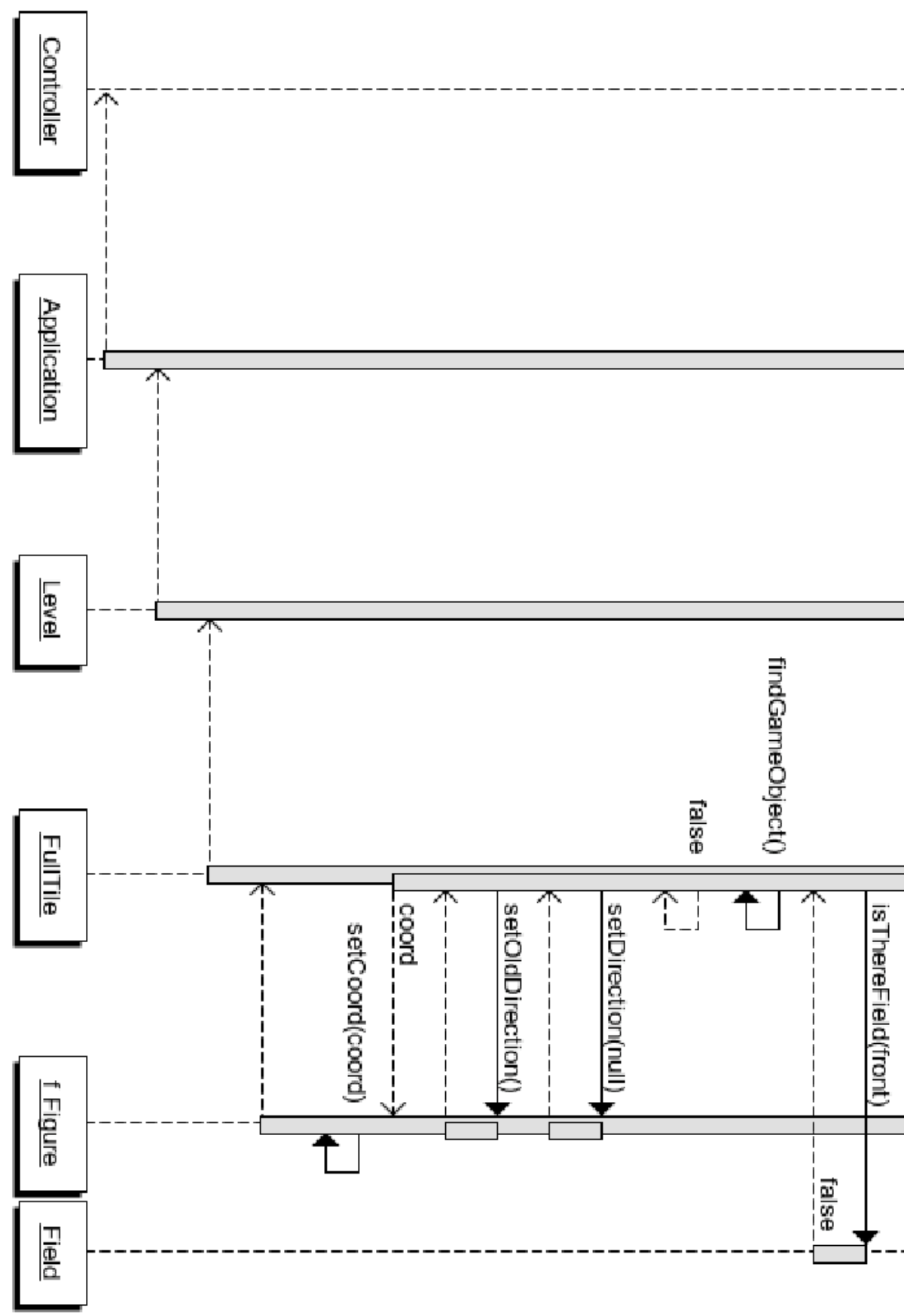


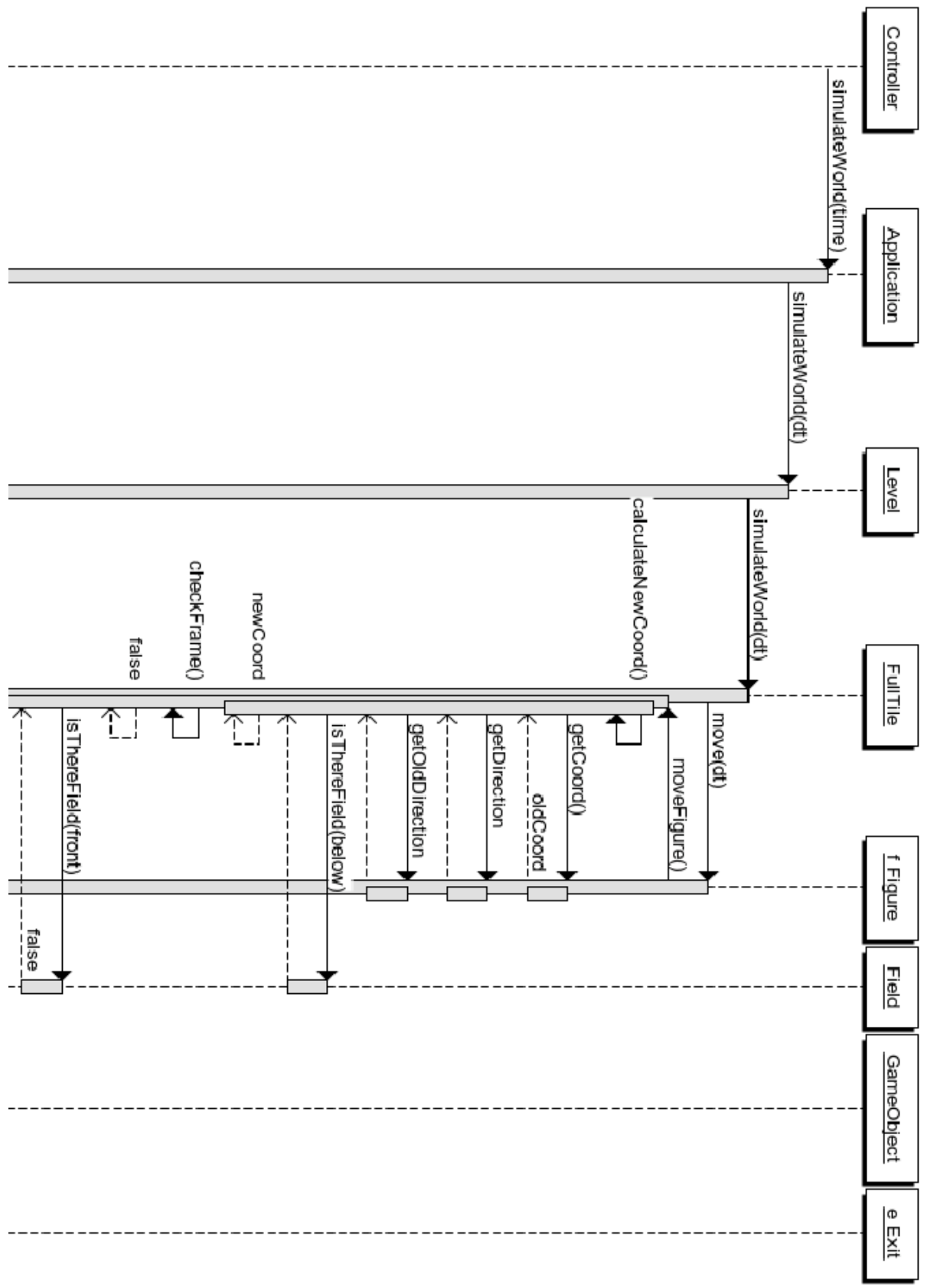
Falnak megy a bábu, nem tud arrébb menni – 2. rész:



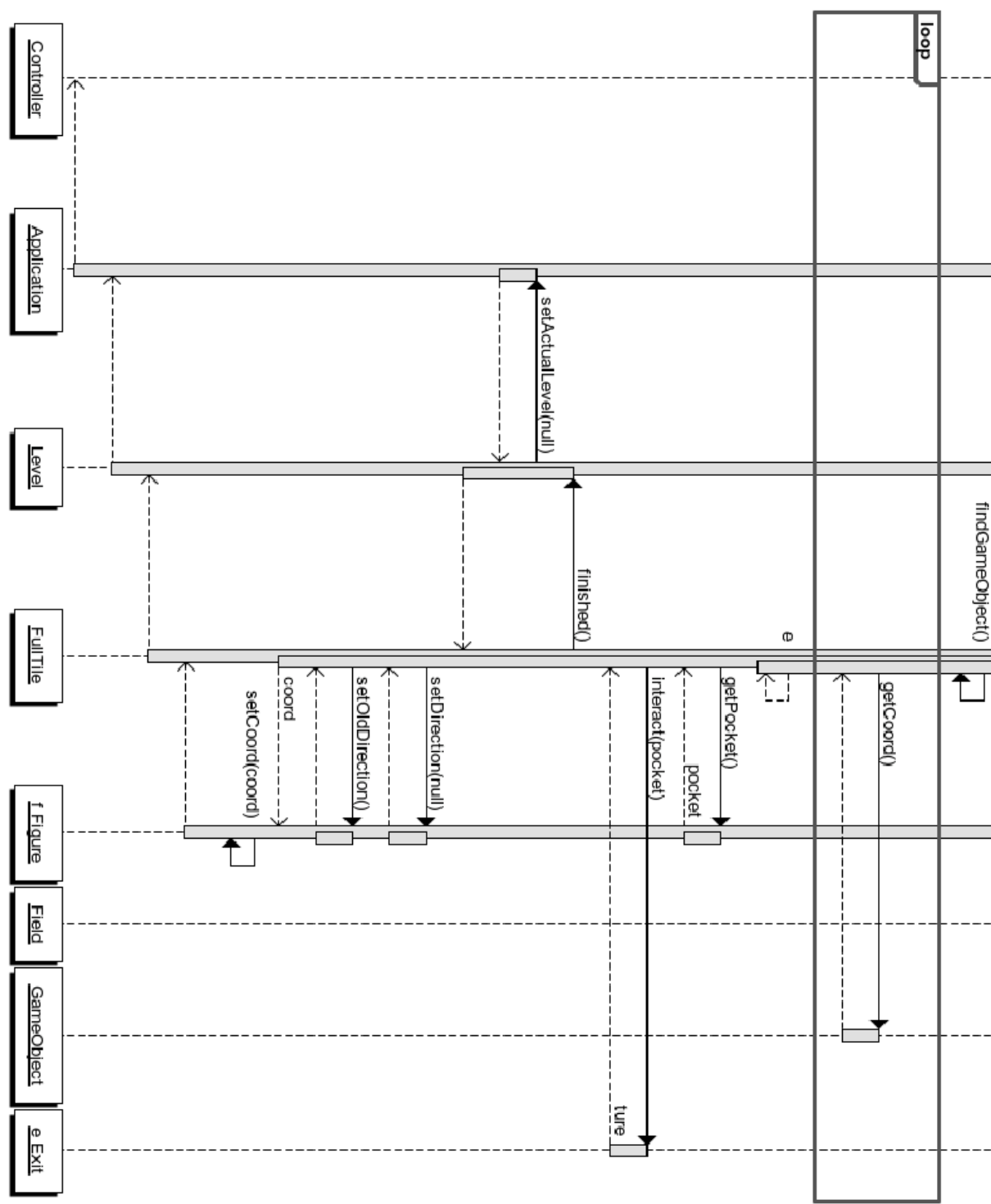
Fut vagy áll a bábu – 1. rész:

Fut vagy áll a bábu – 2. rész:

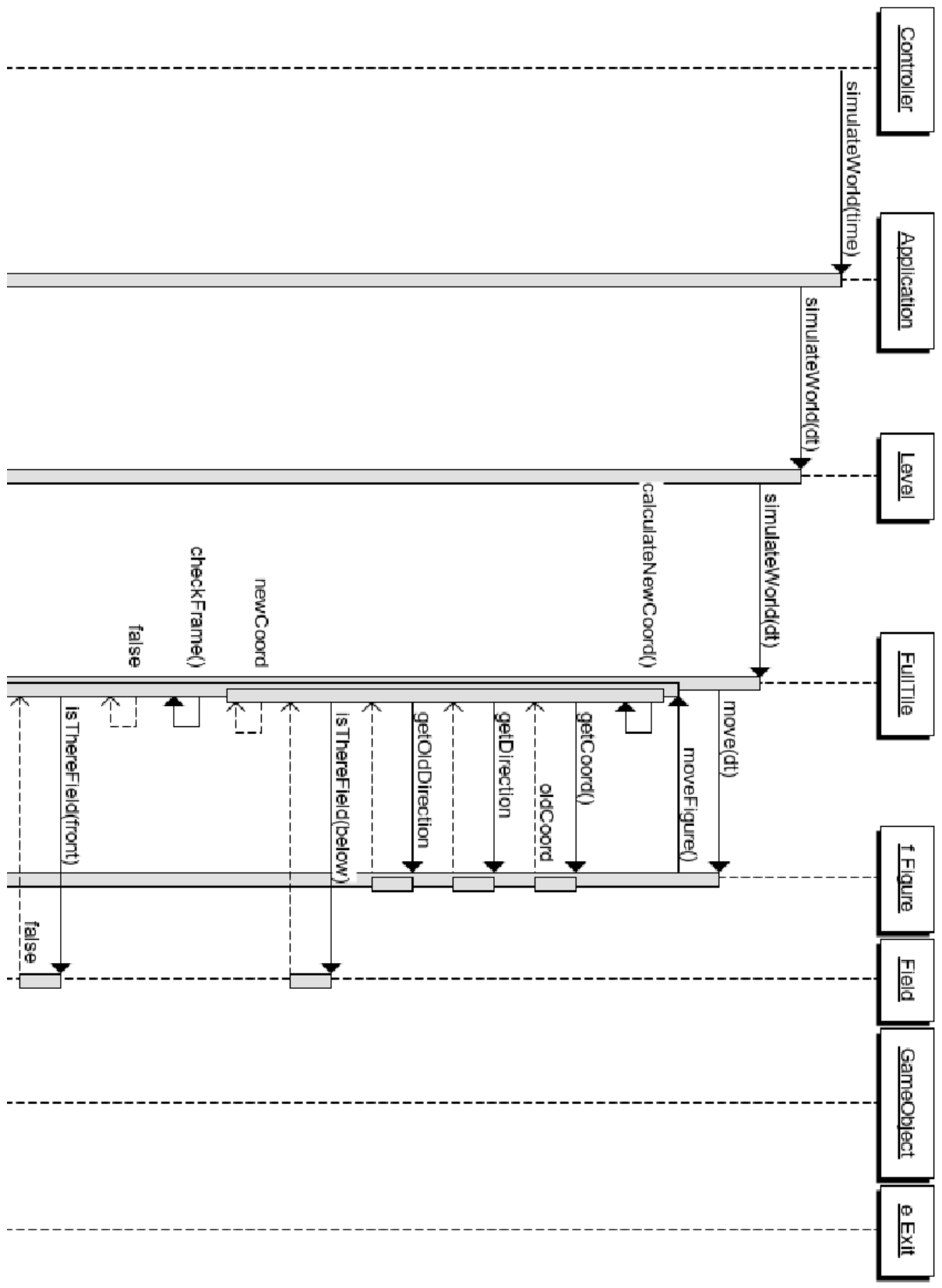


Kijáratot talál a bábu és kész a pálya – 1. rész:

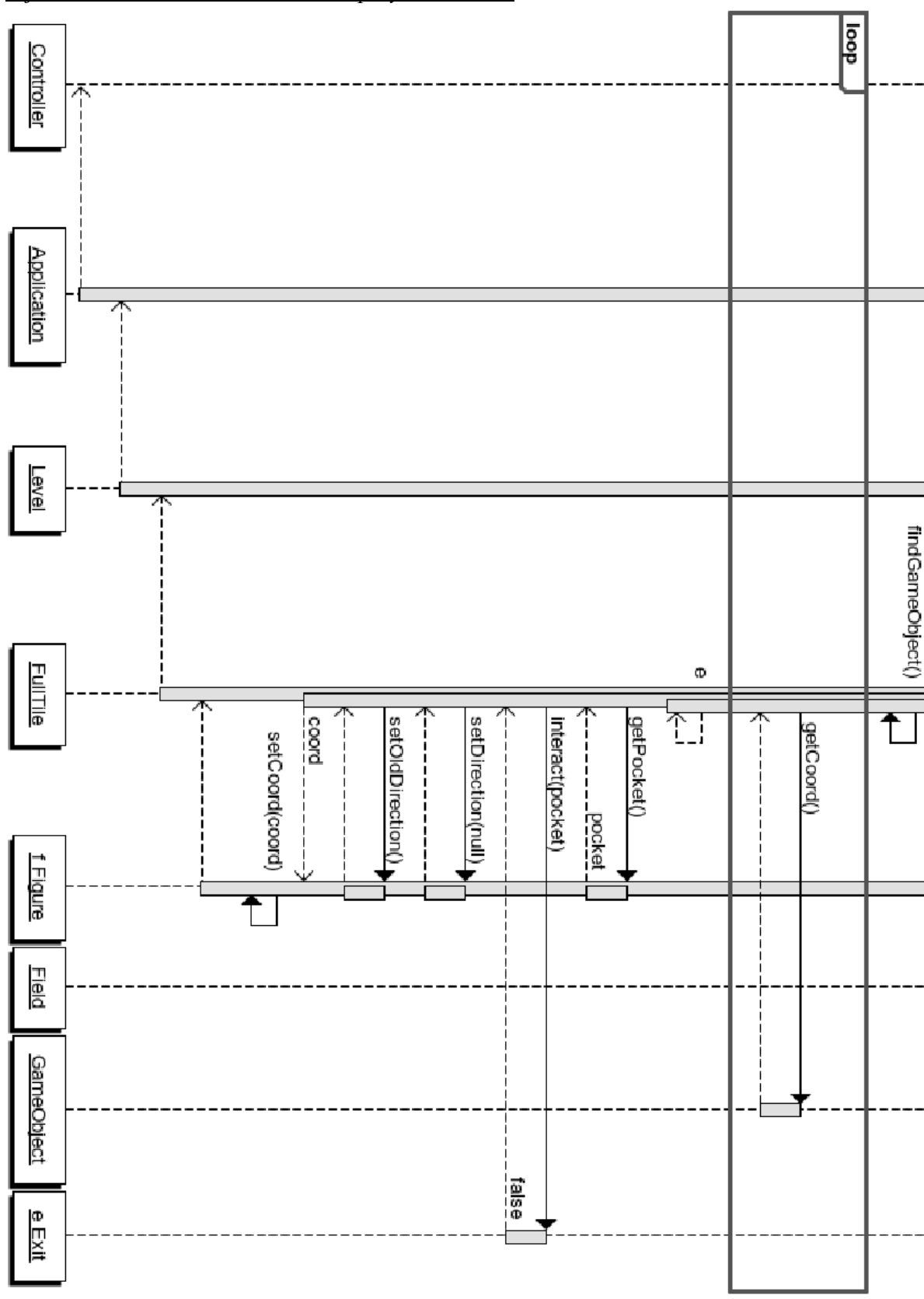
Kijáratot talál a bábu és kész a pálya – 2. rész:

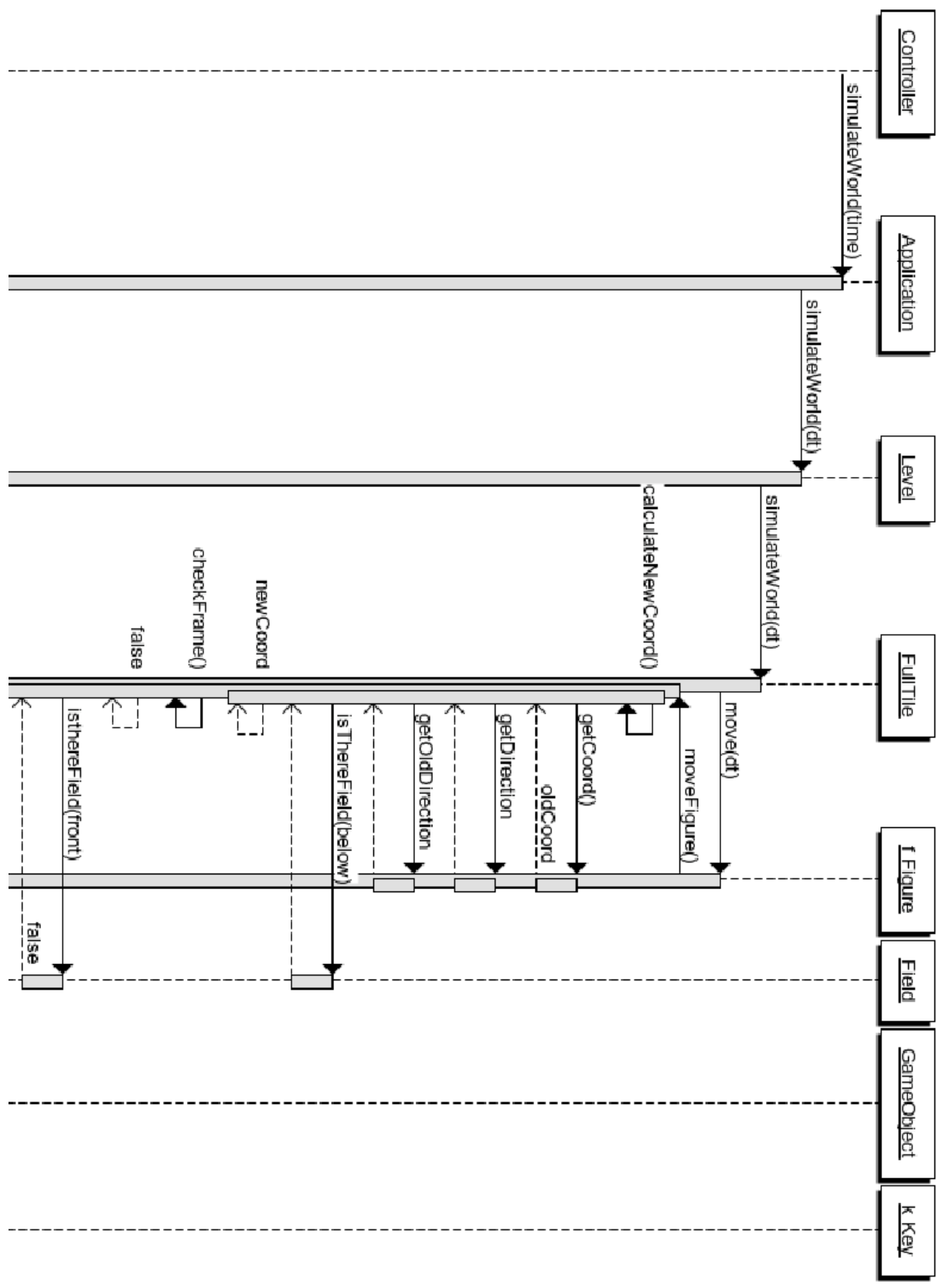


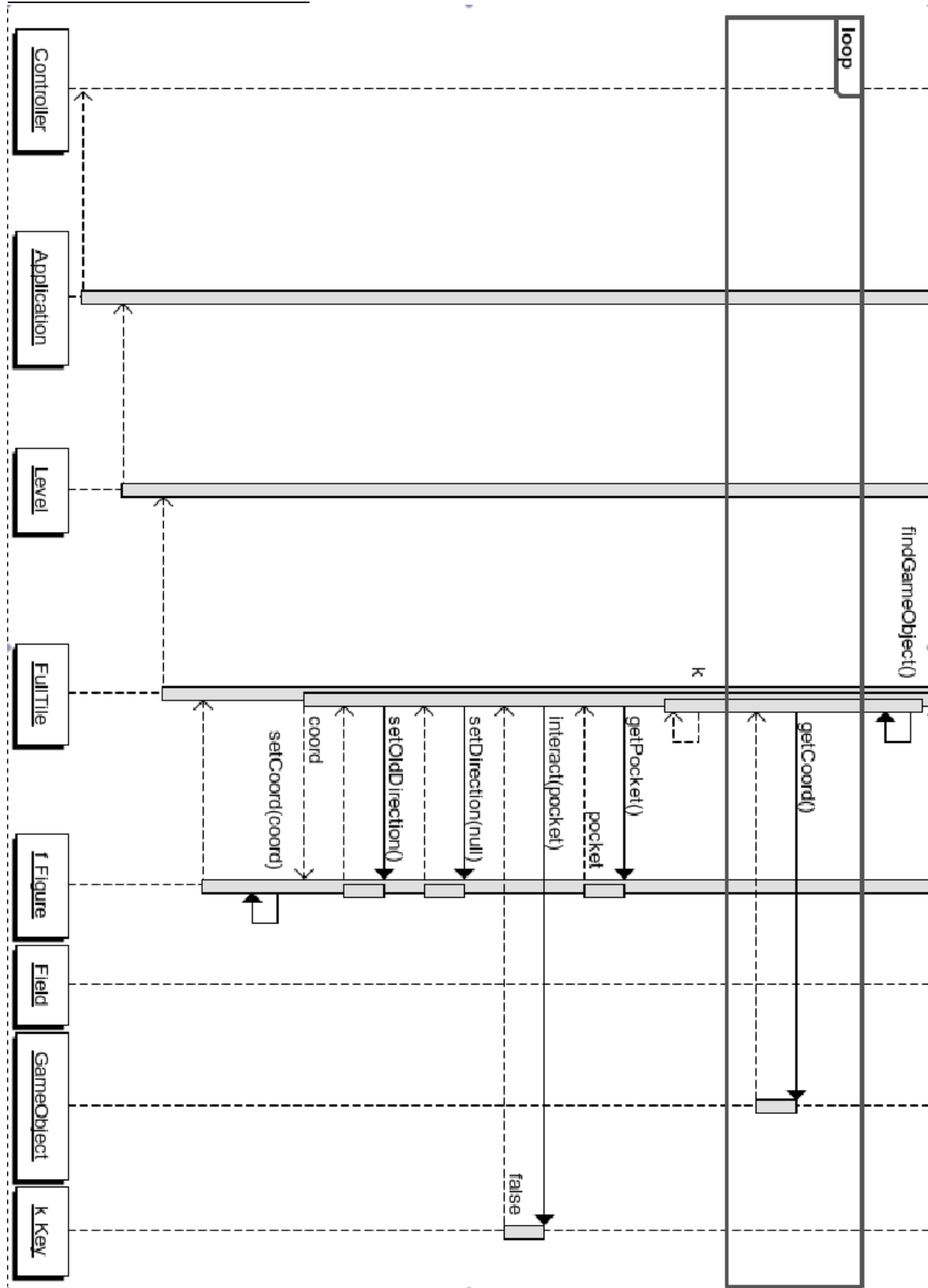
Kijáratot talál a bábu, nincs kész a pálya – 1. rész:



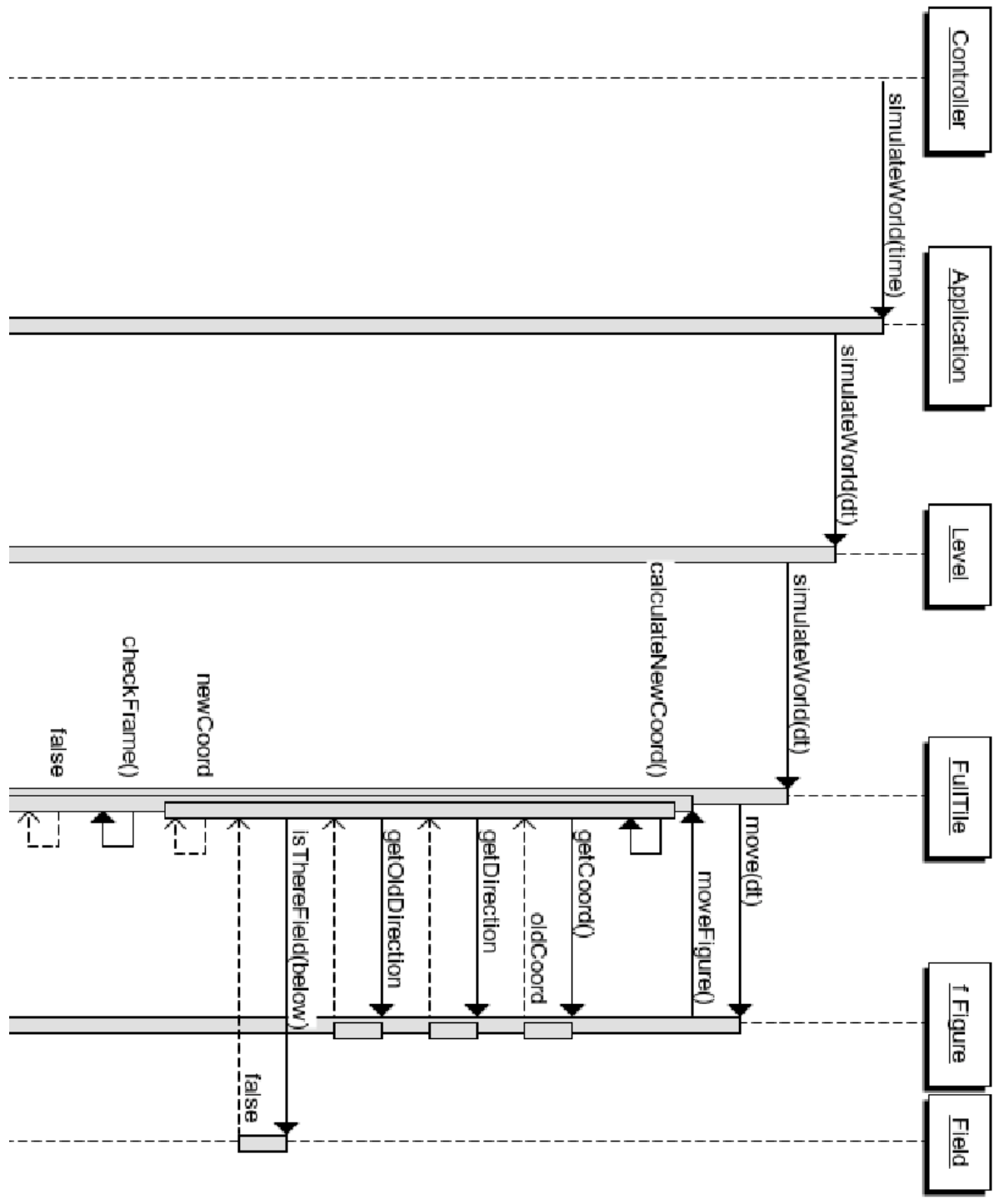
Kijáratot talál a bábu, nincs kész a pálya – 2. rész:



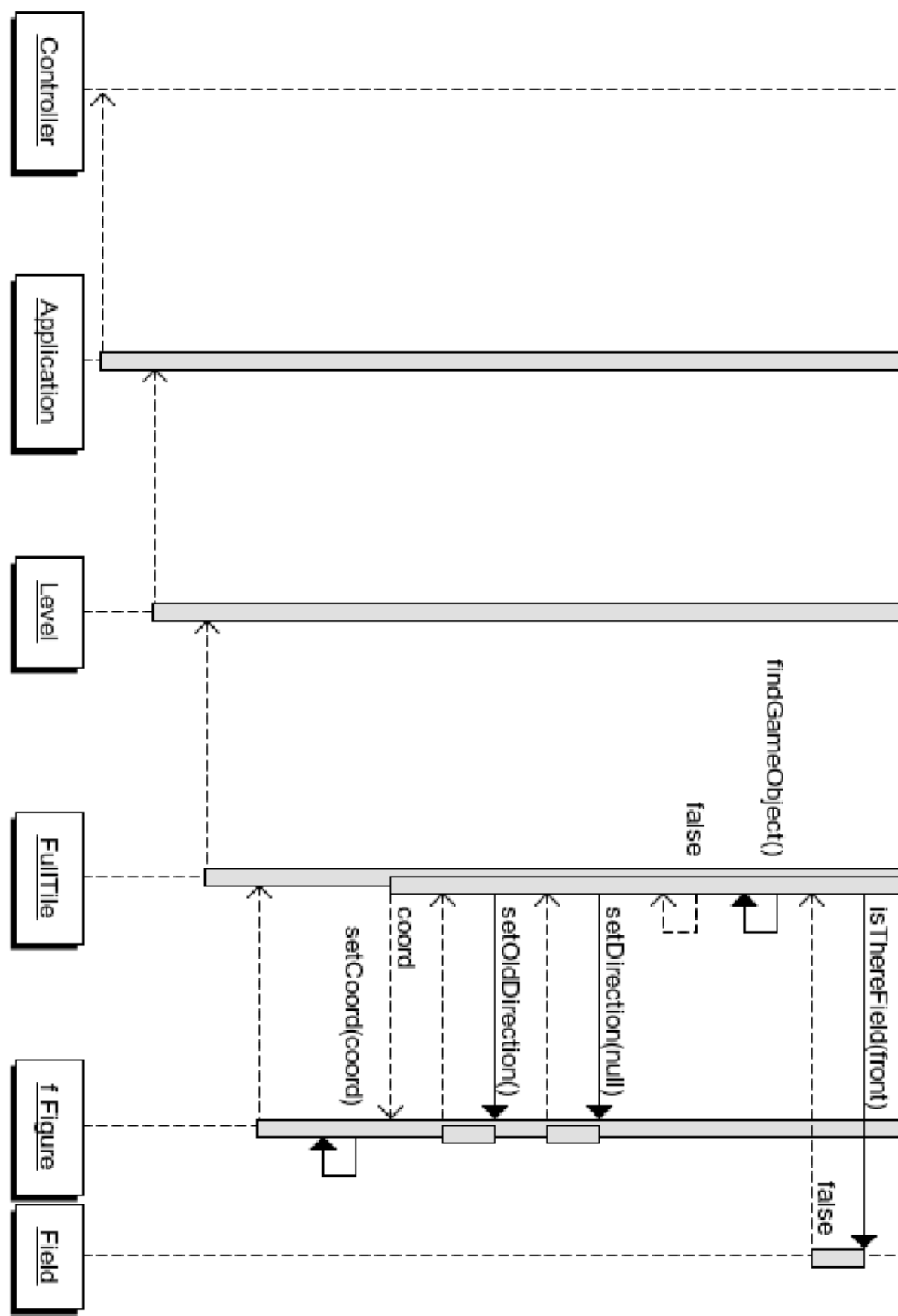
Kulcsot talál a bábu – 1. rész:

Kulcsot talál a bábu – 2. rész:

Zuhan a bábu – 1. rész:



Zuhan a bábu – 2. rész:



4.5 State-chartok

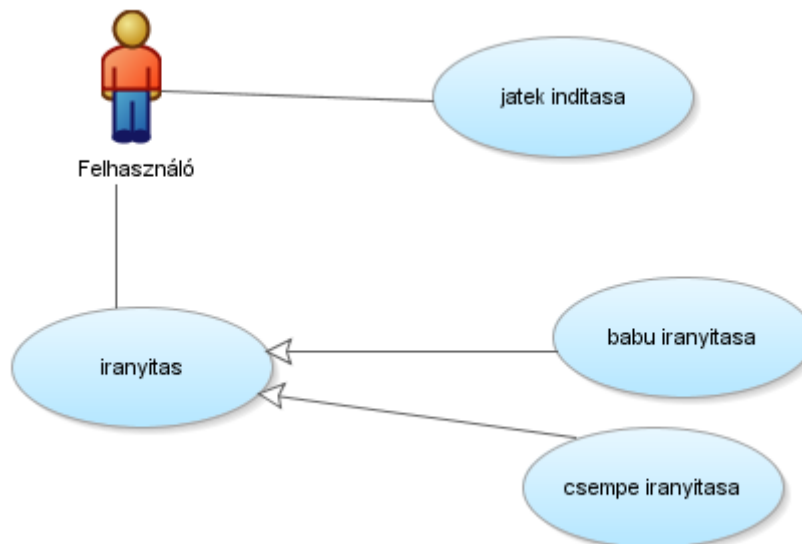
A programban nincs állapotfüggő viselkedés, azaz nincs benne állapotgép.

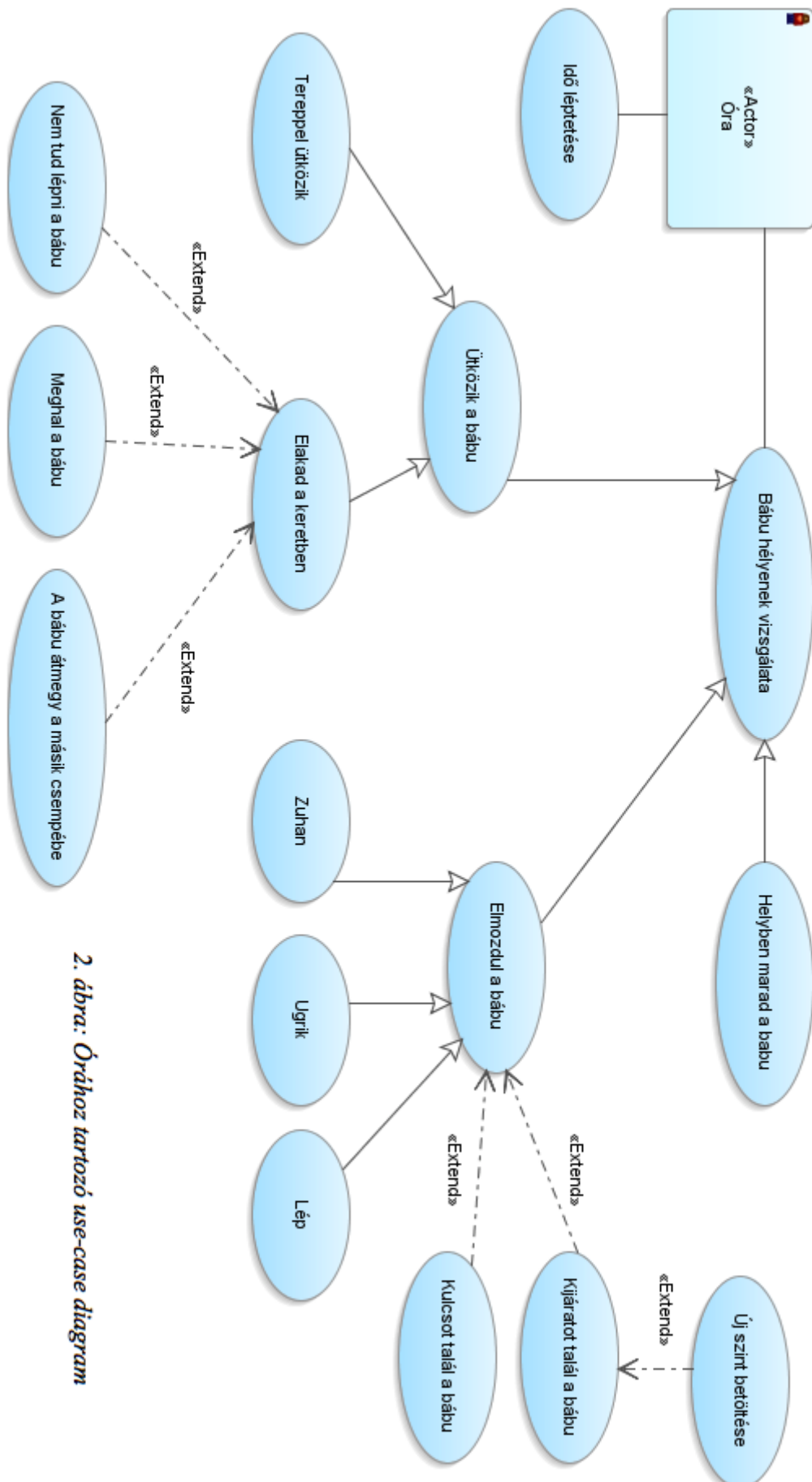
5. Szkeleton tervezése

5.1 A szkeleton modell valóságos use-case-ei

5.1.1 Use-case diagram

1. ábra: Felhasználóhoz tartozó use-case diagram:





2. ábra: Órához tartozó use-case diagram

5.1.2 Use-case leírások

1. ábra Use-case leírásai:

Use-case neve	Játék indítása
Rövid leírás	A felhasználó elindítja a játékot.
Aktorok	Felhasználó
Forgatókönyv	Betöltődik az első pálya.

Use-case neve	Irányítás
Rövid leírás	A felhasználó irányítja a játékot.
Aktorok	Felhasználó
Forgatókönyv	A felhasználó irányítása alapján elmozdul az irányított elem.

Use-case neve	Bábu irányítása
Rövid leírás	A felhasználó a bábút irányítja.
Aktorok	Felhasználó
Forgatókönyv	A felhasználótól kapott mozgásirányt beállítjuk a bábunak.

Use-case neve	Csempe irányítása
Rövid leírás	A felhasználó mozgatja az üres csempét.
Aktorok	Felhasználó
Forgatókönyv	A felhasználó által megadott irányba elmozdul az üres csempe, ha tud arra mozdulni.

2. ábra Use-case leírásai:

Use-case neve	Idő léptetése
Rövid leírás	Az idő telik.
Aktorok	Óra
Forgatókönyv	Az óra számolja a játékban eltelt időt.

Use-case neve	Bábu helyének vizsgálata
Rövid leírás	A bábu elmozdulás utáni helyének vizsgálata.
Aktorok	Óra
Forgatókönyv	Az eltelt idő alapján kiszámolt új koordináta hely vizsgálata, az ott lévő alakzat alapján mozog a bábút.

Use-case neve	Ütközik a bábu
Rövid leírás	A bábu beleütközik valamibe.
Aktorok	Óra
Forgatókönyv	A csempe által kiszámolt koordinátán keret, vagy terep van. A bábu ennek megfelelően megáll, vagy kapcsolatba kerül a kerettel.

Use-case neve	Tereppel ütközik
Rövid leírás	A bábu terepbe ütközik.
Aktorok	Óra
Forgatókönyv	A bábu terepbe ütközik, nem tud tovább menni. Ugyanazon a koordinátán marad.

Use-case neve	Kerettel ütközik a bábu
Rövid leírás	A bábu csempe keretébe ütközik.
Aktorok	Óra
Forgatókönyv	A bábu keretbe ütközik és illeszkedéstől függően vagy átmegy másik csempébe vagy egyhelyben marad vagy meghal.

Use-case neve	Elakad a keretben
Rövid leírás	A bábu a csempe keretébe ütközik, ahol elakad.
Aktorok	Óra
Forgatókönyv	A bábu keretbe ütközik. A csempék keretei nem illeszkednek ezért a bábu nem mozdul el, ugyanott marad.

Use-case neve	Meghal a bábu
Rövid leírás	A bábu a csempe olyan keretébe ütközik, ahol meghal.
Aktorok	Óra
Forgatókönyv	A bábu keretbe ütközik, meghal, majd visszarakjuk a kezdőcsempe kezdőkoordinátájára.

Use-case neve	Átmegy a bábu másik csempébe
Rövid leírás	A bábu csempe keretébe ütközik, és átmegy a másik csempébe.
Aktorok	Óra
Forgatókönyv	A bábu keretbe ütközik, a keretek illeszkednek és átmegy a bábu a másik csempébe, a másik csempe koordináta rendszerének megfelelő helyre kerül, ott folytatja mozgását.

Use-case neve	Helyben marad a bábu
Rövid leírás	A bábu nem akar elmozdulni.
Aktorok	Óra
Forgatókönyv	A bábu nem akar mozdulni, csak a gravitáció hat rá, de talaj van alatta. Ebben az esetben egyhelyben áll a bábu.

Use-case neve	Elmozdul a bábu
Rövid leírás	Megtörténik a bábu elmozdulása.
Aktorok	Óra
Forgatókönyv	A bábu elmozdul a felhasználó által beállított irányba. Az új helyét a csempe számítja ki.

Use-case neve	Kulcsot talál a bábu
Rövid leírás	Megtörténik a bábu kulccsal való interakciója.
Aktorok	Óra
Forgatókönyv	A bábu elmozdul a felhasználó által beállított irányba. Az új helyét a csempe számítja ki, ahol kulcs található.

Use-case neve	Kijáratot talál a bábu
Rövid leírás	Megtörténik a bábu kijáratral való interakciója.
Aktorok	Óra
Forgatókönyv	A bábu elmozdul a felhasználó által beállított irányba. Az új helyét a csempe számítja ki, ahol kijárat található.

Use-case neve	Új szint betöltése
Rövid leírás	A bábu kijáratra lép és teljesíti a szintet.
Aktorok	Óra
Forgatókönyv	A bábunak elég kulcsa van ahhoz hogy teljesítse a szintet, új szint töltődik be.

Use-case neve	Lép
Rövid leírás	A bábu lép egy irányba.
Aktorok	Óra
Forgatókönyv	A bábu lép a felhasználó által beállított irányba. Az új helyét a csempe számítja ki. A bábu alatt talaj van ezért nem zuhan le.

Use-case neve	Ugrik
Rövid leírás	A bábu ugrik.
Aktorok	Óra
Forgatókönyv	A bábu felfelé elmozdul. Az új helyét a csempe számítja ki.

Use-case neve	Zuhan
Rövid leírás	A bábu zuhan
Aktorok	Óra
Forgatókönyv	A bábu zuhan, alatta nincs terep. Az új helyét a csempe számítja ki.

5.2 Architektúra

5.2.1 Szkeleton architektúrális leírása

A szkeleton segítségével egyszerűen tudjuk tesztelni a programunkat, ellenőrizni tudjuk, hogy a szekvencia diagramoknak megfelelő módon hívódnak egymás után a metódusok.

A szkeleton egy olyan egyszálú tesztelés, melyet már akkor el tudunk végezni, mikor még nincs kész a programunk, csak az alap váz. Ehhez az alap vázhoz kellene előre megtervezett és megírt tesztesetek, melyek segítségével be tudjuk mutatni és le tudjuk ellenőrizni az objektumok közötti kapcsolatot és kommunikációt. Próbálunk minden előfordulható esetet reprodukálni. Kizárólag az osztályok közötti kapcsolatok teszteléséről szól a szkeleton, bonyolult metódusok és tényleges számítások nem szerepelnek a programban. A grafikus felület sem fontos ennél a tesztelési fázisnál, konzolos megjelenítést használunk.

5.2.2 Teszteset #1

Azt vizsgáljuk hogy betölt-e az új szint illetve azon elkezdhetünk-e játszani.

Tesztelendő use-case-ek:

- Játék indítása

5.2.3 Teszteset #2

A bábu és a csempe irányítása use-case tesztelésére szolgáló teszteset.

A felhasználótól megkérdezzük hogy mit akar irányítani, majd azt hogy merre.

Tesztelendő use-case-ek:

- Bábu irányítása
- Csempe irányítása

5.2.4 Teszteset #3

Az idő elteltésének, azaz a bábu mozgásának egyik tesztesete.

Pontosabban az az eset, amikor a bábu nem tud elmozdulni, mert beleütközik valamibe. A bábu ebben az esetben nem halhat meg. A felhasználótól megkérdezzük, hogy ott terep vagy keret van-e. Ha keret, akkor megkérdezzük, hogy illeszkednek-e a csempék.

Tesztelendő use-case-ek:

- Tereppel ütközik
- Nem tud lépni a bábu
- A bábu átmegy a másik csempébe

5.2.5 Teszteset #4

Az idő eltelésének, azaz a bábu mozgásának egyik tesztesete.

Pontosabban az az eset, amikor a bábu lefelé akarja elhagyni a csempét. A felhasználótól megkérdezzük, hogy illeszkednek-e a csempék keretei.

Tesztelendő use-case-ek:

- Meghal a bábu
- A bábu átmegy a másik csempébe

5.2.6 Teszteset #5

Az idő eltelésének, azaz a bábu mozgásának egyik tesztesete.

Pontosabban az az eset, amikor a bábu nem akar elmozdulni, csak egy helyben áll és van alatta terep.

Tesztelendő use-case:

- Helyben marad a bábu

5.2.7 Teszteset #6

Az idő eltelésének, azaz a bábu mozgásának egyik tesztesete.

Pontosabban az az eset, amikor a bábu tud lépni, ugrani vagy zuhanni egy adott helyre. Ott megkérdezzük a felhasználótól, hogy van-e alatta talaj.

Tesztelendő use-case-ek:

- Zuhan
- Lép
- Ugrik

5.2.8 Teszteset #7

Az idő eltelésének, azaz a bábu mozgásának egyik tesztesete.

Pontosabban az az eset, amikor a bábu kulcsot vagy kijáratot talál. Megkérdezzük a felhasználótól, hogy kulcs vagy kijárat van ott ahova elmozdul. Ha kijárat, akkor megkérdezzük, hogy a szint betöltéséhez van-e elég kulcsa.

Tesztelendő use-case-ek:

- Kulcsot talál a bábu
- Kijáratot talál a bábu
- Új szint betöltése.

5.3 A szkeleton kezelői felületének terve, dialógusok

A szkeleton egy program váz, melyben minden objektum megtalálható, de azoknak csak az interfésze van megírva.

Több tesztet futtatására kell alkalmasnak lennie a szkeletonnak. A szkeleton indulásakor megadható hogy hányas számú tesztet akarjuk futtatni. A tesztesetek számozása megegyezik az 5.2 Architektúra rész kifejtésében szereplő tesztet számozással.

Minden metódusban csak annyit csinálunk, hogy kiírjuk a metódus nevét és paramétereit, majd meghívja azokat a metódusokat amik az adott használati esethez szükségesek. Ez alapján ellenőrizhetőek a szekvencia-diagramok.

A garantált egyszerűség érdekében csak karakteres képernyőkezelést használunk a grafikus felület helyett. A jól követhető futás érdekében egységes dialógus konvenciót követ a szkeleton.

- minden metódushívást kiírunk, annak paramétereivel és azok típusával együtt, illetve azt hogy melyik objektum metódusát hívtuk
- metódushívást '-->' jellel, metódus visszatérését '<--' jellel jelöljük
- ha egy metódus másik metódust hív, azt behúzással jelöljük
- a felhasználónak szóló kérdéseket '[?]' jellel jelöljük és megadjuk a lehetséges válaszokat is (ha nem megfelelő választ adott, akkor újra feltesszük a kérdést)

Példa:

```
-->Figure.move(int:dt)
  -->FullTile.moveFigure()
    -->FullTile.calculateNewCoord()
      -->Figure.getCoord()
        <--Point:oldCoord
      -->Figure.getDirection()
        <--Direction:dir
      -->Figure.getOldDirection ()
        <--Point:oldDir
      -->Field.isThereField(Point:below)
        [?] Talaj van-e a bábu alatt? [igen/nem] : nem
        <--bool:false
      <--Point:newCoord
    <--Point:coord
  <--void
```

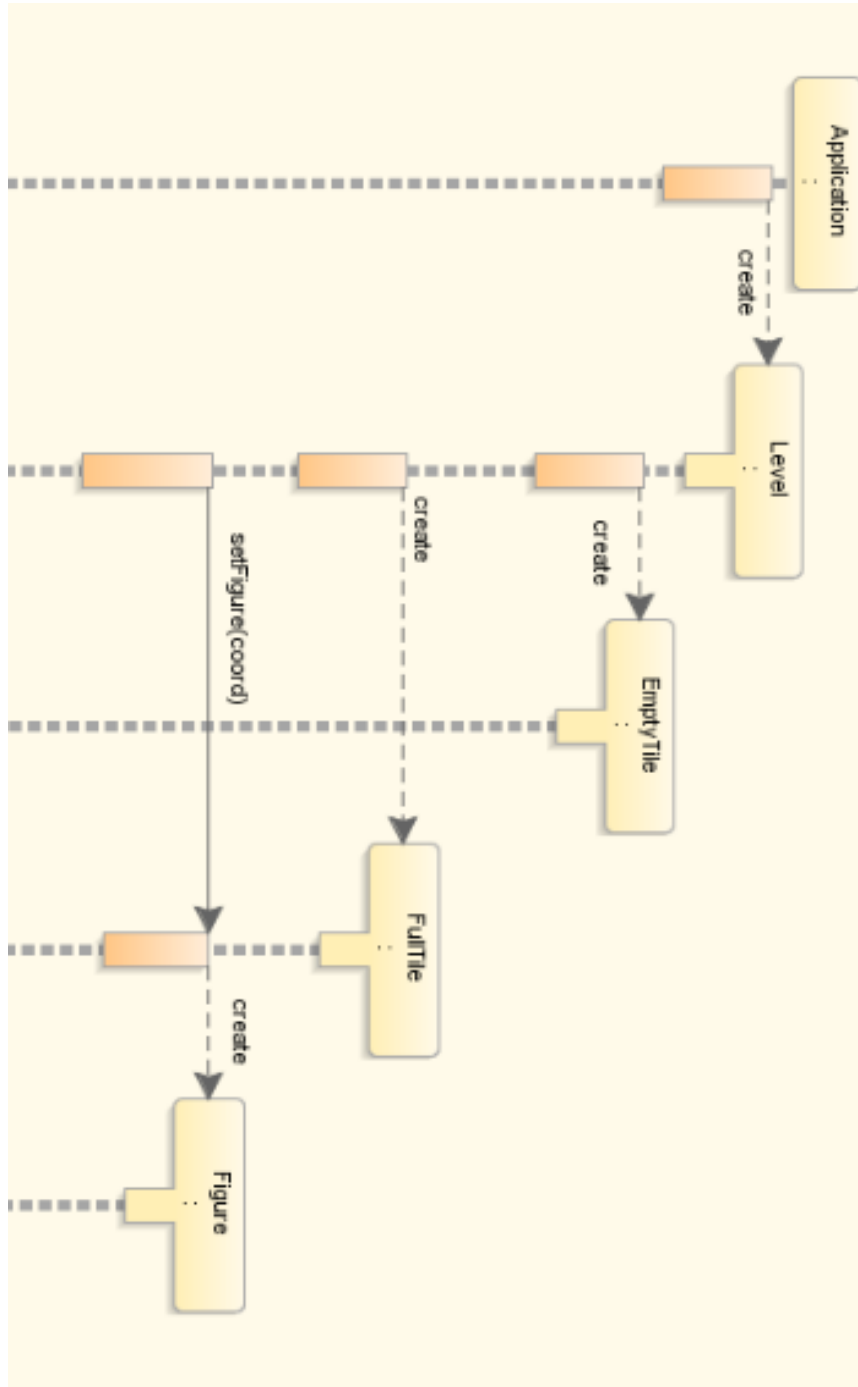

5.4 Szekvencia diagramok a belső működésre

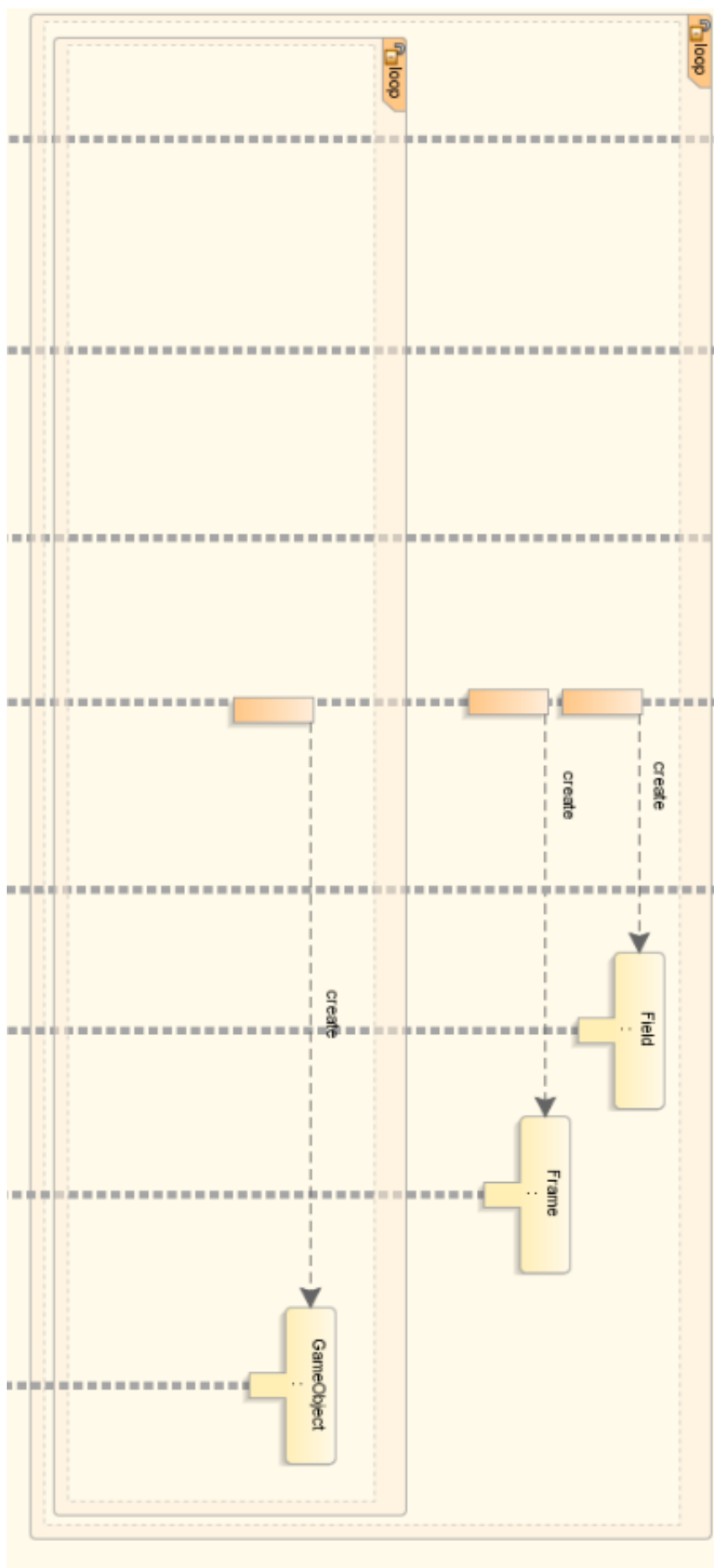
Minden egyes tesztesethez egy vagy több szekvenciadiagram tartozik. A felhasználói döntés miatt tartozhat több szekvenciadiagram egy tesztesethez, illetve azért hogy egyszerűsítsük a szekvenciadiagramokat.

5.4.1 Teszteset #1

Tesztelendő use-case-ek:

- Játék indítása



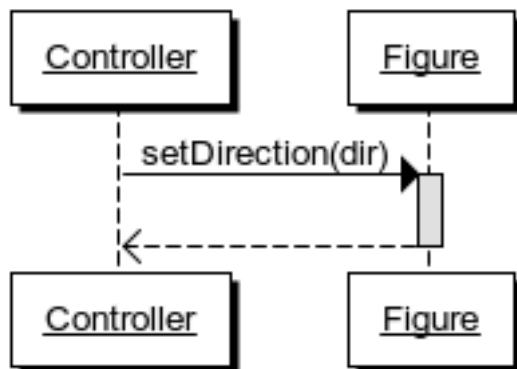


5.4.2 Teszteset #2

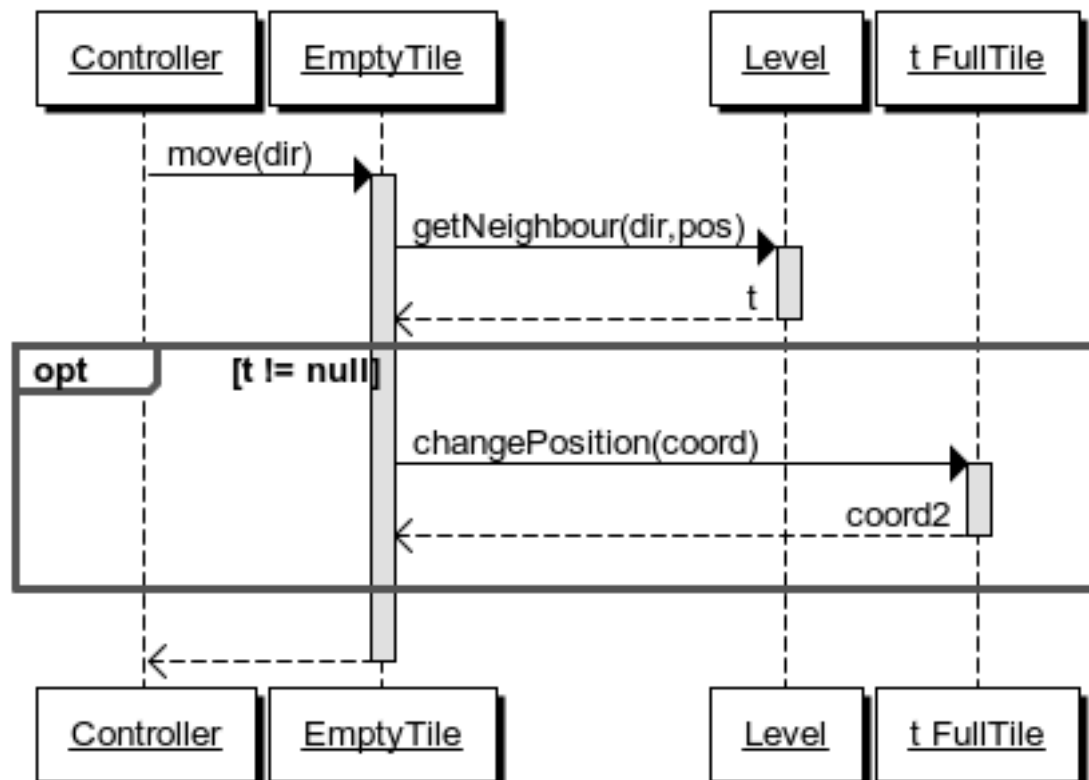
Tesztelendő use-case-ek:

- Bábu irányítása
- Csempe irányítása

Bábu mozgásirányának beállítása.



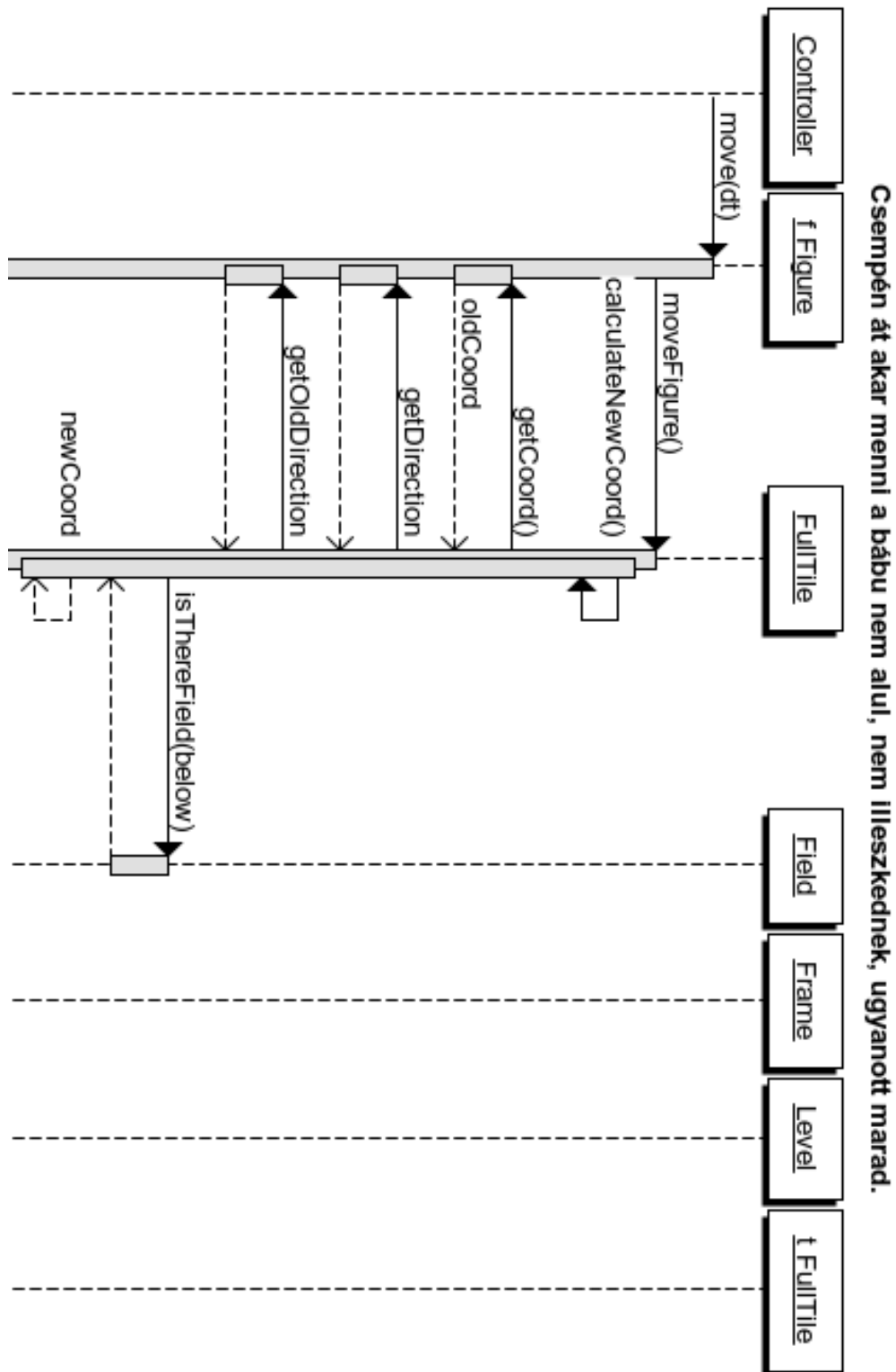
Csempe tolása sikeres/sikertelen.



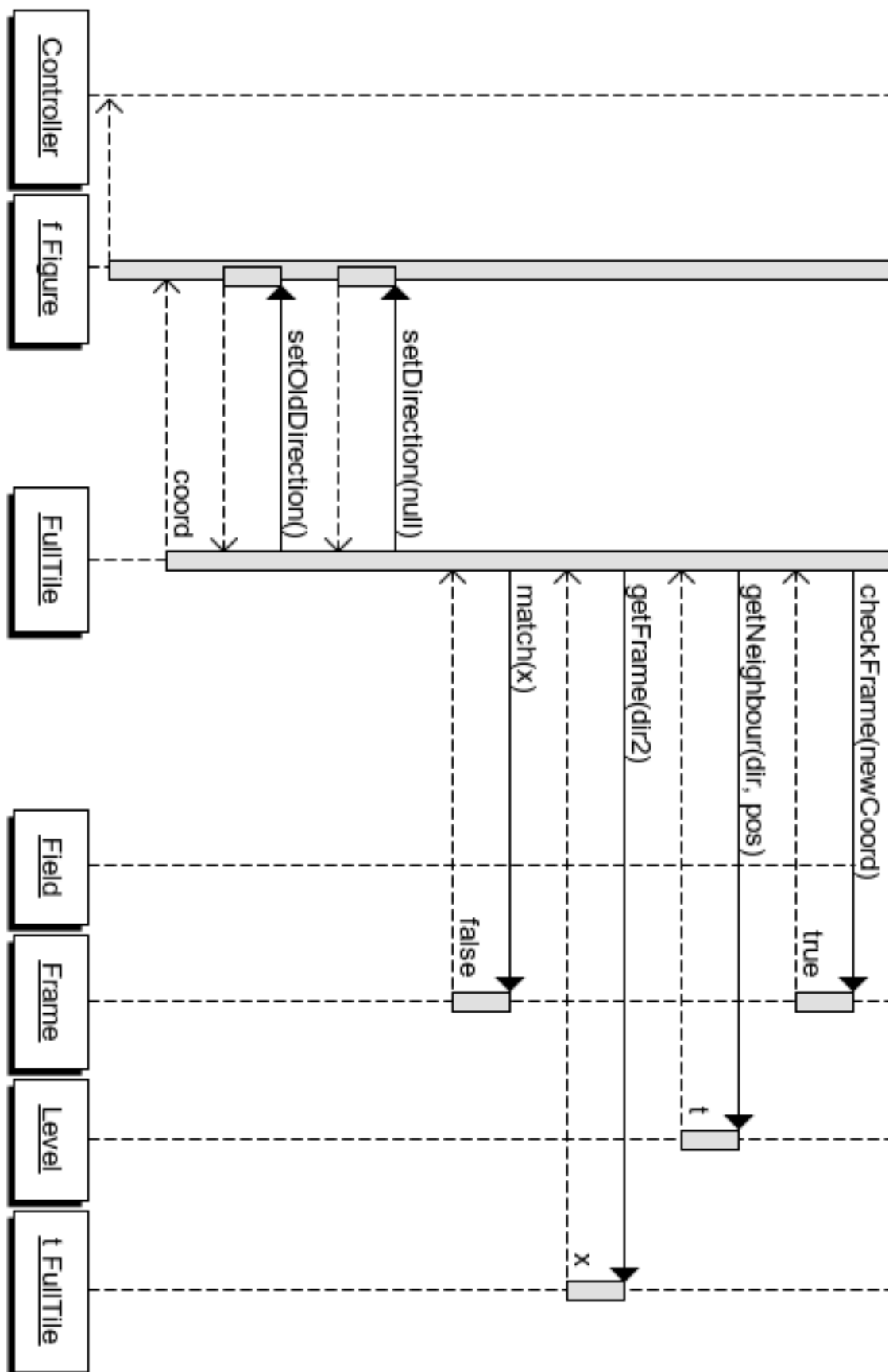
5.4.3 Teszteset #3

Tesztelendő use-case-ek:

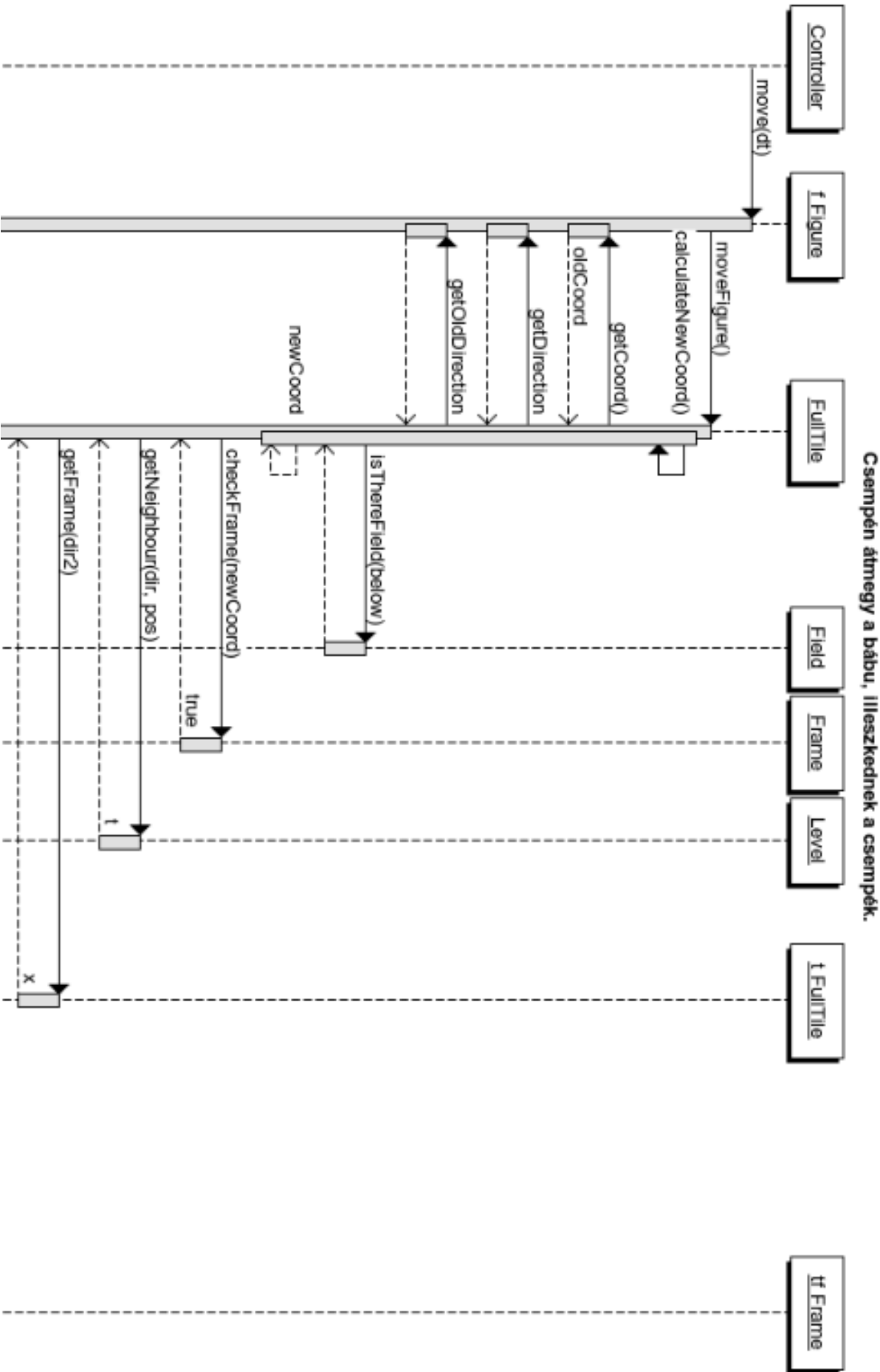
- Tereppel ütközik
- Nem tud lépni a bábu
- A bábu átmegy a másik csempébe

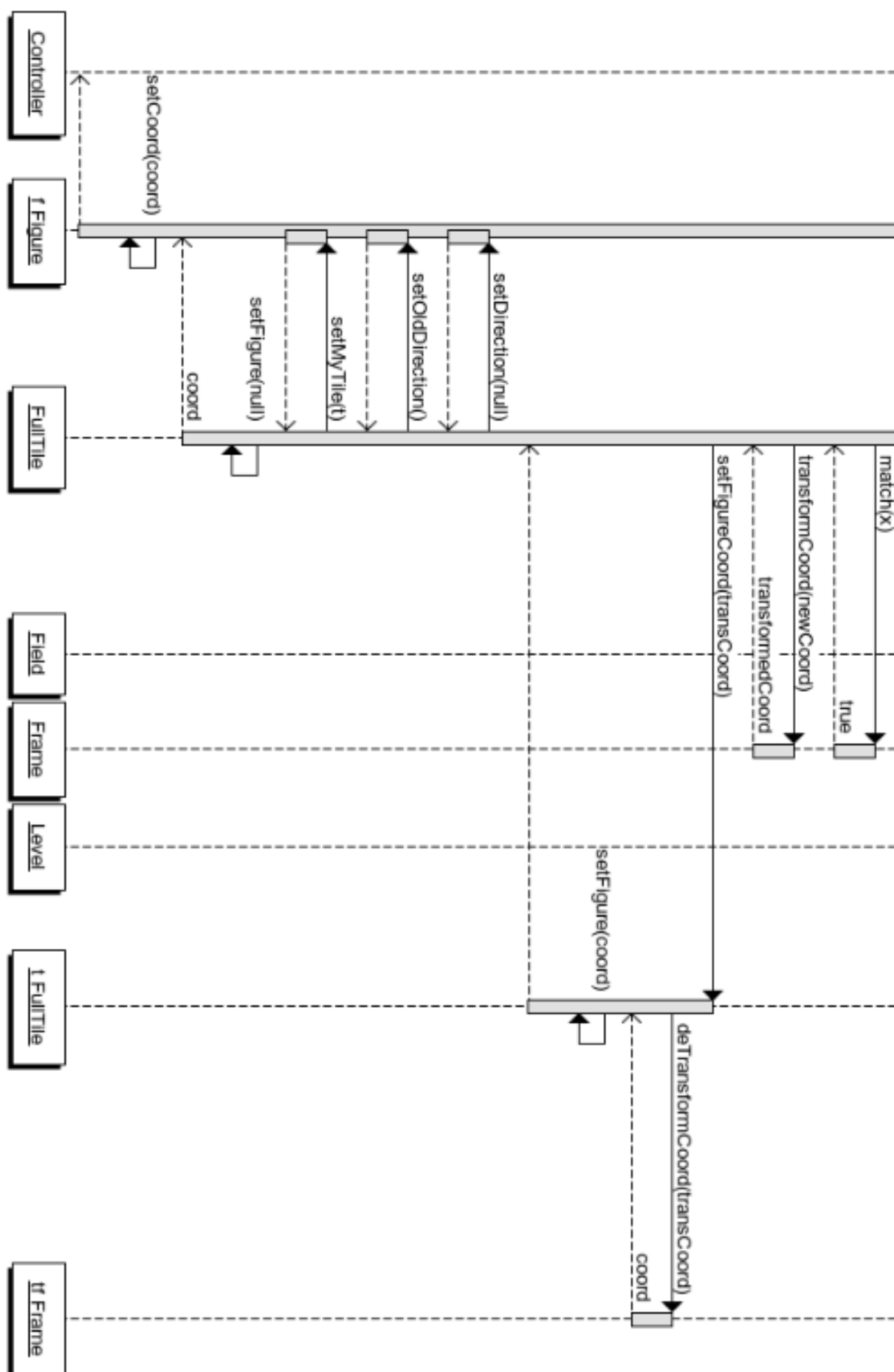


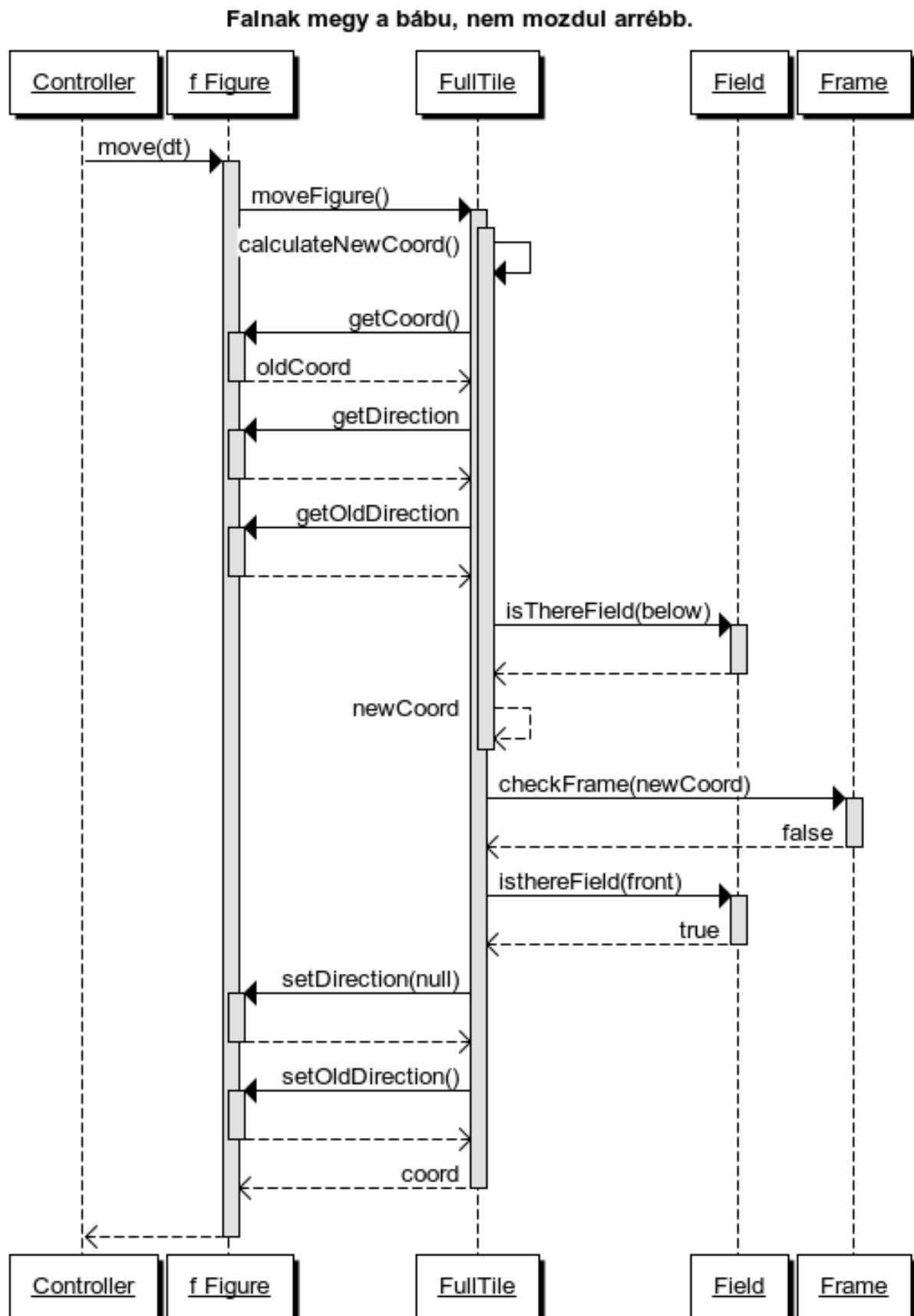
•



•



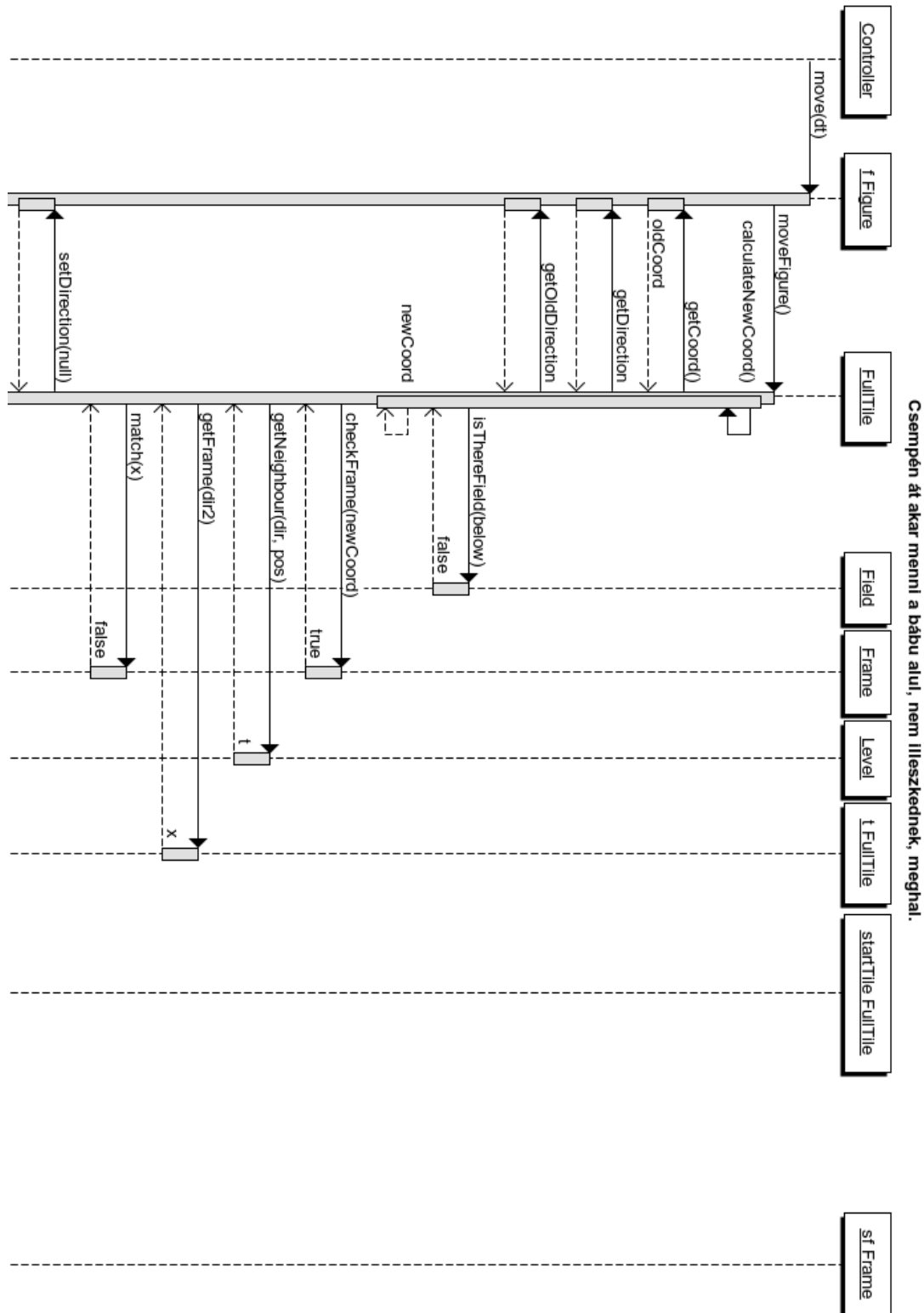


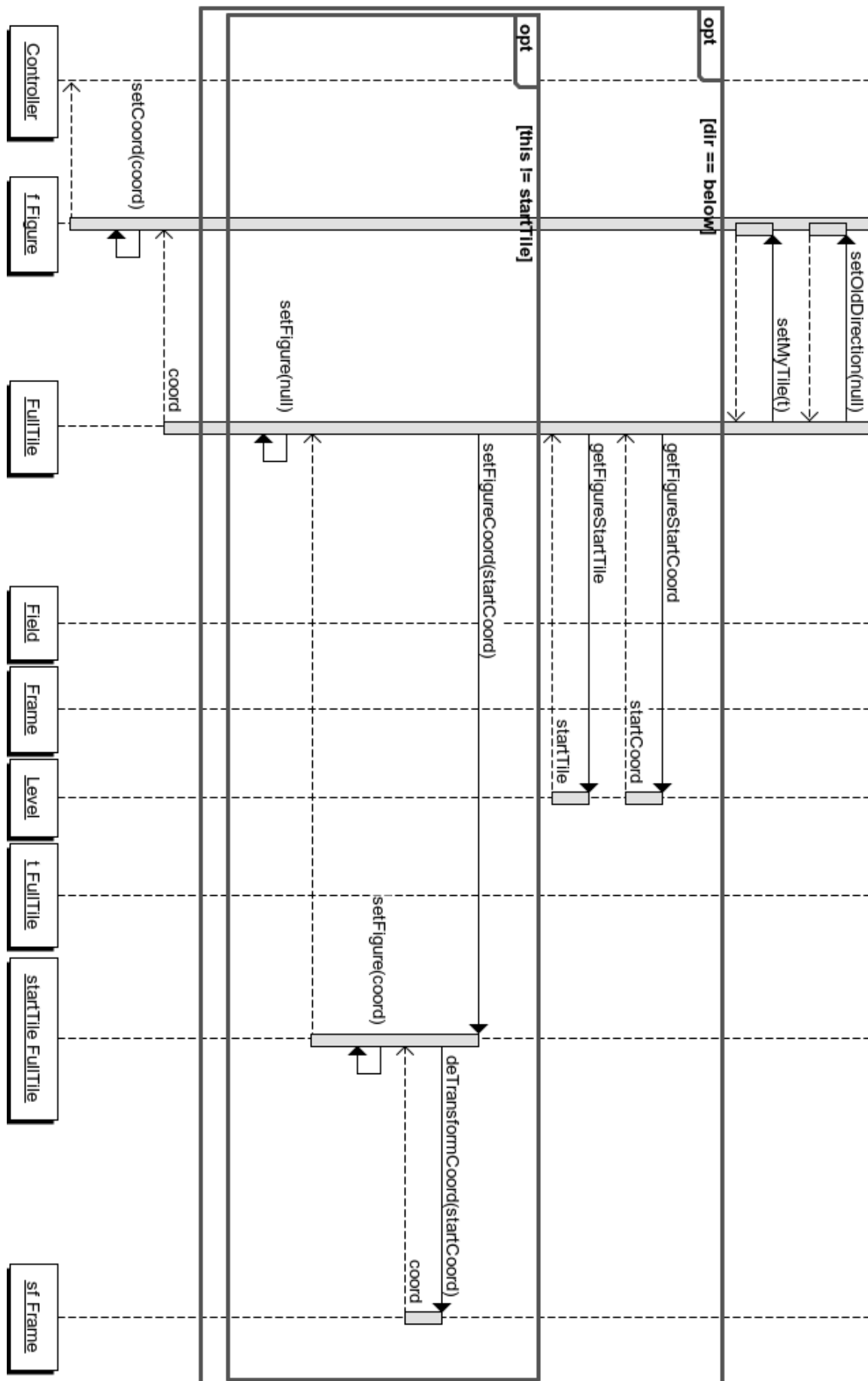


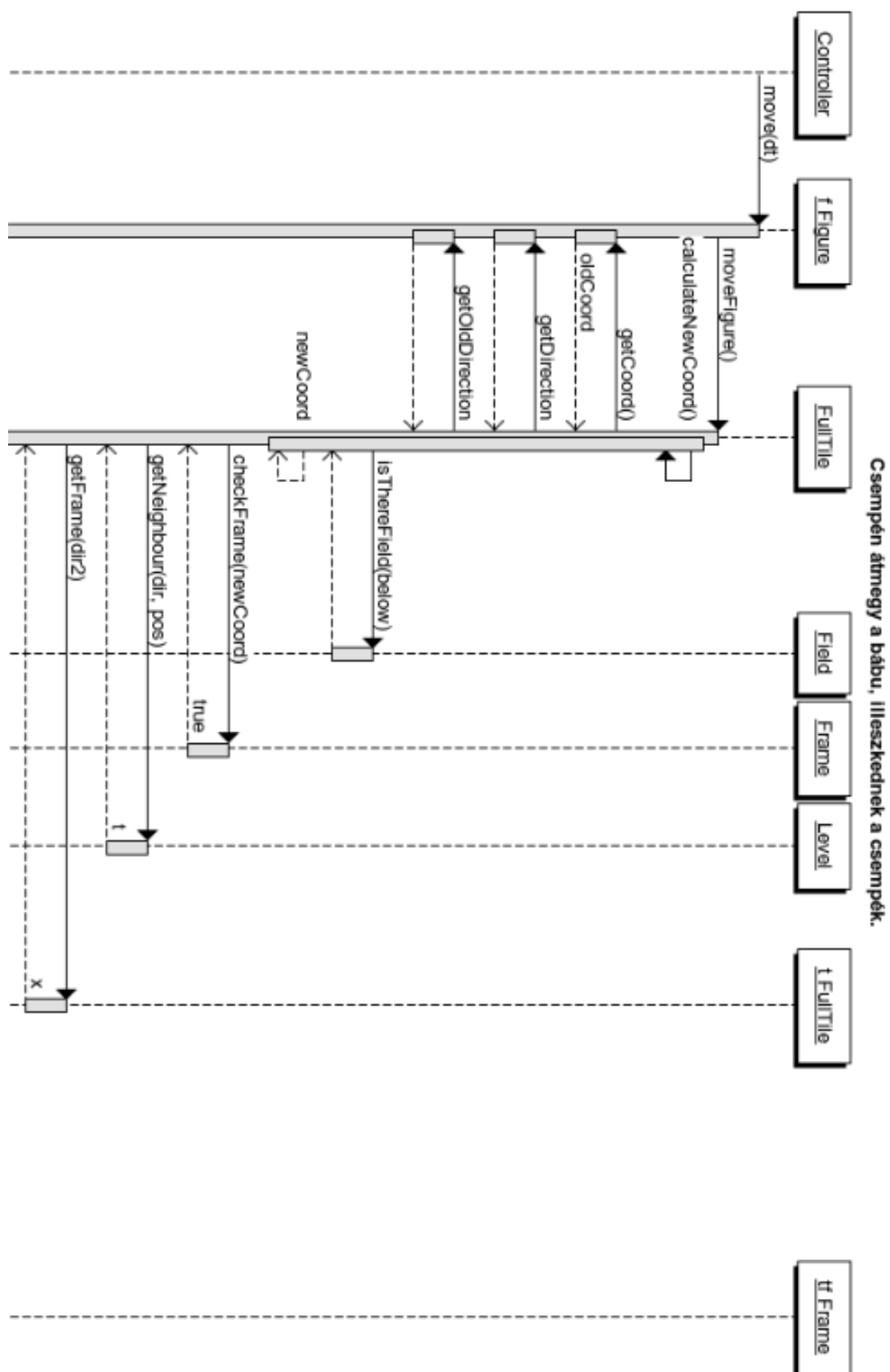
5.4.4 Teszteset #4

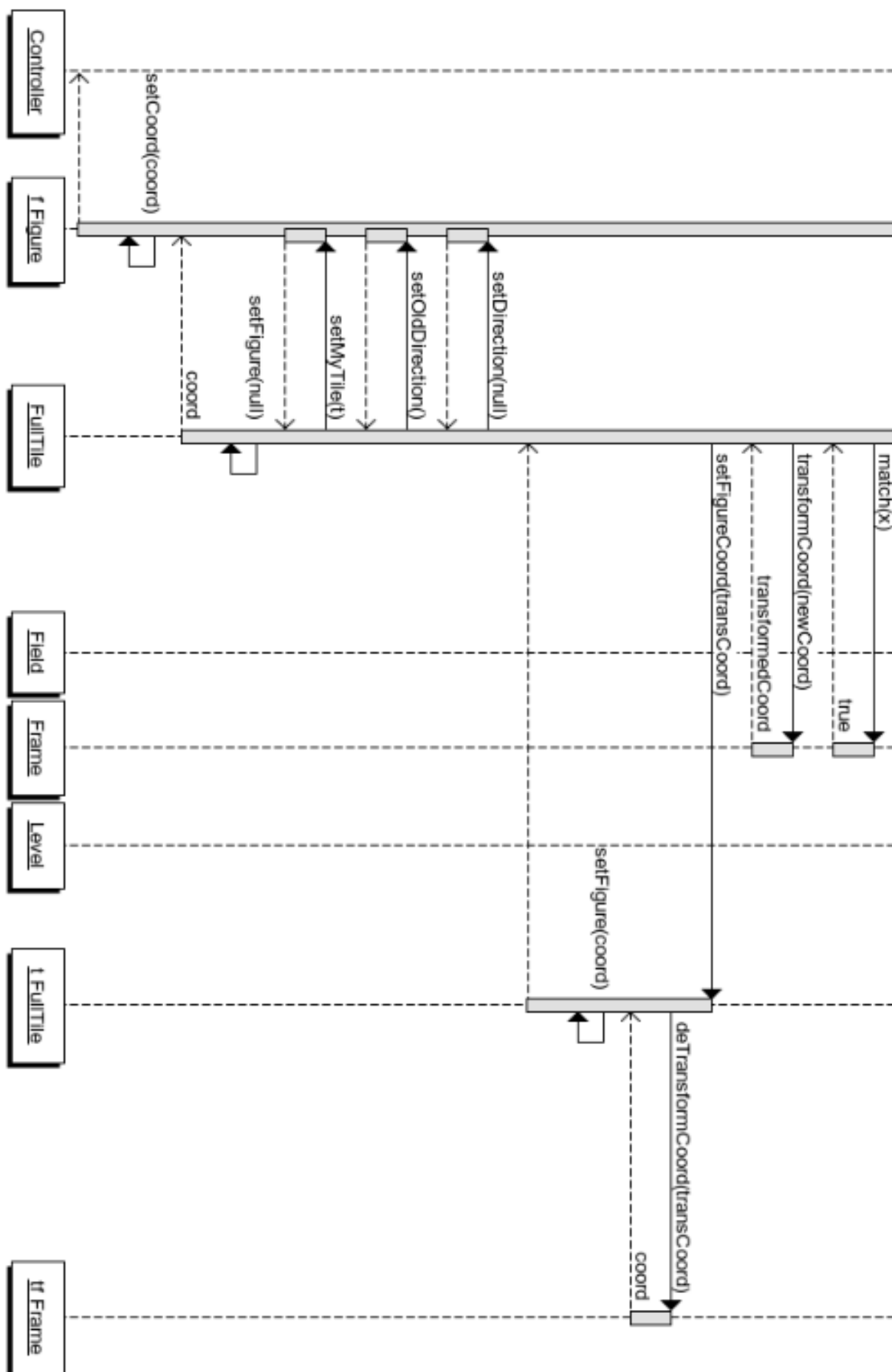
Tesztelendő use-case-ek:

- Meghal a bábú
- A bábú átmegy a másik csempébe







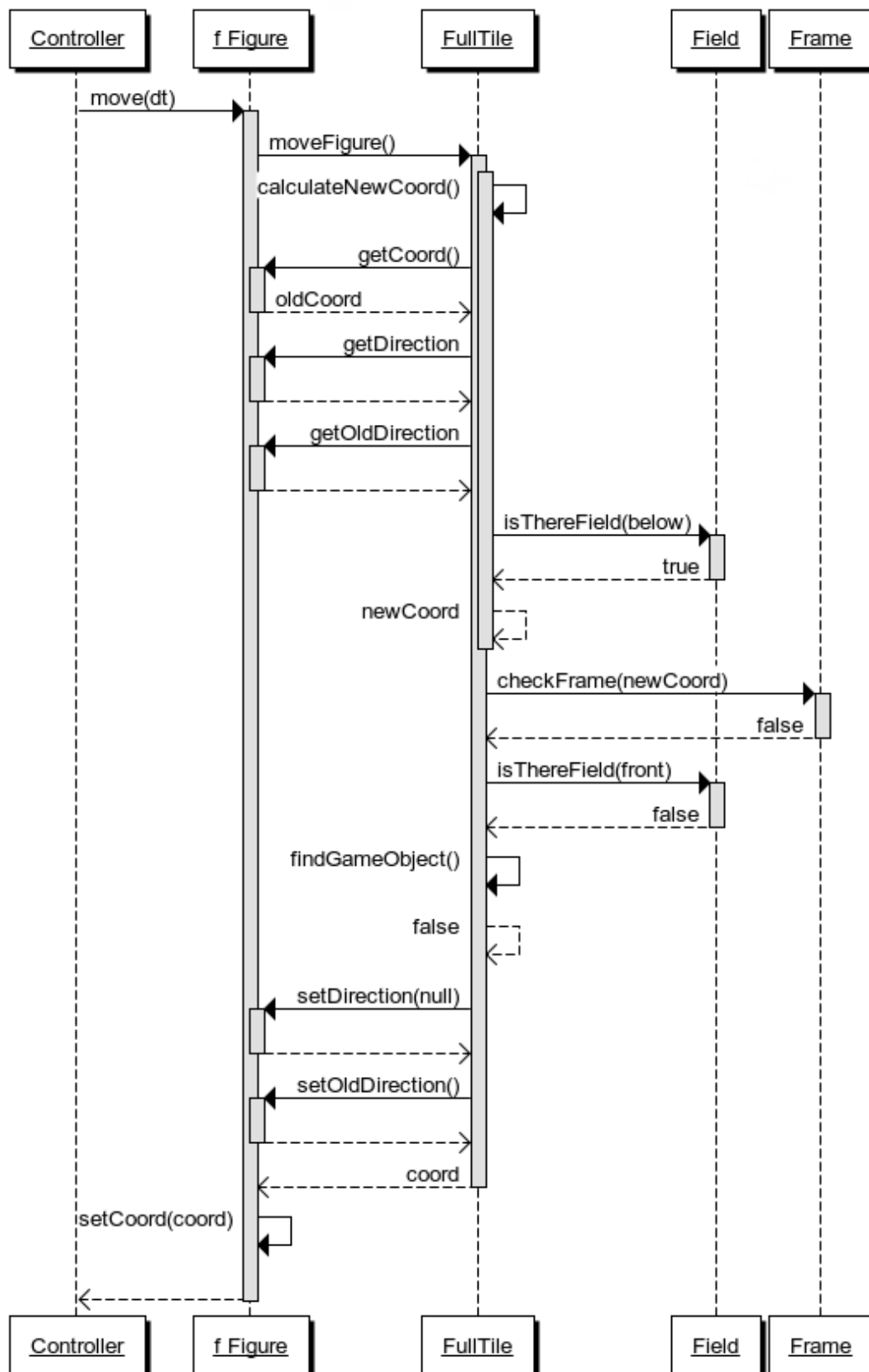


5.4.5 Teszteset #5

Tesztelendő use-case:

- Helyben marad a bábu

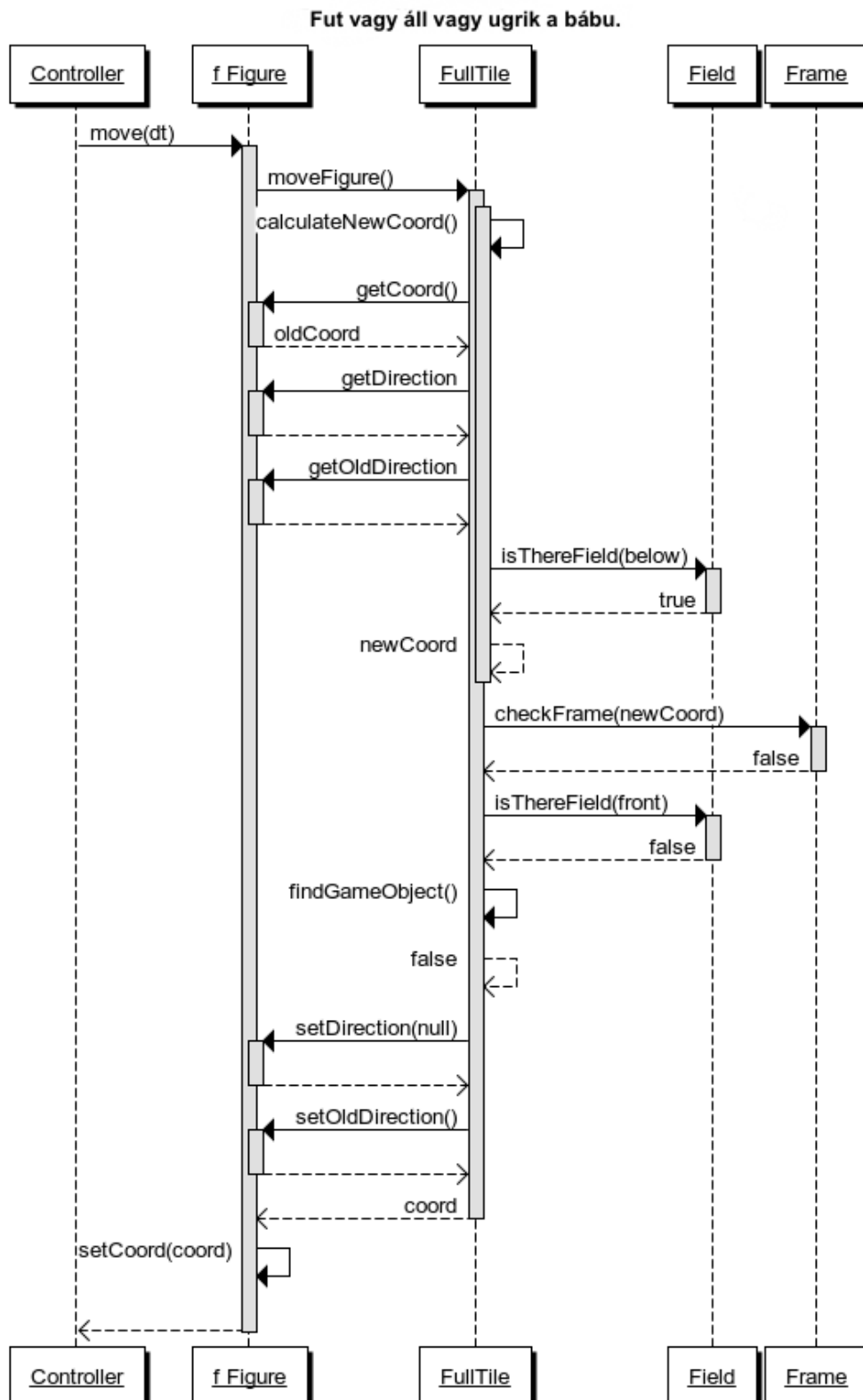
Fut vagy áll vagy ugrik a bábu.

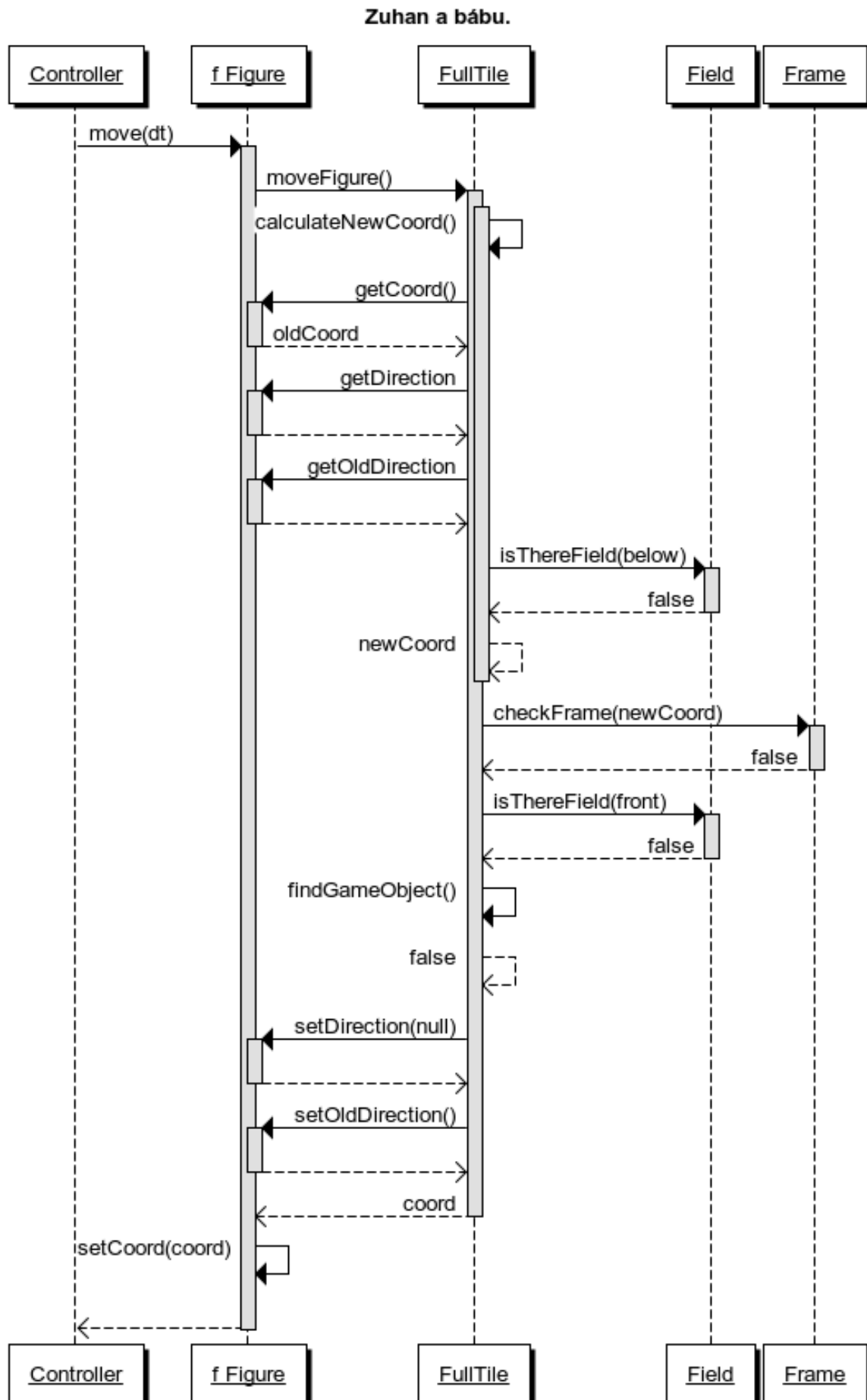


5.4.6 Teszteset #6

Tesztelendő use-case-ek:

- Zuhan
- Lép
- Ugrik

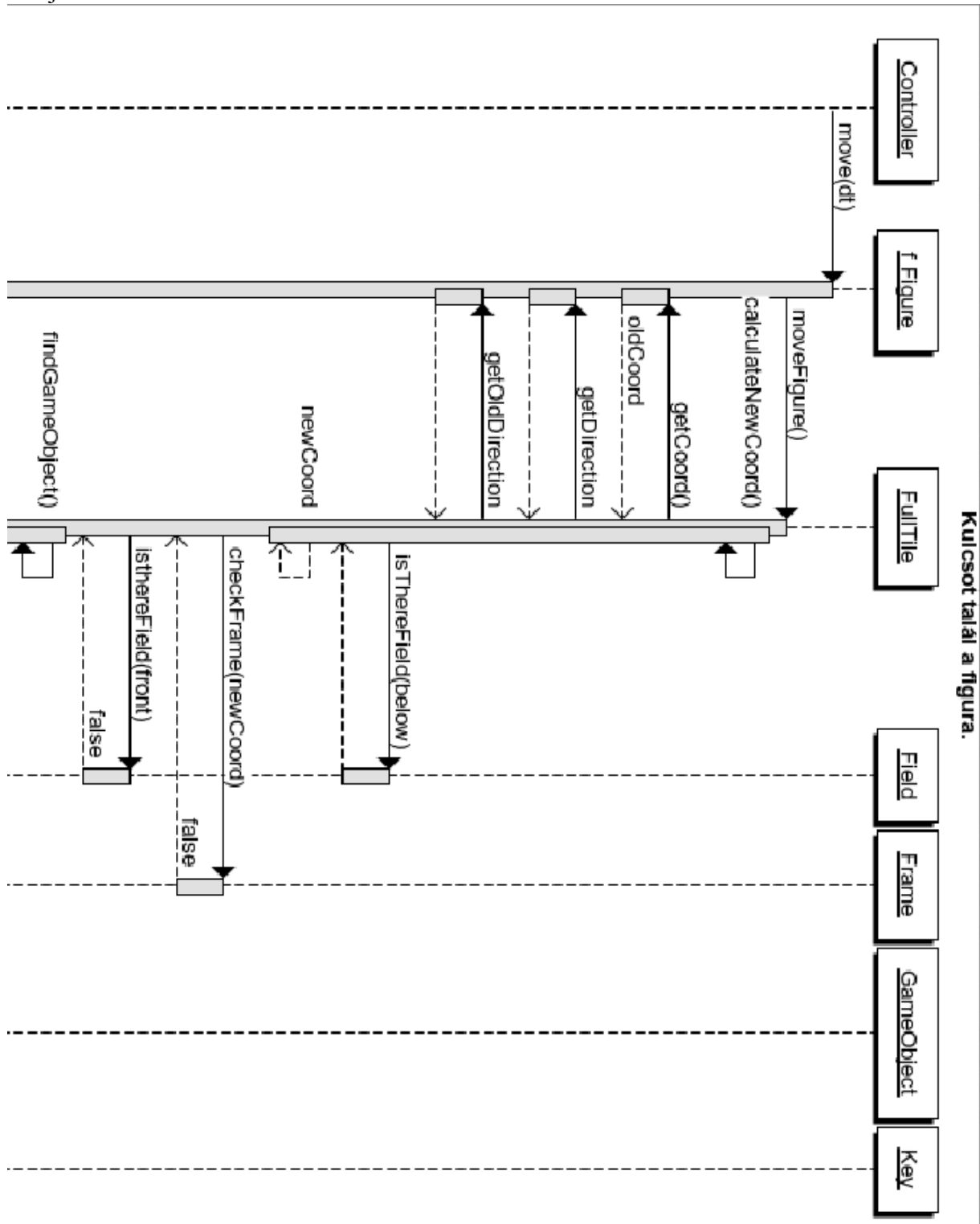


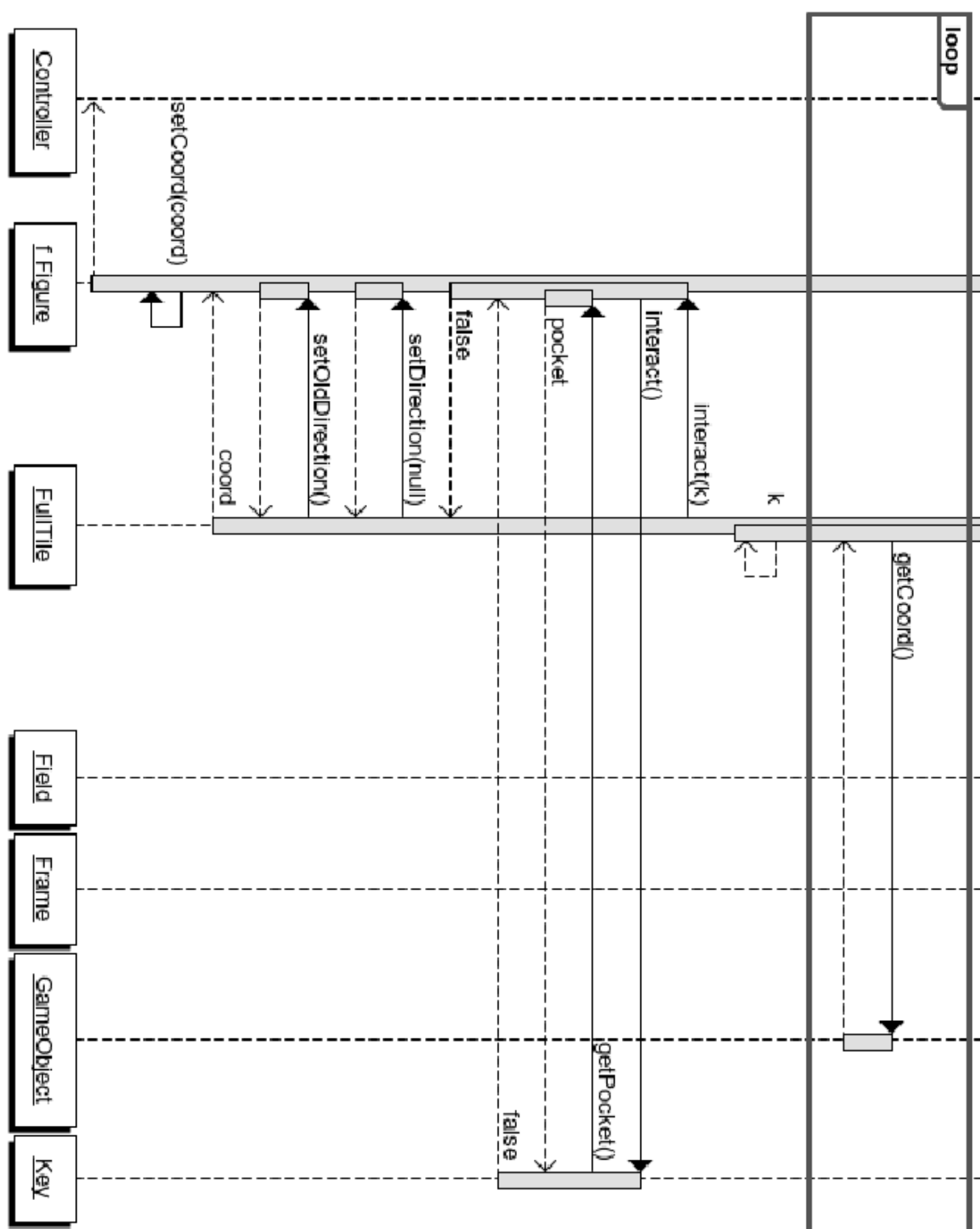


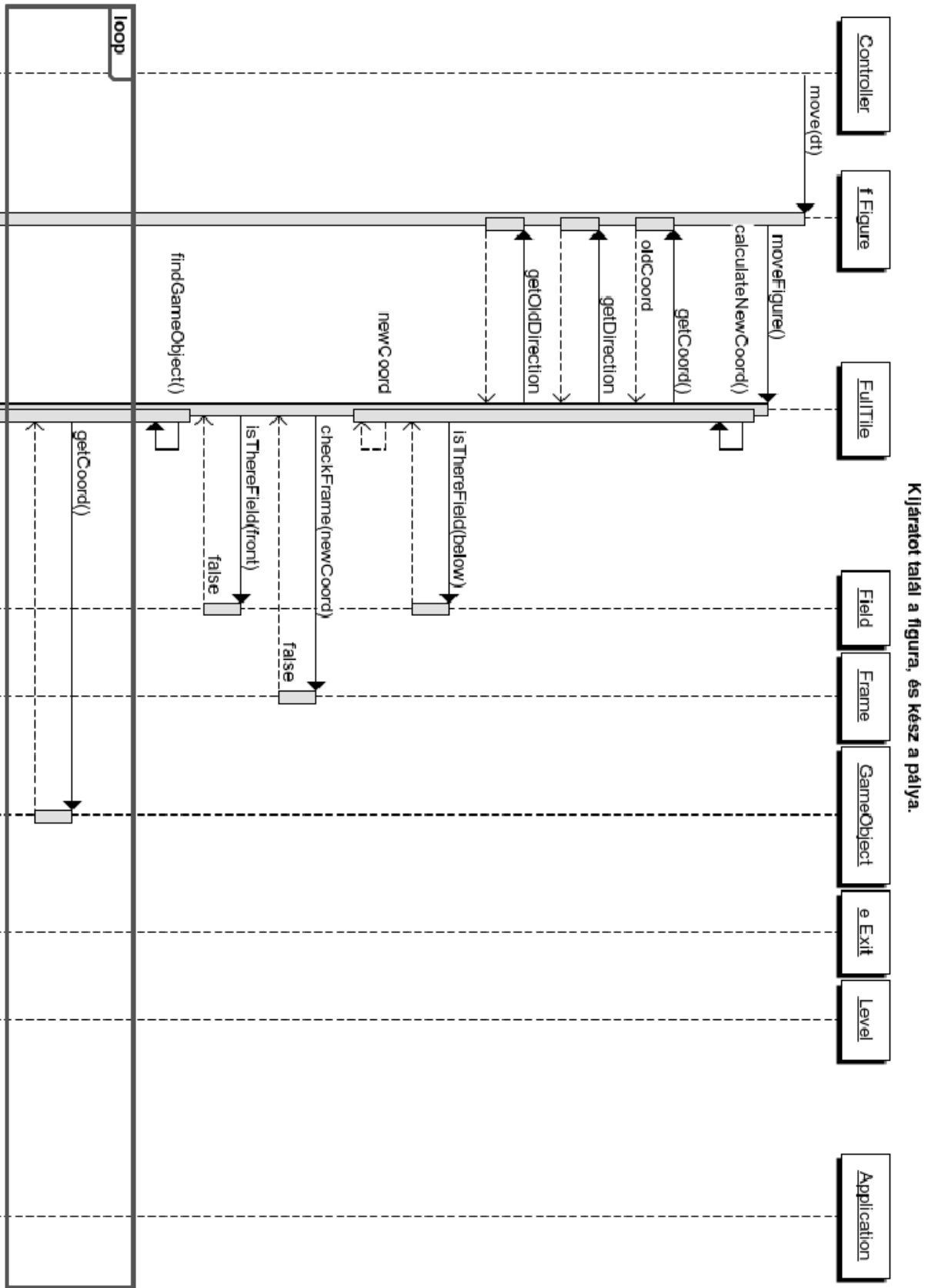
5.4.7 Teszteset #7

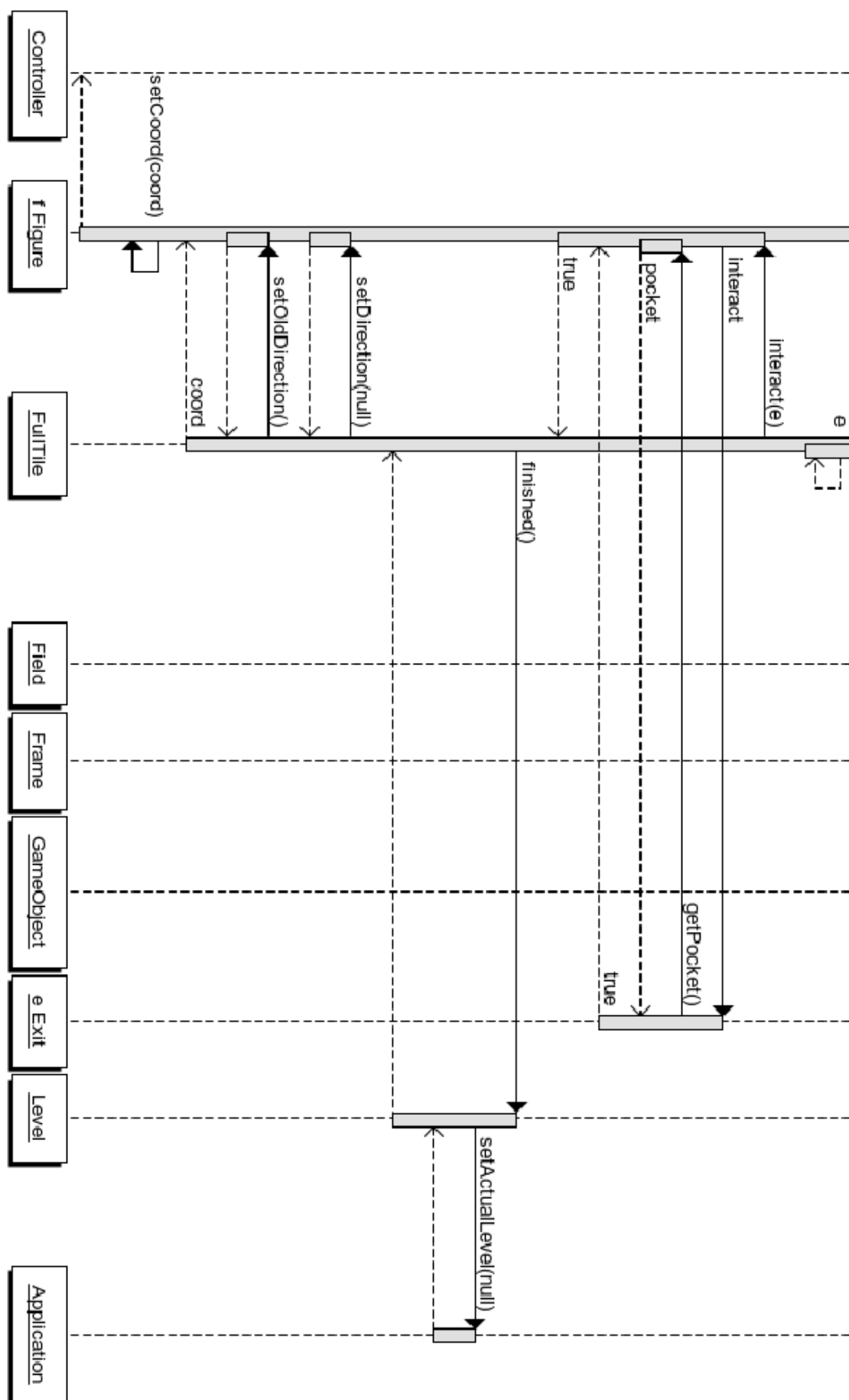
Tesztelendő use-case-ek:

- Kulcsot talál a bábu
- Kijáratot talál a bábu
- Új színt betöltése

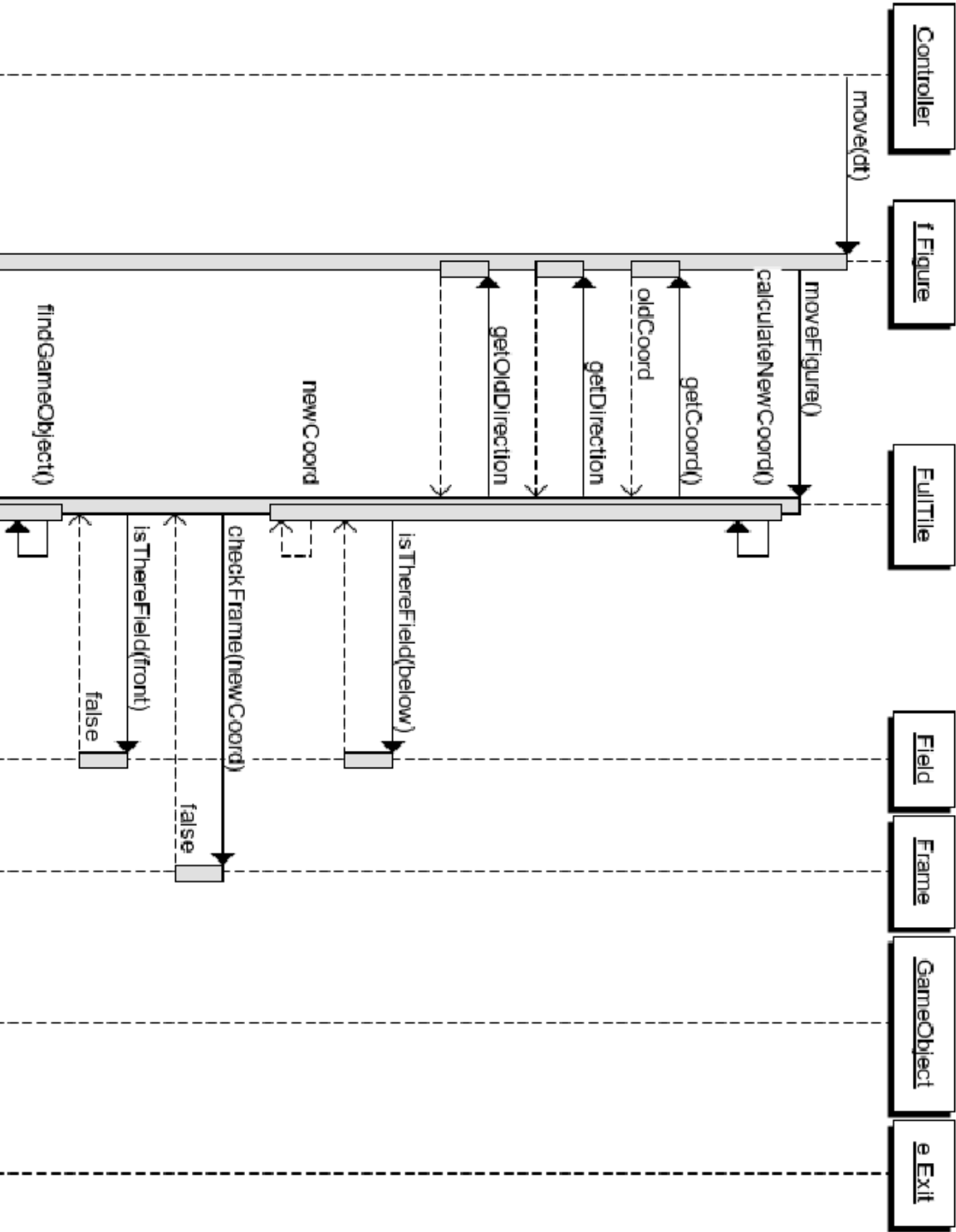


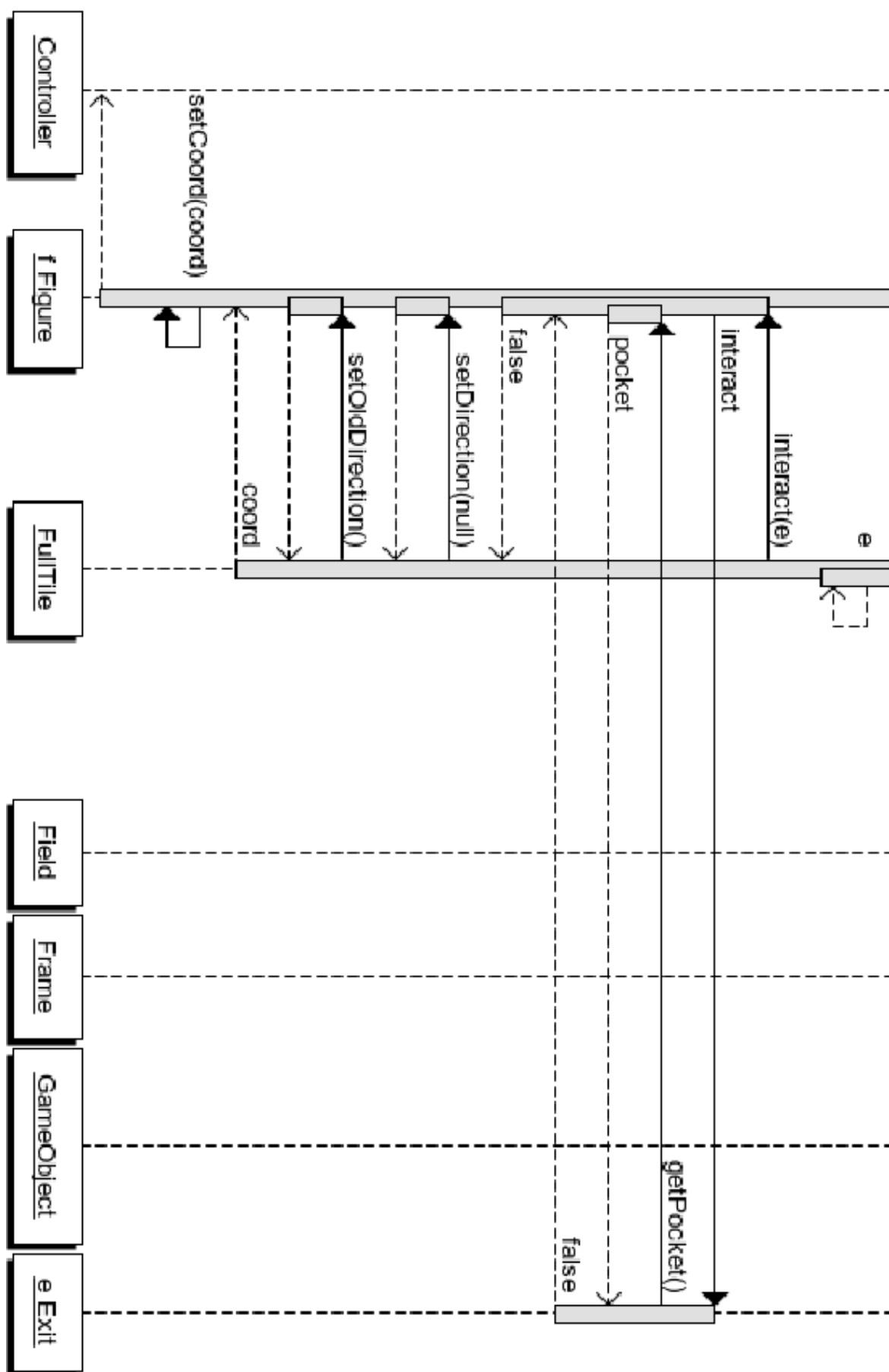






Kijáratot talál a figura, és nincs kész a pálya.





6. Szkeleton beadás

6.1 Fordítási és futtatási útmutató

6.1.1 Fájllista

Fájl neve	Méret	Keletkezés ideje	Tartalom
Controller.java	4 KB	2012.03.19.	A Controller osztály függvényei és változói.
Application.java	3 KB	2012.03.19.	Az Application osztály függvényei és változói.
Direction.java	1 KB	2012.03.19.	A Direction osztály függvényei és változói
EmptyTile.java	2 KB	2012.03.19.	Az üres csempe osztály függvényei és változói.
Exit.java	3 KB	2012.03.19.	A kijárat osztály függvényei és változói.
Field.java	3 KB	2012.03.19.	A mező osztály függvényei és változói,
Figure.java	7 KB	2012.03.19.	A bábu osztály függvényei és változói.
Frame.java	4 KB	2012.03.19.	A keret osztály függvényei és változói.
FullTile.java	12KB	2012.03.19.	A teli csempe osztály függvényei és változói.
GameObject.java	2 KB	2012.03.19.	A alakzat ösosztály függvényei és változói.
Key.java	2 KB	2012.03.19.	A kulcs osztály függvényei és változói.
Level.java	5 KB	2012.03.19.	A szint osztály függvényei és változói.
Tile.java	2 KB	2012.03.19.	A csempe ösosztály függvényei és változói.
Type.java	1 KB	2012.03.19.	A típus enum függvényei és változói.
Continuity.jar	19 KB	2012.03.19.	A projekt futtatható formában.

6.1.2 Fordítás

Nem szükséges fordítani, a Continuity.jar futtatásra kész.

6.1.3 Futtatás

Szükséges legalább Java 1.5 futtatókörnyezet jelenléte, és hogy az a PATH környezeti változóhoz hozzá legyen adva. Ez Windows XP-n a *Start -> Vezérlőpult -> Rendszer -> Speciális -> Környezeti változók -> Rendszerváltozók -> PATH*, Windows 7-en *Start -> Számítógép -> Rendszer tulajdonságai -> Speciális rendszerbeállítások -> Környezeti változók -> Rendszerváltozók -> PATH* kulcsban ellenőrizhető. Ha nincs hozzáadva, a kulcs elejéhez fűzzük hozzá a futtatókörnyezet elérési útját, majd egy pontosvesszőt.

Ez után indítsunk egy parancssort, navigáljunk el a .jar-t tartalmazó mappába, ott adjuk ki a következő parancsot: `java -jar Continuity.jar`.

6.2 Értékelés

Tag neve	Munka százalékban
Berki Ramóna	14.5 %
Gerlei Balázs	19.8 %
Halász Antal István	16.4 %
Kiss Beáta	19.6 %
Szabó Csaba	29.7 %

7. Prototípus koncepciója

7.1 Prototípus interface-definíciója

A prototípus program célja annak demonstrálása, hogy az eddigi tervek alapján a program helyesen működik. A prototípus a programunknak egy változata, mely grafikus interfészt és a végleges irányítást nem tartalmaz. A főbb objektumok már a végleges algoritmusokat tartalmazzák.

A prototípus kezelése konzolon keresztül történik az előre definiált ki- és bemeneti nyelv segítségével. A teszteléshez előre meghatározott teszteseteket készítünk, illetve a felhasználó maga is tesztelheti a program működését, ha tudja milyen kimenetet vár adott bemenetre. Az előre elkészített tesztesetekkel kapcsolatos alapelv, hogy azok átláthatóak, megismerhetőek, reprodukálhatóak legyenek, és a program valamennyi kritikus pontját lefedjék. Ahhoz, hogy a tesztesetek reprodukálhatóak legyenek, ki kell küszöbölni a véletlen faktort, amely ugyan a végső játékot változatossá teszi, de a tesztelés során elfogadhatatlan az, hogy nem ellenőrizzük le a véletlen esemény összes lehetséges kimenetelét. Csapatunk ezeket szem előtt tartva és ezeknek megfelelően állítja elő a prototípust és a teszteseteket.

7.1.1 Az interfész általános leírása

A prototípus program a szabványos be- és kimeneten kommunikál a felhasználóval az előre definiált ki- és bementi nyelv segítségével, parancsokat kap a szabványos bemenetről vagy előre elkészített parancsfájlból. A pályákat mindig fájlból tölti be. A program minden egyes parancs végrehajtásakor kiírja a szabványos kimenetre a parancs eredményét, illetve tesztesetek futtatásakor fájlba is menti a képernyő tartalmát, ezzel könnyítve a tesztelést.

7.1.2 Bemeneti nyelv

loadCommands <fileName> – Beolvassa és végrehajtja a paraméterként kapott fájlból a parancsokat. Pálya betöltésére, tesztesetek lefuttatására használjuk.
- `fileName`: A beolvasott fájl neve.

beginWriteCommands <fileName> –
- `fileName`: A kimeneti fájl neve.

endWriteCommands – Fájlba írás befejezése.

beginLevel <levelName> <x> <y> - Új szint létrehozása és szerkesztése. A szintenkénti egy darab üres (mozgásra képes) csempe automatikusan létrejön.
- `levelName`: A pálya neve, pl.: level1.
- `x, y`: Hányszor hányas "mátrixként" épül fel a szint a csempékből.

endLevel - Szint szerkesztésének befejezése. Csak még be nem zárt `beginLevel` parancs után hívható.

beginFullTile – A szinthez egy teli csempe hozzáadása és szerkesztése.
`beginLevel` és `endLevel` parancsok között hívható csak.

endFullTile – A beginFullTile-lal hozzáadott teli csempe szerkesztésének befejezése. beginFullTile parancs után hívható csak.

beginField– beginFullTile és endFullTile parancsok között használható parancs, egy összefüggő terepalakzat szerkesztésének megkezdése.

endField – Terepalakzat szerkesztésének vége.

addFieldPoint <x> <y> – Csak beginField és endField parancsok között hívható, a terep egy kontrollpontja adható meg vele.
- x, y: Az új kontrollpont koordinátái.

addKey <x> <y> - Csak beginFullTile és endFullTile parancsok között használható, mellyel egy kulcsot adunk a csempéhez.
- x, y: A kulcs koordinátái.

addExit <x> <y> <numberOfKeys> - Csak beginFullTile és endFullTile parancsok között használható, mellyel egy kijáratot adunk a csempéhez.
- x, y: A kijárat koordinátái.
- numberOfKeys: Hány kulcs szükséges a kijárhoz (szint teljesítéséhez).

addFigureA <row> <column> <startCoordX> <startCoordY> - Csak beginLevel és endLevel parancsok között hívható, az első figurát helyezük el az egyik csempébe, és beállítjuk a bábu kezdőkoordinátáját is.
- row, column: Melyik csempébe rakjuk a bábút.
- startCoordX, startCoordY: A csempében hova kerüljön a bábu.

addFigureB <row> <column> <startCoordX> <startCoordY> - Csak beginLevel és endLevel parancsok között hívható, az második figurát helyezük el az egyik csempébe, és beállítjuk a bábu kezdőkoordinátáját is.
- row, column: Melyik csempébe rakjuk a bábút.
- startCoordX, startCoordY: A csempében hova kerüljön a bábu.

showTiles - Táblázatosan kiírja a csempék állását.

moveTile <direction> - Csempe mozgatása.
- direction: Az irány, amerre a csempe mozog.

moveFigure <figureName> <direction> - Bábu mozgatása.
- figureName: A mozgatandó bábu neve. Csak "FigureA" vagy "FigureB" lehet.
- direction: Az irány, amerre a bábu mozog.

simulate - Egységnyi idő elteltének szimulálása. Ekkor mozognak a bábuk, és lépnek interakcióba a játék többi elemével.

help - Sógó hívása.

exit - Kilépés a programból.

7.1.3 Kimeneti nyelv

loadCommands <fileName> – Kiírja a parancsfájl betöltésének eredményét.

A lehetséges kimenetek:

1. <fileName> betöltése sikeres.
2. <fileName> betöltési hiba!

beginWriteCommands <fileName> – Üzenet a fájlba kiírás megkezdéséről.

A kimenet: nincs kimenet

endWriteCommands – Fájlba írás befejezése.

A kimenet: nincs kimenet

beginLevel <levelName> <x> <y> – Kiírja a szint létrehozásának eredményét.

A lehetséges kimenetek:

1. <levelName> szint létrehozása sikeres.
2. Hiba a <levelName> szint létrehozásakor!

endLevel – Kiírja a szintszerkesztés befejezésének eredményét.

A lehetséges kimenetek:

1. Szint szerkesztése befejezve.
2. Hiba a szint szerkesztésének befejezése közben!

beginFullTile – Üzenet egy teli csempe hozzáadásáról.

A lehetséges kimenetek:

1. Teli csempe hozzáadva.
2. Hiba a teli csempe hozzáadásakor!

endFullTile – Üzenet a teli csempe szerkesztésének befejezéséről.

A lehetséges kimenetek:

1. Teli csempe szerkesztése befejezve.
2. Hiba a teli csempe szerkesztésének befejezésekor!

beginField – Üzenet az adott teli csempében lévő új terepalakzat szerkesztésének megkezdéséről.

A lehetséges kimenetek:

1. Új terepalakzat szerkesztése.
2. Hiba a terepalakzat szerkesztésekor!

endField – üzenet a terepalakzat szerkesztésének befejezéséről.

A lehetséges kimenetek:

1. Terepalakzat szerkesztése befejezve.
2. Hiba a terepalakzat szerkesztésének befejezésekor!

addFieldPoint <x> <y> – Üzenet egy kontrollpont hozzáadásáról a terepalakzathoz.

A lehetséges kimenetek:

1. Kontrollpont <x> <y> hozzáadva.
2. Hiba a kontrollpont <x> <y> hozzáadásakor!

addKey <x> <y> - Üzenet arról, hogy egy kulcs hozzá lett adva egy teli csempéhez.

A lehetséges kimenetek:

1. Kulcs <x> <y> helyen hozzáadva.
2. Hiba a kulcs hozzáadásakor <x> <y> helyre!

addExit <x> <y> <numberOfKeys> - Üzenet arról, hogy egy kijárat hozzá lett adva egy teli csempéhez.

A lehetséges kimenetek:

1. <numberOfKeys> kulcsos kijarat <x> <y> helyen hozzáadva.
2. Hiba a <numberOfKeys> kulcsos kijarat hozzáadásakor <x> <y> helyre!

addFigureA <row> <column> <startCoordX> <startCoordY> - Kiírja az első bábu hozzáadásának eredményességét.

A lehetséges kimenetek:

1. FigureA a(z) <column> <row> pozícióju teli csempeben <startCoordX> <startCoordY> pozícióba hozzáadva.
2. Hiba FigureA hozzáadásakor a <column> <row> pozícióju teli csempeben <startCoordX> <startCoordY> pozícióba!

addFigureB <row> <column> <startCoordX> <startCoordY> - Kiírja a második bábu hozzáadásának eredményességét.

A lehetséges kimenetek:

1. FigureB a <column> <row> pozícióju teli csempeben <startCoordX> <startCoordY> pozícióba hozzáadva.
2. Hiba FigureB hozzáadásakor a <column> <row> pozícióju teli csempeben <startCoordX> <startCoordY> pozícióba!

showTiles – Táblázatosan kiírja a csempék jelenlegi állását.

Egy lehetséges kimenet 4 csempe esetén:

```
-----
EmptyTile      FullTile1
FullTile2      FullTile3
-----
```

moveTile <direction> - Táblázatosan kiírja a csempék mozgítás utáni állását, illetve hibát ír, ha nem lehetséges a mozgítás, ekkor a csempék jelenlegi állását írja ki táblázatosan.

A lehetséges kimenetek 4 csempe esetén (a csempék helyzete különbözhet):

1. -----

FullTile1	EmptyTile
FullTile2	FullTile3

2. Az adott irányba nem mozdulhatnak a csempek! A csempek jelenlegi állása:

EmptyTile	Fulltile1
FullTile2	FullTile3

moveFigure <figureName> <direction> - Kiírja az adott bábu adott irányba történő mozgásirány-beállításának eredményességét.

A lehetséges kimenetek:

1. <figureName> mozgásiránya beállítva <direction> irányba.
2. Hiba <figureName> mozgásirányának beállításakor <direction> irányba!

simulate - Kiírja a szimulálásának eredményeit. Megjeleníti a szimuláció alatt történt fontosabb interakciókat, amik a következők: a bábu falnak ütközik, kulcsot vagy kijáratot talál, meghal, befejeződik a szint. Hibát ír, ha olyan módon vagyunk (csempemozgatás, szintszerkesztés), amiben nem lehet időt léptetni.

A lehetséges kimenetek:

1. ~~<time> ms szimulálása.~~
2. <figureName> falnak ütközött!
3. <figureName> kulcsot talált.
4. <figureName> kijárhoz ért, de nincs elég kulcsa.
5. <figureName> kijárhoz ért, és van elég kulcsa. A szint teljesítve!
6. <figureName> meghalt, majd a <startCoordX> <startCoordY> koordinátán újraeledt.
7. <figureName> elkezdett ugrani.
8. <figuraName> elkezdett zuhanni.
9. <figureName> atmegy <direction> irányban egy másik csempebe.
10. <figureName> <direction> irányban nem tudja elhagyni a csempet!
11. A két figura találkozott!

Kiírjuk a bábu új koordinátáját is:

<figuraName> a(z) <row> <column> csempében a(z) <X> <Y> koordinátára került.

help – Kiírja a programban elérhető parancsok listáját és azok rövid használati utasítását.

exit – Nincs kimenet.

7.2 Összes részletes use-case

Use-case neve	Játék indítása
Rövid leírás	A felhasználó elindítja a játékot.
Aktorok	Felhasználó
Forgatókönyv	A felhasználó elindítja a játékot, majd a kiválasztott pálya betölt.

Use-case neve	Bábu irányítása
Rövid leírás	A felhasználó a bábút irányítja.
Aktorok	Felhasználó
Forgatókönyv	A felhasználó valamilyen irányba mozgathatja a bábút, ha az tud arra mozogni.

Use-case neve	Csempe irányítása
---------------	-------------------

Rövid leírás	A felhasználó mozgatja az üres csempét.
Aktorok	Felhasználó
Forgatókönyv	A felhasználó négy irányba mozgathatja az üres csempét, ha az tud arra mozogni.

Use-case neve	Bábu csempét vált
Rövid leírás	A bábu egyik csempéből a másikba megy.
Aktorok	Óra
Forgatókönyv	A bábu átmegy egyik csempéből a másikba, ha azok illeszkednek.

Use-case neve	Bábu meghal
Rövid leírás	A bábu meghal és visszatér a kezdőkoordinátájára.
Aktorok	Óra
Forgatókönyv	A bábu egy alatta lévő csempébe akar átmenni a levegőn keresztül, de azok nem illeszkednek, ilyenkor a bábu visszatér a kiinduló koordinátájára a start csempébe.

Use-case neve	Kijáratot talál a bábu
Rövid leírás	Megtörténik a bábu kijáratral való interakciója.
Aktorok	Óra
Forgatókönyv	A bábu elmozdul és az új helyén egy kijárat, ha van elég kulcsa, akkor vége a pályának, ha nincs akkor továbbmegy.

Use-case neve	Kulcsot talál a bábu
Rövid leírás	Megtörténik a bábu kulccsal való interakciója.
Aktorok	Óra
Forgatókönyv	A bábu elmozdul és az új helyen egy kulcs lesz. Ha még nem vette fel a kulcsot, akkor felveszi.

Use-case neve	Bábu mozgatása
Rövid leírás	A bábu a felhasználó által megadott irány felé elmozdul.
Aktorok	Óra
Forgatókönyv	A felhasználó megad egy irányt, és ha az óra lép egyet, a bábu a kiszámolt (gravitációt figyelembe vevő) koordinátára lép.

Use-case neve	Bábu tereppel ütközik
Rövid leírás	A bábu tereppel ütközik.
Aktorok	Óra
Forgatókönyv	A bábu nem tud a felhasználó által beadott irányba elmozdulni, mert ott terep van, így ott marad ahol eddig állt.

7.3 Tesztelési terv

A teszteléshez előre elkészítjük a bemeneti és az elvárt kimeneti fájlokat. A tesztelés során megvizsgáljuk, hogy a játék a megfelelő eredményt produkálja-e. Amennyiben jó a kimenet, a tesztet sikeresnek tekintjük, különben megpróbáljuk meghatározni a hiba okát, majd kijavítani azt.

A kimeneti eredményeket minden tesztelés alkalmával fájlban rögzítjük.

Teszteset neve	Pálya betöltése
Rövid leírás	A pályát betöltjük egy adott fájlból (itt betöltődnek a csempék, a terep, a kulcs, a kijárat).
Teszt célja	Leellenőrizzük, hogy a pálya és a rajta található fix elemek betöltődnek-e.

Teszteset neve	A bábu mozgása
Rövid leírás	A bábu fut, ugrik, zuhan, közben ütközik is.
Teszt célja	A bábu mozgásának és a gravitációnak a tesztelése.

Teszteset neve	A csempe mozgatása
Rövid leírás	Csempéket mozgatunk különböző sorrendben.
Teszt célja	A csempe mozgatásának tesztelése.

Teszteset neve	A bábu át tud haladni a csempén
Rövid leírás	A bábu át akar haladni a csempén oldalt, a csempék illeszkednek, átmegy, majd folytatja az útját.
Teszt célja	A csempe illeszkedés függvényének vizsgálata.

Teszt-eset neve	A bábu nem tud áthaladni a csempén.
Rövid leírás	A bábu át akar haladni a csempén oldalt, de nem illeszkednek.
Teszt célja	Az illeszkedés függvényének a vizsgálata.

Teszt-eset neve	A bábu meghal
Rövid leírás	A bábu át akar menni a csempe alsó keretén, de azok nem illeszkednek, a bábu meghal, majd a bábu visszatér a kiinduló pontjára.
Teszt célja	A bábu halálának vizsgálata.

Teszteset neve	Bábu kulcsot talál, majd eléri a kijáratot és kimegy rajta.
Rövid leírás	Megy a csempében a bábu, megtalál egy kulcsot, felveszi, majd elmegy a kijáratához és a kijárat leellenőrzi és kiengedi.

Teszt célja	Fel tudja-e venni a bábu a kulcsot és el tudja-e raktározni, ekkor a kulcs inaktiválódik-e, le tudja-e ellenőrizni a kijárat, hogy megfelelő számú kulcs van-e a bábunál.
--------------------	---

Teszteset neve	Bábu áthalad egy felszedett kulcson, kijáratot talál, de nincs nála elég kulcs
Rövid leírás	A bábu megy, áthalad egy felvett kulcs helyén, majd talál egy kijáratot, de továbbhalad, mert nincs nála elég kulcs.
Teszt célja	A felvett kulcs tényleg deaktiválódik-e, ha nincs elég kulcs a kijárat valóban nem enged-e ki.

Teszteset neve	Két bábu találkozik
Rövid leírás	A két bábu találkozik egymással, majd továbbmennek.
Teszt célja	A bábuk találkozásának tesztelése.

Teszteset neve	A bábu felszedi az első kulcsot, zuhanás közben éri el a második kulcsot, majd a kijáratot.
Rövid leírás	A bábu zuhanás közben eléri a kulcsot, aztán megáll egy terepen kicsit tovább megy megint elkezd zuhanni és eléri a kijáratot.
Teszt célja	Annak ellenőrzése, hogy a bábu zuhanás közben képes-e felvenni a kulcsot és elérni a kijáratot.

7.4 Tesztelést támogató segéd- és fordítóprogramok specifikálása

Az előre definiált teszteseteink eredményét mindig kiíratjuk fájlba, illetve definiálunk mindegyik esethez egy output file-t is, amely az elvárt kimenetet tartalmazza. A gyorsabb munka érdekében a teszteset eredményét tároló fájlt és az elvárt kimenetet tartalmazó file-t egy egyszerű programmal szeretnénk összehasonlítani, amely visszaadja, ha azonos a két fájl tartalma, illetve ha nem egyeznek meg, akkor kiírja, hogy mely sorokkal volt gond.

8. Részletes tervek

8.1 Osztályok és metódusok tervei.

8.1.1 Application

- **Felelősség**
Ezt hozza létre először a modellek közül a Controller, felelős a szintek betöltéséért, a játékaállítás mentéséért, egy korábban mentett állás betöltéséért, a program futásának befejezéséért. Mozgásra utasítja a csempéket vagy a figurát. Kezeli az idő múlását.
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - - **actualLevel : Level** *Az éppen betöltött szint.*
 - - **levelsPath : String[]** *A szintek elérési útját tárolja.*
 - - **levelsPathIndex : int** *Értéke az aktuális szint indexe a levelsPath tömbben.*
 - - **previousTime : int** *Az előző simulateWorld futási ideje.*
 - - **savedLevelPath : String** *Az elmentett szint elérési útja.*
- **Metódusok**
 - + **clearActualLevel() : void** *Kinullázza a pillanatnyi pályát.*
 - + **exit() : void** *Kilép a programból és automatikusan menti a játék állását.*
 - + **loadLevel() : void** *Betölti az átadott elérési útnak megfelelő szintet. Visszaadja, hogy sikerült-e a betöltés.*
 - + **move(what:Type, dir:Direction) : void** *Ha van betöltött szint, akkor továbbítja a szintnek, hogy mi mozogjon (what paraméter) és milyen irányba (dir paraméter).*
 - + **simulateWorld(time:int) : void** *Az idő múlását szimulálja, a betöltött szint gravitációt kezelő függvényét hívja, ha van betöltött szint.*

8.1.2 Controller

- **Felelősség**
Program indításakor jön létre. Felelős az idő kezeléséért és a csempe vagy bábu mozgatásáért.
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - - **app: Application** *A futtatandó program.*
- **Metódusok**
 - + **main(args:String[]) : void** *A program main függvénye.*

8.1.3 Direction

- **Felelősség**
Segédosztály, mely az irányvektort reprezentálja.
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**

- - **angle : double** *Irányvektor szöge*
- - **length : double** *Irányvektor hossza*
- **Metódusok**
 - + **getAngle : double** *Visszaadja az irányvektor szögét.*
 - + **getLength : double** *Visszaadja az irányvektor hosszát.*

8.1.4 EmptyTile

- **Felelősség**
Olyan mozgásra képes csempe, mely nem tartalmaz terepet. Benne bábu nem lehet.
- **Ősosztályok**
Tile
- **Interfészek**
nincs
- **Attribútumok**
nincs
- **Metódusok**
 - + **move(dir:Direction) : void** *Az üres csempe mozgását végző függvény. Ha van abba az irányba másik csempe, amerre mozgatni akarjuk, akkor megcseréli a két csempét.*

8.1.5 Exit

- **Felelősség**
Ezt kell elérni a megfelelő kulcsokkal a birtokunkban a szint teljesítéséhez. Tudja, hogy hány kulcsra van szüksége. Ha nincs elég kulcs a bábunál, akkor megy tovább a bábu.
- **Ősosztályok**
GameObject
- **Interfészek**
nincs
- **Attribútumok**
 - - **numberOfKeys : int** *A szint teljesítéséhez szükséges kulcsok száma.*
- **Metódusok**
 - + **getNumberOfKeys() : int** *Visszaadja a szint teljesítéséhez szükséges kulcsok számát.*
 - + **interact(figure:Figure) : boolean** *Megvizsgálja a kapott figura zsebében lévő kulcsokat. Ha megfelelő számú van, akkor true-val tér vissza, ha nem akkor false-szal.*
 - + **setNumberOfKeys(number:int) : void** *Beállítja a szint teljesítéséhez szükséges kulcsok számát.*

8.1.6 Field

- **Felelősség**
A terepet reprezentálja. Meg tudja mondani azt, hogy egy adott pontban van-e terep.
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Attribútumok**
 - - **points : ArrayList<Point>** *A terepet reprezentáló pontok. A terepen egyenesek vannak. Egy egyenes két végpontját egy Point collection tárolja index alapján sorban. Több Point tömb (egyenes) reprezentálja a terepet.*

- **Metódusok**

- + **isThereField(point:Point) : boolean** Megmondja, hogy a paraméterként kapott pontban terep van-e.

8.1.7 Figure

- **Felelősség**

A felhasználó által irányított bábut valósítja meg. Tud mozogni. Képes megadni a szükséges adatokat a pontos elmozdulás-számításhoz, valamint képes visszaadni a zsebet.

- **Ősosztályok**

Object

- **Interfészek**

nincs

- **Attribútumok**

- - **coord : Point** A bábu koordinátája.
- - **dir : Direction** A felhasználói input által beállított irányvektor, ebbe az irányba akar elmozdulni a bábu.
- - **myTile: FullTile** Referencia arra a csempére, amiben éppen van a bábu.
- - **oldDirection : Direction** Az előző lépésben kiszámított elmozdulás vektor.
- - **pocket : ArrayList<Key>** A bábu zsebe, melyben kulcsokat tárolunk.

- **Metódusok**

- + **getCoord() : Point** Visszaadja a bábu koordinátáját.
- + **getDirection() : Direction** Visszaadja az input alapján beállított irányvektort.
- + **getOldDirection() : Direction** Visszaadja az előző lépésben számított elmozdulásvektort.
- + **getPocket() : ArrayList<Key>** Visszaadja a bábu zsebet.
- + **interact(g:GameObject) : boolean** GameObject interact függvény. Paraméterként kapja azt a GameObject-et, amivel interactolni akarunk.
- + **move(dt:int) : void** Mozgatja a bábut.
- + **setCoord(point:Point) : void** Beállítja a bábu koordinátáját.
- + **setDirection(dir:Direction) : void** Beállítja az input alapján a kívánt elmozdulás irányvektorát.
- + **setMyTile(myTile:FullTile) : void** Beállítja a csempére mutató referenciát.
- + **setOldDirection(dir:Direction) : void** Beállítja a kiszámított elmozdulás vektort.

8.1.8 Frame

- **Felelősség**

A csempék keretét reprezentáló osztály. Meg tudja mondani, hogy két csempe terepe megegyezik-e a megadott szélén. A koordináták transzformálását végzi.

- **Ősosztályok**

Object

- **Interfészek**

nincs

- **Attribútumok**

- - **framePoints : ArrayList<Point>** A terepet szélét reprezentáló pontok. Ebből áll a teli csempe kerete.

- **Metódusok**

- + **checkFrame(coord:Point) : boolean** Megnézi, hogy a paraméterként kapott koordinátán keret van-e.
- + **deTransformCoord(coord:Point) : Point** Közös koordinátát belső koordinátává vált át.

- + **match (frame:Frame) : boolean** Megvizsgálja, hogy egyezik-e a paraméterként kapott kerettel.
- + **transformCoord(coord:Point) : Point** Belső koordinátát közös koordinátává vált át.

8.1.9 FullTile

- **Felelősség**

Olyan csempe, amely tartalmaz terepet és olyan alakzatokat mellyel interakcióba tud lépni a bábu. Ebben a bábu tud mozogni. Kiszámolja a bábu pontos elmozdulásának irányát és nagyságát a bábút körülvevő tereptől függően. Meg tudja kérdezni a terepet, hogy adott pontban terep van-e. Meg tudja vizsgálni, hogy illeszkedik-e egy másik csempével, az alapján, hogy a bábu honnan hova akar elmozdulni.

- **Ősosztályok**

Tile

- **Interfészek**

nincs

- **Attribútumok**

- - **field : Field** A benne lévő terep.
- - **frames : ArrayList<Frame>** Csempe kereteit tároló kollekció.
- - **figure : Figure** Referencia a bábura, értéke null, ha a bábu nincs a csempében.
- - **gameObjects : ArrayList<GameObject>** A csempében lévő kulcs(ok)at és/vagy kijáratot tároló tömb.

- **Metódusok**

- - **calculateNewCoord() : Point** Visszaadja a bábu pontos elmozdulását a bábura ható pillanatnyi erőhatások figyelembevételével.
- + **changeDirection(dir:Direction) : void** A bábu kívánt elmozdulásának beállításáért felelős függvény.
- + **changePosition(coord:Point) : Point** Megváltoztatja a csempe helyét, visszaadja a csempe régi helyét.
- - **findGameObject() : GameObject** Visszaadja azt a GameObject objektumot, ami a kiszámított elmozdulás helyén található. Ha nincs ott objektum, vagy a csempe nem tartalmaz GameObjecteket, visszatérési értéke null.
- + **getFrame(dir:Direction) : Frame** Visszaadja a bekért oldalon lévő keretet.
- + **isThereFigure() : boolean** Megmondja, hogy van-e figura a csempében.
- + **moveFigure() : Point** Kiszámítja a bábu pontos elmozdulását.
- + **setFigure(f:Figure) : void** Beállítja a bábu referenciáját.
- + **setFigureCoord(coord:Point) : void** Beállítja a benne lévő bábu koordinátáját.
- + **simulateWorld(dt:int) : void** Időkezelés továbbadása a bábunak.

8.1.10 GameObject

- **Felelősség**

Ez az absztrakt osztály kezeli a bábu zsebét. Megmondja, hogy a vele történő érintkezés hatására a pálya teljesített lett-e.

- **Ősosztályok**

Object

- **Interfészek**

nincs

- **Attribútumok**

- - **coord : Point** Megmondja, hol helyezkedik el a GameObject.

- **Metódusok**

- + **getCoord() : Point** Visszaadja a koordinátáját.

- + **interact (f:Figure) : boolean** *Interactol a kapott figurával.*
- + **setCoord(point:Point) : void** *Beállítja a koordinátáját.*

8.1.11 Key

- **Felelősség**

A szint teljesítéséhez szükséges. Amikor a bábu eléri a kulcsot, akkor a kulcs beteszi önmagát a bábu zsebébe. Meg tudja mondani, hogy felvették-e.

- **Ősosztályok**

GameObject

- **Interfészek**

nincs

- **Attribútumok**

- - **pickedUp : boolean** *Megmondja, hogy felvették-e már.*

- **Metódusok**

- + **isPickedUp() : boolean** *Visszaadja, hogy felvették-e már.*
- + **interact(figure:Figure) : boolean** *Interactol a paraméterként kapott figura zsebével (beteszi magát).*

8.1.12 Level

- **Felelősség**

Felelős a csempék elrendezéséért, és a bábu kívánt irányú mozgásának beállításáért. Meg tudja mondani az adott szint kezdő csempéjét és a bábu azon belüli kezdőpontját.

- **Ősosztályok**

Object

- **Interfészek**

nincs

- **Attribútumok**

- - **app : Application** Referencia az alkalmazásra, ami létrehozta. A következő szint betöltésekor használjuk.
- - **figureAStartCoord : Point** A bábu kezdőcsempén belüli kezdőkoordinátája.
- - **figureBStartCoord : Point** B bábu kezdőcsempén belüli kezdőkoordinátája.
- - **figureAStartTile : int** A bábu kezdőcsempe referenciája.
- - **figureBStartTile : int** B bábu kezdőcsempe referenciája.
- - **tiles : Collection** A szinten lévő csempék. A 0. indexű elem az egyetlen üres csempe (EmptyTile).

- **Metódusok**

- + **finished() : void** Jelzi az Application felé, hogy a szint teljesítve lett.
- + **getFigureStartCoord() : Point** Referenciát ad vissza a kezdő koordinátákra.
- + **getNeighbour(dir:Direction, pos:Point) : FullTile** Referenciát ad az argumentumok alapján keresett csempére. Egy csempe adott oldali szomszédjának kiderítésére használjuk.
- + **getFigureAStartTile() : FullTile** Referenciát ad vissza A bábu kezdő csempéjére.
- + **getFigureBStartTile() : FullTile** Referenciát ad vissza B bábu kezdő csempéjére.
- + **move(what:Type, dir:Direction) : void** Megkapja, hogy mi mozogjon, és ennek hívja meg a mozgató metódusát, aminek átadja a kapott irányt (dir).
- + **simulateWorld(dt:int) : void** Továbbadja az időt annak a FullTile-nak, amelyekben a bábu van.

8.1.13 Tile

- **Felelősség**

Ezzel az absztrakt osztállyal lehetséges a FullTile-ok és EmptyTile-ok egy tömbben tárolása.

- **Ősosztályok**
nincs
- **Interfészek**
nincs
- **Attribútumok**
 - - **coord : Point** A csempe koordinátája.
 - - **height : int** A csempe magassága.
 - - **myLevel : Level** A csempe saját szintje.
 - - **width : int** A csempe szélessége.
- **Metódusok**
 - + **getCoord() : Point** Megadja a csempe koordinátáját.
 - + **getHeight() : int** Megadja a csempe magasságát.
 - + **getWidth() : int** Megadja a csempe szélességét.
 - + **setCoord() : void** Beállítja a csempe koordinátáját.

8.1.14 Type

- **Felelősség**
Olyan segédosztály, enum típus, melyet a mozgatóskor használunk arra a célra, hogy eldöntsük, csempét vagy bábút mozgatunk.
- **Ősosztályok**
Object
- **Interfészek**
nincs
- **Enum állapotok**
 - Tile
 - FigureA
 - FigureB

8.2 A tesztek részletes tervei, leírásuk a teszt nyelvén

8.2.1 Teszteset1: Pálya betöltése

- **Leírás**

A pályát betöltjük egy adott fájlból (itt betöltődnek a csempék, a terep, a kulcs, a kijárat).

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Leellenőrizzük, hogy a pálya és a rajta található fix elemek betöltődnek-e. Ha nem olyan sorrendben vagy nem pont azok az elemek töltődnek be, mint amik a fájlunkban is találhatóak akkor a programunk hibásan működik.

- **Bemenet**

```
beginWriteCommands teszteset1_output.txt
loadCommands szint1.txt
endWriteCommands
exit
```

- **Elvárt kimenet (teszteset1_output.txt)**

```
level1 szint létrehozasa sikeres.
```

Teli csempe hozzáadva.

Uj terepalakzat szerkesztése.

Kontrollpont 0 100 hozzáadva.

Kontrollpont 100 100 hozzáadva.

Kontrollpont 100 300 hozzáadva.

Kontrollpont 0 300 hozzáadva.

Terepalakzat szerkesztése befejezve.

Uj terepalakzat szerkesztése.

Kontrollpont 200 0 hozzáadva.

Kontrollpont 300 0 hozzáadva.

Kontrollpont 300 100 hozzáadva.

Kontrollpont 200 100 hozzáadva.

Terepalakzat szerkesztése befejezve.

Uj terepalakzat szerkesztése.

Kontrollpont 400 100 hozzáadva.

Kontrollpont 500 100 hozzáadva.

Kontrollpont 500 300 hozzáadva.

Kontrollpont 400 300 hozzáadva.

Terepalakzat szerkesztése befejezve.

Uj terepalakzat szerkesztése.

Kontrollpont 200 300 hozzáadva.

Kontrollpont 300 300 hozzáadva.

Kontrollpont 300 400 hozzáadva.

Kontrollpont 200 400 hozzáadva.

Terepalakzat szerkesztése befejezve.

Teli csempe szerkesztése befejezve.

Teli csempe hozzáadva.

Uj terepalakzat szerkesztése.

Kontrollpont 200 0 hozzáadva.

Kontrollpont 500 0 hozzáadva.

Kontrollpont 500 100 hozzáadva.
 Kontrollpont 200 100 hozzáadva.
 Terepalakzat szerkesztése befejezve.
 Kulcs 450 113 helyen hozzáadva.
 Teli csempe szerkesztése befejezve.

Teli csempe hozzáadva.
 Új terepalakzat szerkesztése.
 Kontrollpont 0 0 hozzáadva.
 Kontrollpont 500 0 hozzáadva.
 Kontrollpont 500 200 hozzáadva.
 Kontrollpont 300 200 hozzáadva.
 Kontrollpont 300 100 hozzáadva.
 Kontrollpont 0 100 hozzáadva.
 Terepalakzat szerkesztése befejezve.
 Új terepalakzat szerkesztése.
 Kontrollpont 200 300 hozzáadva.
 Kontrollpont 500 300 hozzáadva.
 Kontrollpont 500 400 hozzáadva.
 Kontrollpont 200 400 hozzáadva.
 Terepalakzat szerkesztése befejezve.
 1 kulcsos kijarat 50 123 helyen hozzáadva.
 Teli csempe szerkesztése befejezve.
 FigureA a(z) 2 1 pozícióju teli csempeben 250 118 pozícióba hozzáadva.
 FigureB a(z) 2 2 pozícióju teli csempeben 450 218 pozícióba hozzáadva.
 Szint szerkesztése befejezve.
 szint1.txt betöltése sikeres.

8.2.2 Teszteset2: A bábu mozgatása

- **Leírás**
 A bábu fut, ugrik, zuhan, közben ütközik is.
- **Ellenőrzött funkcionalitás, várható hibahelyek**
 A tesztesetben elsősorban azt teszteljük, hogy a bábút megfelelően tudjuk-e mozgatni. A bábu új koordinátáját kiszámoló matematika formulában lehet gond illetve a figura és egyéb alakzatok ütközésetektálásánál lehet hiba a programban.
- **Bemenet (teszteset2.txt)**

```
loadCommands szint2.txt
beginWriteCommands teszteset2_output.txt
moveFigure FigureA E
simulate
moveFigure FigureA W
simulate
moveFigure FigureA N
simulate
simulate
simulate
simulate
simulate
moveFigure FigureA NE
simulate
simulate
```

```

simulate
simulate
moveFigure FigureA NW
simulate
simulate
simulate
simulate
moveFigure FigureB E
simulate
moveFigure FigureB W
simulate
moveFigure FigureB NW
simulate
moveFigure FigureB W
simulate
moveFigure FigureB W
simulate
simulate
simulate
moveFigure FigureB W
simulate
moveFigure FigureB W
simulate
endWriteCommands
exit

```

- **Elvárt kimenet (teszteset2_output.txt)**

```

FigureA mozgásiránya beállítva Kelet irányba.
FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara kerult.
FigureA mozgásiránya beállítva Nyugat irányba.
FigureA a(z) 2 1 csempeben a(z) 250 118 koordinatara kerult.
FigureA mozgásiránya beállítva Eszak irányba.
FigureA elkezdett ugrani.
FigureA a(z) 2 1 csempeben a(z) 250 210 koordinatara kerult.
FigureA a(z) 2 1 csempeben a(z) 250 256 koordinatara kerult.
FigureA elkezdett zuhanni.
FigureA a(z) 2 1 csempeben a(z) 250 210 koordinatara kerult.
FigureA a(z) 2 1 csempeben a(z) 250 164 koordinatara kerult.
FigureA a(z) 2 1 csempeben a(z) 250 118 koordinatara kerult.
FigureA mozgásiránya beállítva Eszakkelet irányba.
FigureA elkezdett ugrani.
FigureA a(z) 2 1 csempeben a(z) 350 218 koordinatara kerult.
FigureA elkezdett zuhanni.
FigureA a(z) 2 1 csempeben a(z) 405 172 koordinatara kerult.
FigureA kulcsot talalt.
FigureA a(z) 2 1 csempeben a(z) 460 126 koordinatara kerult.
FigureA a(z) 2 1 csempeben a(z) 472 118 koordinatara kerult.
FigureA mozgásiránya beállítva Eszaknyugat irányba.
FigureA elkezdett ugrani.
FigureA a(z) 2 1 csempeben a(z) 372 218 koordinatara kerult.
FigureA elkezdett zuhanni.
FigureA a(z) 2 1 csempeben a(z) 317 172 koordinatara kerult.
FigureA a(z) 2 1 csempeben a(z) 262 126 koordinatara kerult.

```


FigureA a(z) 2 1 csempeben a(z) 250 118 koordinatara kerult.
 FigureB mozgásiranya beallitva Kelet irányba.
 FigureB Kelet irányban nem tudja elhagyni a csempet.
 FigureB a(z) 2 2 csempeben a(z) 495 218 koordinatara kerult.
 FigureB mozgásiranya beallitva Nyugat irányba.
 FigureB a(z) 2 2 csempeben a(z) 395 218 koordinatara kerult.
 FigureB mozgásiranya beallitva Eszaknyugat irányba.
 FigureB elkezdett ugrani.
 FigureB falnak utkozott.
 FigureB elkezdett zuhanni.
 FigureB a(z) 2 2 csempeben a(z) 303 292 koordinatara kerult.
 FigureB mozgásiranya beallitva Nyugat irányba.
 FigureB a(z) 2 2 csempeben a(z) 248 246 koordinatara kerult.
 FigureB mozgásiranya beallitva Nyugat irányba.
 FigureB a(z) 2 2 csempeben a(z) 193 200 koordinatara kerult.
 FigureB a(z) 2 2 csempeben a(z) 193 154 koordinatara kerult.
 FigureB a(z) 2 2 csempeben a(z) 193 118 koordinatara kerult.
 FigureB mozgásiranya beallitva Nyugat irányba.
 FigureB a(z) 2 2 csempeben a(z) 93 118 koordinatara kerult.
 FigureB mozgásiranya beallitva Nyugat irányba.
 FigureB kijarathoz ert, de nincs eleg kulcsa.
 FigureB atmegy Nyugat irányban egy másik csempebe.
 FigureB a(z) 2 1 csempeben a(z) 493 118 koordinatara kerult.

8.2.3 Teszteset3: Csempe mozgatása

- **Leírás**
Csempéket mozgatunk különböző sorrendben.
- **Ellenőrzött funkcionalitás, várható hibahelyek**
A csempék megfelelő mozgatásának letesztelése a cél. Hiba lehet az, hogy nem az a csempe mozdul el amelyiket irányítjuk, illetve olyan hiba is előfordulhat, hogy nem az üres csempe helyére toljuk az irányított csempét.
- **Bemenet (teszeset3.txt)**

```
loadCommands szint1.txt
beginWriteCommands teszteset3_output.txt
showTiles
moveTile N
moveTile W
moveTile S
moveTile E
moveTile S
endWriteCommands
exit
```

Elvárt kimenet (teszteset3_output.txt)

```
-----
EmptyTile      FullTile1
FullTile2      FullTile3
-----
```

```
-----
FullTile2      FullTile1
EmptyTile      FullTile3
-----
```

```
-----
FullTile2      FullTile1
FullTile3      EmptyTile
-----
```

```
-----
FullTile2      EmptyTile
FullTile3      FullTile1
-----
```

```
-----
EmptyTile      FullTile2
FullTile3      FullTile1
-----
```

Az adott irányba nem mozdulhatnak a csempek. A csempek jelenlegi állása:

```
-----
EmptyTile      FullTile2
FullTile3      FullTile1
-----
```

8.2.4 Teszteset4: A bábu át tud haladni a csempén

- **Leírás**

A bábu át akar haladni a csempén oldalt, a csempek illeszkednek, átmegy, majd folytatja az útját.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A teszteset célja azt az esetet kitesztelni, mikor illeszkedő csempéken halad át a figura. Hiba lehet az, ha illeszkedő csempéket nem illeszkedőnek véli a program, vagy ha rossz csempébe rakja át a bábút a program, vagy nem megfelelő kezdőkoordinátára.

- **Bemenet (teszeset4.txt)**

```
loadCommands szint2.txt
beginWriteCommands teszteset4_output.txt
moveFigure FigureA E
simulate
moveFigure FigureA E
simulate
moveFigure FigureA E
simulate
endWriteCommands
```

```
exit
```

- **Elvárt kimenet (teszteset4_output.txt)**

```
FigureA mozgásiranya beállítva Kelet irányba.
FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara kerult.
FigureA mozgásiranya beállítva Kelet irányba.
FigureA kulcsot talalt.
FigureA a(z) 2 1 csempeben a(z) 450 118 koordinatara kerult.
FigureA mozgásiranya beállítva Kelet irányba.
FigureA kijarathoz ert, de nincs eleg kulcsa.
FigureA atmegy Kelet irányba egy másik csempebe.
FigureA a(z) 2 2 csempeben a(z) 50 118 koordinatara kerult.
```

8.2.5 Teszteset5: A bábu nem tud áthaladni a csempén

- **Leírás**

A bábu át akar haladni a csempén oldalt, de nem illeszkednek.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A teszteset célja azt az esetet kitesztelni, amikor nem illeszkedő csempéken halad át a figura. Hiba lehet az, ha a nem illeszkedő csempéket illeszkedőnek véli a program, vagy ha olyankor is meghal a bábu, ha nem alsó élnél éri el a nem illeszkedő csempék szélét.

- **Bemenet (teszteset5.txt)**

```
loadCommands szint1.txt
beginWriteCommands teszteset5_output.txt
showTiles
moveTile N
moveFigure FigureA E
simulate
moveFigure FigureA E
simulate
moveFigure FigureA E
simulate
endWriteCommands
exit
```

- **Elvárt kimenet (teszteset5_output.txt)**

```
-----
EmptyTile          FullTile1
FullTile2          FullTile3
-----

FullTile2          FullTile1
EmptyTile          FullTile3
-----

FigureA mozgásiranya beállítva Kelet irányba.
FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara kerult.
FigureA mozgásiranya beállítva Kelet irányba.
FigureA kulcsot talalt.
FigureA a(z) 2 1 csempeben a(z) 450 118 koordinatara kerult.
```

```
FigureA mozgásiranya beallitva Kelet irányba.
FigureA Kelet irányba nem tudja elhagyni a csempet!
FigureA a(z) 2 1 csempeben a(z) 495 118 koordinatara kerult.
teszteset5.txt betoltese sikeres.
```

8.2.6 Teszteset6: A bábu meghal

- **Leírás**

A bábu át akar menni a csempe alsó keretén, de azok nem illeszkednek, a bábu meghal, majd a bábu visszatér a kiinduló pontjára.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A teszteset célja azt az esetet kitesztelni, amikor az alul nem illeszkedő csempéken halad át a figura. Hiba lehet az, ha az alul nem illeszkedő csempéket illeszkedőnek véli a program, vagy ha nem hal meg a bábu, hanem csak a csempe alján marad.

- **Bemenet (teszteset6.txt)**

```
loadCommands szint1.txt
beginWriteCommands teszteset6_output.txt
moveFigure FigureA W
simulate
simulate
simulate
endWriteCommands
exit
```

- **Elvárt kimenet (teszteset6_output.txt)**

```
FigureA mozgásiranya beallitva Nyugat irányba.
FigureA elkezdett zuhanni.
FigureA a(z) 2 1 csempeben a(z) 174 95 koordinatara kerult.
FigureA a(z) 2 1 csempeben a(z) 174 49 koordinatara kerult.
FigureA meghalt, majd a(z) 250 118 koordinatan ujraeledt.
teszteset6.txt betoltese sikeres.
```

8.2.7 Teszteset7: Bábu kulcsot talál, majd eléri a kijáratot és kimegy rajta.

- **Leírás**

Megy a csempében a bábu, megtalál egy kulcsot, felveszi, majd elmegy a kijáratához és a kijárat leellenőrzi és kiengedi.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ellenőrizzük azt, hogy tényleg teljesíthető-e a pálya azzal, hogy az egy kulcsos kijáratot elérjük egy kulcs felvétele után. Ha egy kulcs felvétele után sem tudjuk teljesíteni a pályát, akkor hibát találtunk.

- **Bemenet (teszteset7.txt)**

```
loadCommands szint1.txt
beginWriteCommands teszteset7_output.txt
moveFigure FigureA E
simulate
moveFigure FigureA E
simulate
moveFigure FigureA E
simulate
endWriteCommands
```

exit

- **Elvárt kimenet (teszteset7_output.txt)**

FigureA mozgásiránya beállítva Kelet irányba.
FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara került.
FigureA mozgásiránya beállítva Kelet irányba.
FigureA kulcsot talált.
FigureA a(z) 2 1 csempeben a(z) 450 118 koordinatara került.
FigureA mozgásiránya beállítva Kelet irányba.
FigureA atmegy Kelet irányban egy másik csempebe.
FigureA kijarathoz ért, es van eleg kulcsa. A szint teljesítve!

Teli csempe hozzáadva.

Uj terepalakzat szerkesztése.

Kontrollpont 0 100 hozzáadva.

Kontrollpont 100 100 hozzáadva.

Kontrollpont 100 300 hozzáadva.

Kontrollpont 0 300 hozzáadva.

Terepalakzat szerkesztése befejezve.

Uj terepalakzat szerkesztése.

Kontrollpont 200 0 hozzáadva.

Kontrollpont 300 0 hozzáadva.

Kontrollpont 300 100 hozzáadva.

Kontrollpont 200 100 hozzáadva.

Terepalakzat szerkesztése befejezve.

Uj terepalakzat szerkesztése.

Kontrollpont 400 100 hozzáadva.

Kontrollpont 500 100 hozzáadva.

Kontrollpont 500 300 hozzáadva.

Kontrollpont 400 300 hozzáadva.

Terepalakzat szerkesztése befejezve.

Uj terepalakzat szerkesztése.

Kontrollpont 200 300 hozzáadva.

Kontrollpont 300 300 hozzáadva.

Kontrollpont 300 400 hozzáadva.

Kontrollpont 200 400 hozzáadva.

Terepalakzat szerkesztése befejezve.

Teli csempe szerkesztése befejezve.

Teli csempe hozzáadva.

Uj terepalakzat szerkesztése.

Kontrollpont 0 100 hozzáadva.

Kontrollpont 100 100 hozzáadva.

Kontrollpont 100 300 hozzáadva.

Kontrollpont 0 300 hozzáadva.

Terepalakzat szerkesztése befejezve.

Uj terepalakzat szerkesztése.

Kontrollpont 200 0 hozzáadva.

Kontrollpont 300 0 hozzáadva.

Kontrollpont 300 100 hozzáadva.

Kontrollpont 200 100 hozzáadva.

Terepalakzat szerkesztése befejezve.

Uj terepalakzat szerkesztese.
Kontrollpont 400 100 hozzaadva.
Kontrollpont 500 100 hozzaadva.
Kontrollpont 500 300 hozzaadva.
Kontrollpont 400 300 hozzaadva.
Terepalakzat szerkesztese befejezve.
Uj terepalakzat szerkesztese.
Kontrollpont 200 300 hozzaadva.
Kontrollpont 300 300 hozzaadva.
Kontrollpont 300 400 hozzaadva.
Kontrollpont 200 400 hozzaadva.
Terepalakzat szerkesztese befejezve.
Teli csempe szerkesztese befejezve.
Teli csempe hozzaadva.
Uj terepalakzat szerkesztese.
Kontrollpont 200 0 hozzaadva.
Kontrollpont 500 0 hozzaadva.
Kontrollpont 500 100 hozzaadva.
Kontrollpont 200 100 hozzaadva.
Terepalakzat szerkesztese befejezve.
Kulcs 450 150 helyen hozzaadva.
Teli csempe szerkesztese befejezve.

Teli csempe hozzaadva.
Uj terepalakzat szerkesztese.
Kontrollpont 0 0 hozzaadva.
Kontrollpont 500 0 hozzaadva.
Kontrollpont 500 200 hozzaadva.
Kontrollpont 300 200 hozzaadva.
Kontrollpont 300 100 hozzaadva.
Kontrollpont 0 100 hozzaadva.
Terepalakzat szerkesztese befejezve.
Uj terepalakzat szerkesztese.
Kontrollpont 200 300 hozzaadva.
Kontrollpont 500 300 hozzaadva.
Kontrollpont 500 400 hozzaadva.
Kontrollpont 200 400 hozzaadva.
Terepalakzat szerkesztese befejezve.
1 kulcsos kijarat 50 123 helyen hozzaadva.
Teli csempe szerkesztese befejezve.

Teli csempe hozzaadva.
Uj terepalakzat szerkesztese.
Kontrollpont 0 100 hozzaadva.
Kontrollpont 100 100 hozzaadva.
Kontrollpont 100 300 hozzaadva.
Kontrollpont 0 300 hozzaadva.
Terepalakzat szerkesztese befejezve.
Uj terepalakzat szerkesztese.
Kontrollpont 200 0 hozzaadva.
Kontrollpont 300 0 hozzaadva.
Kontrollpont 300 100 hozzaadva.
Kontrollpont 200 100 hozzaadva.
Terepalakzat szerkesztese befejezve.
Uj terepalakzat szerkesztese.
Kontrollpont 400 100 hozzaadva.

```

        Kontrollpont 500 100 hozzaadva.
        Kontrollpont 500 300 hozzaadva.
        Kontrollpont 400 300 hozzaadva.
    Terepalakzat szerkesztese befejezve.
    Uj terepalakzat szerkesztese.
        Kontrollpont 200 300 hozzaadva.
        Kontrollpont 300 300 hozzaadva.
        Kontrollpont 300 400 hozzaadva.
        Kontrollpont 200 400 hozzaadva.
    Terepalakzat szerkesztese befejezve.
    Teli csempe szerkesztese befejezve.

    FigureA a(z) 2 1 pozicioju teli csempeben 250 118
    pozicioba hozzaadva.
    FigureB a(z) 2 2 pozicioju teli csempeben 250 218
    pozicioba hozzaadva.
    Szint szerkesztese befejezve.
    szint2.txt betoltese sikeres.
    teszteset7.txt betoltese sikeres.

```

8.2.8 Teszteset8: Bábu áthalad egy felszedett kulcson, kijáratot talál, de nincs nála elég kulcs.

- **Leírás**

A bábu megy, áthalad egy kulcson, majd talál egy kijáratot, de továbbhalad, mert nincs nála elég kulcs.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A második számú pályánkon két kulcs van ezért egy kulcs nem elég a pálya teljesítéséhez. Ha mégis elég lenne, akkor hibás a programunk.

- **Bemenet (teszteset8.txt)**

```

loadCommands szint2.txt
beginWriteCommands teszteset8_output.txt
moveFigure FigureA E
simulate
moveFigure FigureA E
simulate
moveFigure FigureA E
simulate
endWriteCommands
exit

```

- **Elvárt kimenet (teszteset8_output.txt)**

```

FigureA mozgásiranya beallitva Kelet irányba.
FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara kerult.
FigureA mozgásiranya beallitva Kelet irányba.
FigureA kulcsot talalt.
FigureA a(z) 2 1 csempeben a(z) 450 118 koordinatara kerult.
FigureA mozgásiranya beallitva Kelet irányba.
FigureA atmegy Kelet irányban egy másik csempebe.
FigureA kijarathoz ert, de nincs eleg kulcsa.
FigureA a(z) 2 2 csempeben a(z) 50 112 koordinatara kerult.
teszteset8.txt betoltese sikeres.

```

8.2.9 Teszteset9: Két bábu találkozik.

- **Leírás**

A két bábu találkozik egymással, majd továbbmennek.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

Ha a két bábu találkozik, akkor csak elmennek egymás mellett. Ha valami más történik (például meghal az egyik bábu), akkor hibás a programunk.

- **Bemenet (teszteset9.txt)**

```
loadCommands szint2.txt
beginWriteCommands teszteset9_output.txt
moveFigure FigureA E
simulate
moveFigure FigureB W
simulate
moveFigure FigureB W
simulate
moveFigure FigureB W
simulate
moveFigure FigureB W
simulate
moveFigure FigureB W
simulate
moveFigure FigureA W
moveFigure FigureB E
simulate
endWriteCommands
exit
```

- **Elvárt kimenet (teszteset9_output.txt)**

```
FigureA mozgásiránya beállítva Kelet irányba.
FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara került.
FigureB mozgásiránya beállítva Nyugat irányba.
FigureB a(z) 2 2 csempeben a(z) 150 118 koordinatara került.
FigureB mozgásiránya beállítva Nyugat irányba.
FigureB kijarathoz ért, de nincs elég kulcsa.
FigureB a(z) 2 2 csempeben a(z) 50 118 koordinatara került.
FigureB mozgásiránya beállítva Nyugat irányba.
FigureB kulcsot talált.
FigureB atmegy Nyugat irányba egy másik csempebe.
FigureB a(z) 2 1 csempeben a(z) 450 118 koordinatara került.
FigureB mozgásiránya beállítva Nyugat irányba.
A két figura találkozott!
FigureB a(z) 2 1 csempeben a(z) 350 118 koordinatara került.
FigureB mozgásiránya beállítva Nyugat irányba.
FigureB a(z) 2 1 csempeben a(z) 250 118 koordinatara került.
FigureA mozgásiránya beállítva Nyugat irányba.
FigureB mozgásiránya beállítva Kelet irányba.
A két figura találkozott!
FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara került.
FigureB a(z) 2 1 csempeben a(z) 250 118 koordinatara került.
teszteset9.txt betöltése sikeres.
```


8.2.10 Teszteset10: A bábu felszedi az első kulcsot, zuhanás közben éri el a második kulcsot, majd a kijáratot.

- **Leírás**

A bábu zuhanás közben eléri a kulcsot, aztán megáll egy terepen kicsit tovább megy megint elkezd zuhanni és eléri a kijáratot.

- **Ellenőrzött funkcionalitás, várható hibahelyek**

A legfőbb ellenőrzött funkció az, hogy a bábu zuhanás közben is fel tudja-e venni a kulcsot. Ha nem veszi fel zuhanás közben, akkor hibás a programunk.

- **Bemenet (teszteset10.txt)**

```
loadCommands szint2.txt
beginWriteCommands teszteset10_output.txt
moveFigure FigureA E
simulate
moveFigure FigureA E
simulate
showTiles
moveTile N
moveTile W
moveFigure FigureA W
simulate
moveFigure FigureA W
simulate
moveFigure FigureA W
simulate
moveFigure FigureA W
simulate
moveFigure FigureA W
simulate
simulate
simulate
simulate
endWriteCommands
exit
```

- **Elvárt kimenet (teszteset10_output.txt)**

```
FigureA mozgásiránya beállítva Kelet irányba.
FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara kerult.
FigureA mozgásiránya beállítva Kelet irányba.
FigureA kulcsolt talalt.
FigureA a(z) 2 1 csempeben a(z) 450 118 koordinatara kerult.
-----
EmptyTile          FullTile1
FullTile2          FullTile3
-----
-----
FullTile2          FullTile1
EmptyTile          FullTile3
-----
-----
FullTile2          FullTile1
FullTile3          EmptyTile
-----
FigureA mozgásiránya beállítva Nyugat irányba.
FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara kerult.
```

FigureA mozgásiranya beállítva Nyugat irányba.
FigureA a(z) 2 1 csempeben a(z) 250 118 koordinatara kerult.
FigureA mozgásiranya beállítva Nyugat irányba.
FigureA elkezdett zuhanni.
FigureA a(z) 2 1 csempeben a(z) 174 95 koordinatara kerult.
FigureA mozgásiranya beállítva Nyugat irányba.
FigureA a(z) 2 1 csempeben a(z) 119 49 koordinatara kerult.
FigureA atmegy Del irányban egy másik csempebe.
FigureA a(z) 2 1 csempeben a(z) 64 395 koordinatara kerult.
FigureA kulcsot talalt.
FigureA a(z) 2 1 csempeben a(z) 64 349 koordinatara kerult.
FigureA a(z) 2 1 csempeben a(z) 64 303 kitaláljuk
koordinatara kerult.
FigureA a(z) 2 1 csempeben a(z) 64 257 kitaláljuk
koordinatara kerult.
FigureA a(z) 2 1 csempeben a(z) 64 211 kitaláljuk
koordinatara kerult.
FigureA kijarathoz ert, es van eleg kulcsa. A szint
teljesitve!
FigureA a(z) 2 1 csempeben a(z) 64 165 koordinatara kerult.
teszteset10.txt betoltese sikeres.

8.3 A tesztelést támogató programok tervei

Match program:

A teszteteset eredményét tároló fájl és az elvárt kimenetet tartalmazó file-t ezzel az egy egyszerű programmal szeretnénk összehasonlítani, amely visszaadja, ha azonos a két fájl tartalma, illetve ha nem egyeznek meg, akkor kiírja, hogy mely sorokkal volt gond.

Bemenet:

Az általunk elkészített match.exe képes ezt leellenőrizni, mely két parancssori paramétert vár. Első parancssori paraméter a teszteteset eredményét tartalmazó file, a második parancssori paraméter pedig az elvárt kimenetet tartalmazó file.

Kimenet:

Parancssorban karakteres felületen kiírja a program számunkra, hogy egyezik-e a két fájl tartalma. Ha nem, akkor kiírja soronként az elvárt kimenetet és emellett kiírja azt is, hogy a mi kimenetünkben mi található. Ezáltal könnyen megtalálható a hiba pontos helye és oka.

10. Prototípus beadása

10.1 Fordítási és futtatási útmutató

10.1.1 Fájllista

Fájl neve	Méret [byte]	Keletkezés ideje	Tartalom (mi kerül megvalósításra)
compile.bat	214	2012.04.17.	Lefordítja a forráskódot.
run.bat	181	2012.04.17.	Futtatja a prototípust.
Setup.bat	84	2012.04.17.	Beállítja, milyen java compilert és futtatókörnyezetet használunk.
src\Control\Controller.java	9 325	2012.04.17.	Forráskód.
src\Control\InputCommands.java	174	2012.04.17.	Forráskód.
src\Model\Application.java	22 395	2012.04.17.	Forráskód.
src\Model\Direction.java	4 538	2012.04.17.	Forráskód.
src\Model\EmptyTile.java	1 696	2012.04.17.	Forráskód.
src\Model\Exit.java	1 839	2012.04.17.	Forráskód.
src\Model\Field.java	1 294	2012.04.17.	Forráskód.
src\Model\Figure.java	3 625	2012.04.17.	Forráskód.
src\Model\Frame.java	1 928	2012.04.17.	Forráskód.
src\Model\FullTile.java	16 041	2012.04.17.	Forráskód.
src\Model\GameObject.java	1 096	2012.04.17.	Forráskód.
src\Model\Key.java	1 214	2012.04.17.	Forráskód.
src\Model\Level.java	7 382	2012.04.17.	Forráskód.
src\Model\LevelInputCommands.java	204	2012.04.17.	Forráskód.
src\Model\Tile.java	1 577	2012.04.17.	Forráskód.
src\Model\Type.java	367	2012.04.17.	Forráskód.
szint1.txt	1022	2012.04.17.	Előre megírt szint a tesztesetekhez.
szint2.txt	1 988	2012.04.17.	Előre megírt szint a tesztesetekhez.
teszteset1.txt	89	2012.04.17.	Bemenet.
teszteset1_output.txt	1 921	2012.04.17.	A teszt eredménye.
teszteset2.txt	616	2012.04.17.	Bemenet.
teszteset2_output.txt	5 334	2012.04.17.	A teszt eredménye.
teszteset3.txt	160	2012.04.17.	Bemenet.
teszteset3_output.txt	746	2012.04.17.	A teszt eredménye.
teszteset4.txt	185	2012.04.17.	Bemenet.
teszteset4_output.txt	557	2012.04.17.	A teszt eredménye.
teszteset5.txt	208	2012.04.17.	Bemenet.
teszteset5_output.txt	773	2012.04.17.	A teszt eredménye.
teszteset6.txt	141	2012.04.17.	Bemenet.
teszteset6_output.txt	528	2012.04.17.	A teszt eredménye.
teszteset7.txt	185	2012.04.17.	Bemenet.
teszteset7_output.txt	557	2012.04.17.	A teszt eredménye.
teszteset8.txt	185	2012.04.17.	Bemenet.
teszteset8_output.txt	557	2012.04.17.	A teszt eredménye.
teszteset9.txt	335	2012.04.17.	Bemenet.

teszteset9_output.txt	1 472	2012.04.17.	A teszt eredménye.
teszteset10.txt	380	2012.04.17.	Bemenet.
teszteset10_output.txt	2 451	2012.04.17.	
teszteset1_elvart.txt	1 921	2012.04.17.	A teszteset kimenete.
teszteset2_elvart.txt	5 334	2012.04.17.	A teszteset kimenete.
teszteset3_elvart.txt	746	2012.04.17.	A teszteset kimenete.
teszteset4_elvart.txt	557	2012.04.17.	A teszteset kimenete.
teszteset5_elvart.txt	773	2012.04.17.	A teszteset kimenete.
teszteset6_elvart.txt	528	2012.04.17.	A teszteset kimenete.
teszteset7_elvart.txt	557	2012.04.17.	A teszteset kimenete.
teszteset8_elvart.txt	557	2012.04.17.	A teszteset kimenete.
teszteset9_elvart.txt	1472	2012.04.17.	A teszteset kimenete.
teszteset10_elvart.txt	2 334	2012.04.17.	A teszteset kimenete.

10.1.2 Fordítás

Az egyszerűsítés, és a hibák elkerülésének érdekében a fordítást a *compile.bat* nevű batch fájl végzi. A java útvonal beállítása után (pl. set PATH=„c:\Program Files\Java\jdk1.6.0_18\bin”) célszerű végrehajtani. A fordítás eredménye a */bin* könyvtárban lesz látható: elkészülnek a *.class fájlok.

Az előállításához szükséges feltétel a legalább 1.6-os verziójú Java JDK kit megléte.

10.1.3 Futtatás

A futtatás egy batch fájl elindításával lehetséges (run.bat), természetesen csak az után, hogy az alkalmazás hiba nélkül lefordításra került.

Futtatáskor a program felszólítja a felhasználót, hogy válasszon a tesztesetek közül. A megfelelő (1-től 10-ig) kiválasztása után a teszteset neve_out.txt fájlban kapjuk meg a teszt eredményét. Ezeket vethetjük össze az elvárt eredményekkel.

10.2 Tesztek jegyzőkönyvei

10.2.1 Teszteset#1

Tesztelő neve	Szabó Csaba
Teszt időpontja	2012.04.12.

10.2.2 Teszteset#2

Tesztelő neve	Berki Ramóna
Teszt időpontja	2012.04.12.

10.2.3 Teszteset#3

Tesztelő neve	Kiss Beáta
Teszt időpontja	2012.04.15.

10.2.4 Teszteset#4

Tesztelő neve	Kiss Beáta
Teszt időpontja	2012.04.15.

10.2.5 Teszteset#5

Tesztelő neve	Berki Ramóna
Teszt időpontja	2012.04.15.

10.2.6 Teszteset#6

Tesztelő neve	Szabó Csaba
Teszt időpontja	2012.04.17.
Teszt eredménye	Meghalt a bábu majd eltűnt.
Lehetséges hiba ok	Újraéledéskor nem volt beállítva a start csempébe a bábu referenciája.
Változtatások	Beállítottuk.

10.2.7 Teszteset#7

Tesztelő neve	Halász Antal István
Teszt időpontja	2012.04.17.
Teszt eredménye	A bábu nem volt egyenletes.
Lehetséges hiba ok	- Nem vizsgáltuk, ha a bábu kerettel ütközött. - Rosszul adtuk össze az erőhatásokat - A már nem ható erőket nem vettük ki a fizikai modell egyenletéből.
Változtatások	A hiányzó/hibás részeket kijavítottuk.

10.2.8 Teszteset#8

Tesztelő neve	Gerlei Balázs
Teszt időpontja	2012.04.12.

10.2.9 Teszteset#9

Tesztelő neve	Gerlei Balázs
Teszt időpontja	2012.04.17.
Teszt eredménye	Nem írta ki a program, ha két bábu találkozott.
Lehetséges hiba ok	Elfelejtettük megírni a találkozást vizsgáló függvényt.
Változtatások	Megírtuk.

10.2.10 Teszteset#10

Tesztelő neve	Halász Antal István
Teszt időpontja	2012.04.17.
Teszt eredménye	A bábu a csempeváltáskor eltűnt.
Lehetséges hiba ok	Kimaradt a bábu referenciát átállítása a másik csempébe.
Változtatások	A referenciát beállítottuk.

10.3 Értékelés

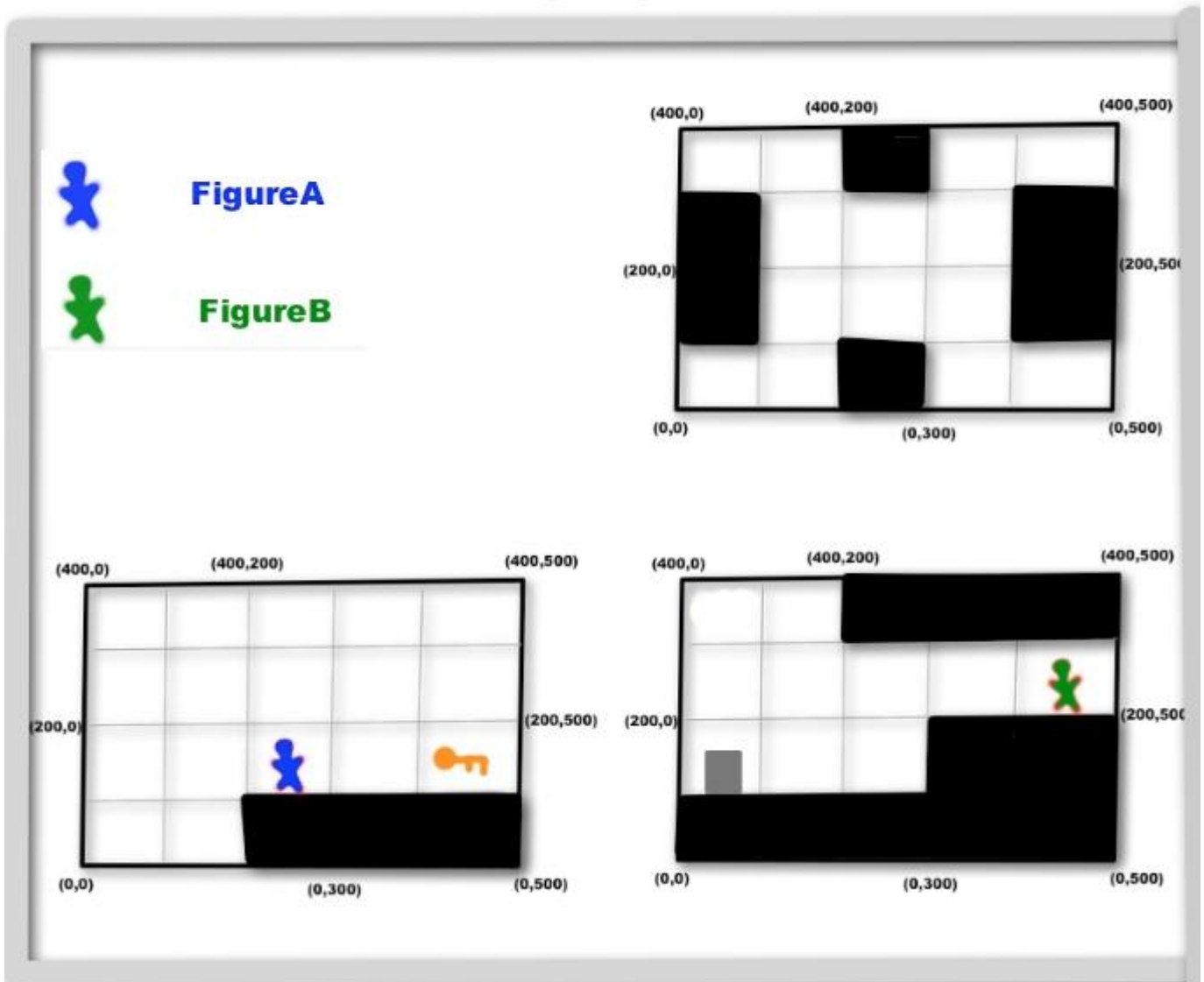
A projekten végzett eddigi munka (munkaórák alapján kerekítve):

Tag neve	Munka százalékban
Berki Ramóna	14
Gerlei Balázs	19
Halász Antal István	19
Kiss Beáta	19
Szabó Csaba	29

10. Részletes tervek kiegészítő

A szintek:

szint 1



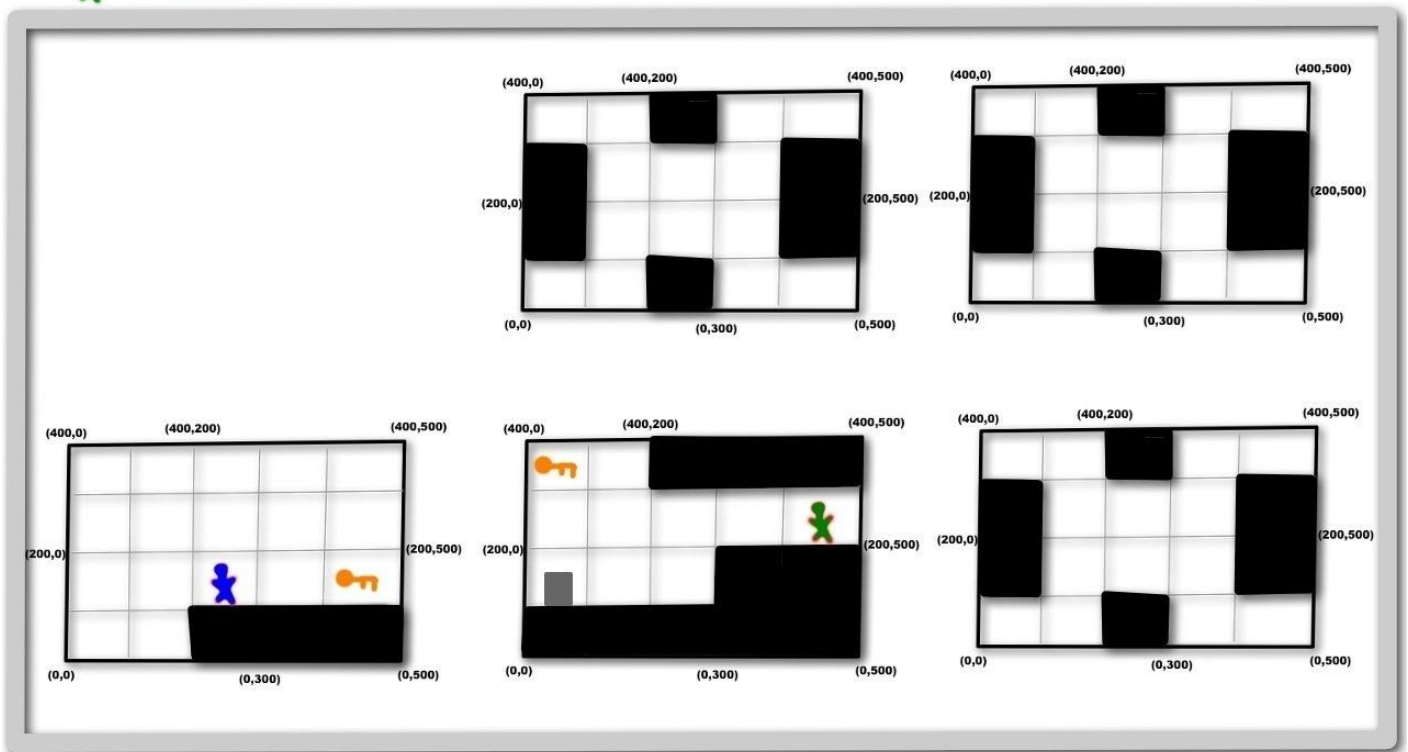


FigureA



FigureB

Szint 2

**A tesztesetek:****Teszteset#1:**

Leírás: Betöltünk egy szintet (jelen esetben az 1-et) egy adott fájlból: itt betöltődik négy csempe (egy üres és három teli), rajtuk a terepek, egy kulcs a 2.1-es csempébe¹, egy kijárat a 2.2-es csempébe illetve FigureA és FigureB a 2.1-es és a 2.2-es csempébe.

Elvárt kimenet:

- level1 szint létrehozása sikeres.
- Teli csempe hozzáadva.
- Új terepalakzat szerkesztése.
- Kontrollpont 0 100 hozzáadva.
- Kontrollpont 100 100 hozzáadva.
- Kontrollpont 100 300 hozzáadva.
- Kontrollpont 0 300 hozzáadva.
- Terepalakzat szerkesztése befejezve.
- Új terepalakzat szerkesztése.
- Kontrollpont 200 0 hozzáadva.
- Kontrollpont 300 0 hozzáadva.
- Kontrollpont 300 100 hozzáadva.
- Kontrollpont 200 100 hozzáadva.

¹ A csempék számozása 1-essel kezdődik és mátrixszerűen növekszik. Például a szint1-nél 1.1 1.2 2.1 és 2.2-es csempék vannak.

Terepalakzat szerkesztése befejezve.
 Új terepalakzat szerkesztése.
 Kontrollpont 400 100 hozzáadva.
 Kontrollpont 500 100 hozzáadva.
 Kontrollpont 500 300 hozzáadva.
 Kontrollpont 400 300 hozzáadva.
 Terepalakzat szerkesztése befejezve.
 Új terepalakzat szerkesztése.
 Kontrollpont 200 300 hozzáadva.
 Kontrollpont 300 300 hozzáadva.
 Kontrollpont 300 400 hozzáadva.
 Kontrollpont 200 400 hozzáadva.
 Terepalakzat szerkesztése befejezve.
 Teli csempe szerkesztése befejezve.
 Teli csempe hozzáadva.
 Új terepalakzat szerkesztése.
 Kontrollpont 200 0 hozzáadva.
 Kontrollpont 500 0 hozzáadva.
 Kontrollpont 500 100 hozzáadva.
 Kontrollpont 200 100 hozzáadva.
 Terepalakzat szerkesztése befejezve.
 Kulcs 450 113 helyen hozzáadva.
 Teli csempe szerkesztése befejezve.
 Teli csempe hozzáadva.
 Új terepalakzat szerkesztése.
 Kontrollpont 0 0 hozzáadva.
 Kontrollpont 500 0 hozzáadva.
 Kontrollpont 500 200 hozzáadva.
 Kontrollpont 300 200 hozzáadva.
 Kontrollpont 300 100 hozzáadva.
 Kontrollpont 0 100 hozzáadva.
 Terepalakzat szerkesztése befejezve.
 Új terepalakzat szerkesztése.
 Kontrollpont 200 300 hozzáadva.
 Kontrollpont 500 300 hozzáadva.
 Kontrollpont 500 400 hozzáadva.
 Kontrollpont 200 400 hozzáadva.
 Terepalakzat szerkesztése befejezve.
 1 kulcsos kijárat 50 123 helyen hozzáadva.
 Teli csempe szerkesztése befejezve.
 FigureA a(z) 2 1 pozícióju teli csempeben 250 118 pozícióba hozzáadva
 FigureB a(z) 2 2 pozícióju teli csempeben 450 218 pozícióba hozzáadva
 szint1.txt betöltése sikeres.

Ábra:

A tesztesethez a *szint1* kép tartozik.

Teszteset2:

Leírás: FigureA felugrik, majd előre megy és újból felugrik. FigureB elindul balra majd, mikor le akar ugrani beveri fejét a felső terepbe, elhalad egy kijárat előtt, de mivel nincs

kulcsa nem történik semmi, ezután átmegy a 2.2-es csempéből a 2.1-esbe és felveszi az ott lévő kulcsot.

Elvárt kimenet:

FigureA mozgásiránya beállítva E irányba.
FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara került.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA mozgásiránya beállítva W irányba.
FigureA a(z) 2 1 csempeben a(z) 250 118 koordinatara került.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA mozgásiránya beállítva N irányba.
FigureA elkezdett ugrani.
FigureA elkezdett zuhanni.
FigureA a(z) 2 1 csempeben a(z) 250 164 koordinatara került.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA elkezdett zuhanni.
FigureA a(z) 2 1 csempeben a(z) 250 163 koordinatara került.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA talajnak utkozott!
FigureA a(z) 2 1 csempeben a(z) 250 118 koordinatara került.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA talajnak utkozott!
FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA mozgásiránya beállítva NE irányba.
FigureA elkezdett ugrani.
FigureA elkezdett zuhanni.
FigureA a(z) 2 1 csempeben a(z) 349 172 koordinatara került.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA talajnak utkozott!
FigureA a(z) 2 1 csempeben a(z) 347 118 koordinatara került.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA maradt a(z) 2 1 csempeben a(z) 347 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA maradt a(z) 2 1 csempeben a(z) 347 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA mozgásiránya beállítva NW irányba.
FigureA elkezdett ugrani.
FigureA elkezdett zuhanni.
FigureA a(z) 2 1 csempeben a(z) 247 172 koordinatara került.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

FigureA talajnak utkozott!
FigureA a(z) 2 1 csempeben a(z) 248 118 koordinatara kerult.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA talajnak utkozott!
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
FigureB mozgásiranya beallitva E irányba.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB Kelet irányban nem tudja elhagyni a csempet!
FigureB a(z) 2 2 csempeben a(z) 495 218 koordinatara kerult.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 495 218 koordinatan.
FigureB mozgásiranya beallitva W irányba.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB a(z) 2 2 csempeben a(z) 395 218 koordinatara kerult.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 395 218 koordinatan.
FigureB mozgásiranya beallitva NW irányba.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB elkezdett ugrani.
FigureB falnak utkozott!
FigureB a(z) 2 2 csempeben a(z) 331 281 koordinatara kerult.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB talajnak utkozott!
FigureB a(z) 2 2 csempeben a(z) 373 218 koordinatara kerult.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB talajnak utkozott!
FigureB maradt a(z) 2 2 csempeben a(z) 373 218 koordinatan.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 373 218 koordinatan.
FigureB mozgásiranya beallitva W irányba.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB a(z) 2 2 csempeben a(z) 273 218 koordinatara kerult.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB elkezdett zuhanni.
FigureB a(z) 2 2 csempeben a(z) 273 172 koordinatara kerult.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB talajnak utkozott!
FigureB a(z) 2 2 csempeben a(z) 273 118 koordinatara kerult.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB maradt a(z) 2 2 csempeben a(z) 273 118 koordinatan.
FigureB mozgásiranya beallitva W irányba.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB a(z) 2 2 csempeben a(z) 173 118 koordinatara kerult.
FigureB mozgásiranya beallitva W irányba.
FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinatan.
FigureB kijarathoz ert, de nincs eleg kulcsa.
FigureB a(z) 2 2 csempeben a(z) 73 118 koordinatara kerult.

FigureB mozgásirányba beállítva W irányba.
 FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinátán.
 FigureB kijáratához ért, de nincs elég kulcsa.
 FigureB atmegy Nyugat irányban egy másik csempebe.
 FigureB a(z) 2 1 csempeben a(z) 495 118 koordinátára került.
 FigureB mozgásirányba beállítva W irányba.
 FigureA maradt a(z) 2 1 csempeben a(z) 248 118 koordinátán.
 FigureB kulcsot talált.
 FigureB a(z) 2 1 csempeben a(z) 395 118 koordinátára került.

Ábra:

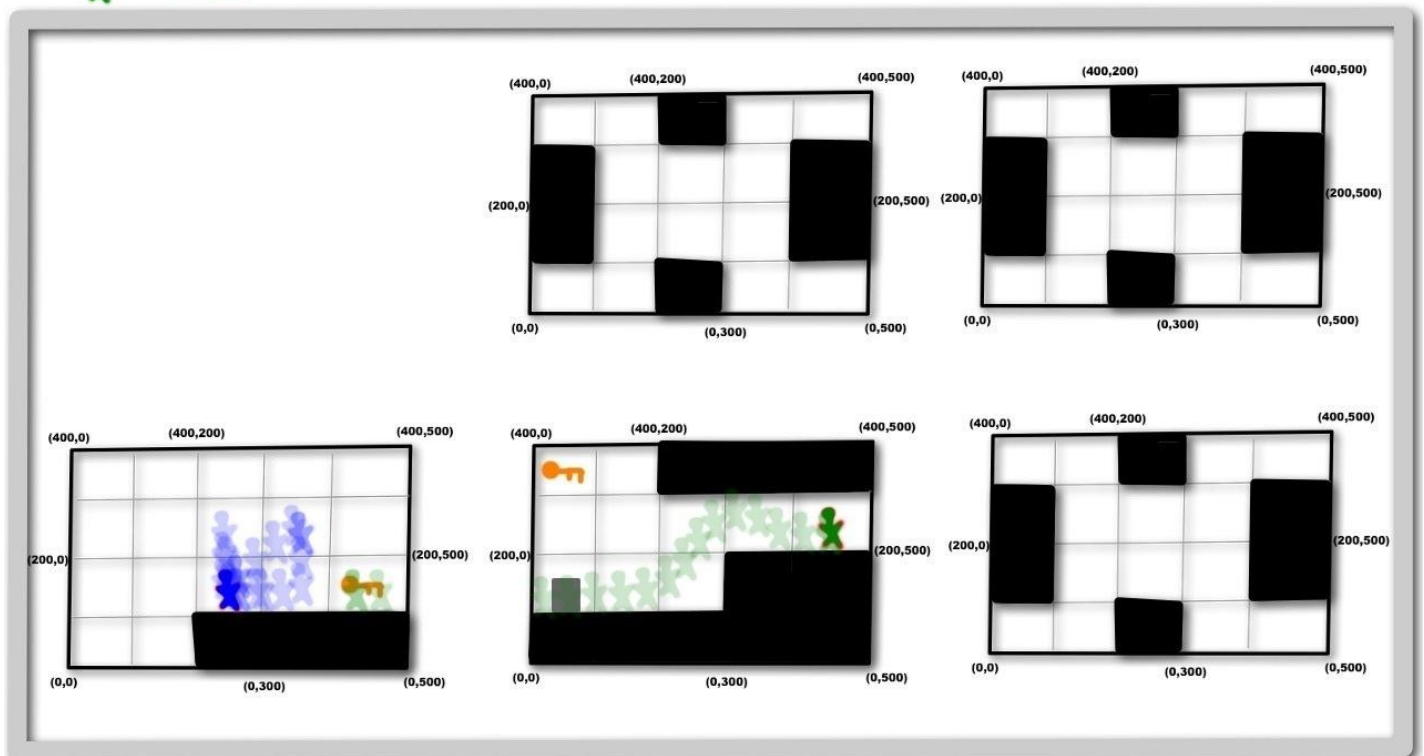


FigureA



FigureB

Teszt eset 2



Teszt eset#3:

Leírás: Csempeket mozgatunk különböző sorrendben, az elvárt kimenetnek megfelelően.

Elvárt kimenet:

```

-----
EmptyTile    FullTile1
FullTile2    FullTile3
-----

```

```

-----
FullTile2    FullTile1
EmptyTile    FullTile3
-----
-----

```

FullTile2	FullTile1
FullTile3	EmptyTile

FullTile2	EmptyTile
FullTile3	FullTile1

EmptyTile	FullTile2
FullTile3	FullTile1

Az adott irányba nem mozdulhatnak a csempek. A csempek jelenlegi állása:

EmptyTile	FullTile2
FullTile3	FullTile1

Ábra: A tesztesethez nem szükséges ábra, mert a kimenet egyértelmű.

Tesztese#t4:

Leírás: FigureA át akar haladni a 2.1-es csempe jobb oldalán, a keretek illeszkednek és átmegy a 2.2-es csempébe.

Elvárt kimenet:

FigureA mozgásiránya beállítva E irányba.
 FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.
 FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiránya beállítva E irányba.
 FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara került.
 FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiránya beállítva E irányba.
 FigureA kulcsot talált.
 FigureA a(z) 2 1 csempeben a(z) 450 118 koordinatara került.
 FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiránya beállítva E irányba.
 FigureA atmegy Kelet irányban egy másik csempebe.
 FigureA a(z) 2 2 csempeben a(z) 5 118 koordinatara került.
 FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

Ábra:

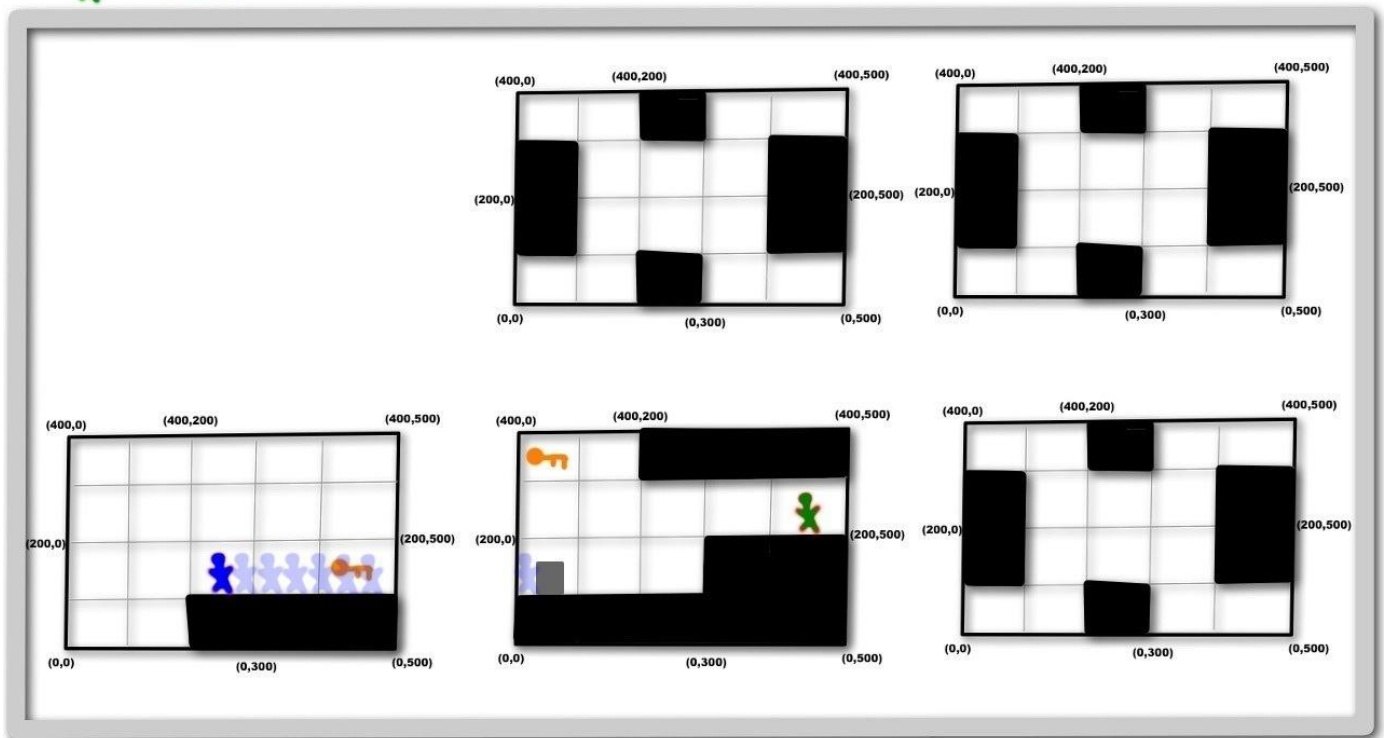


FigureA



FigureB

Teszteset 4



Teszteset#5:

Leírás: Az 1.1-es és a 2.1-es csempe helyet cserél. FigureA elindul jobbra, talál egy kulcsot, majd át akar menni az 1.2-es csempébe, de a keretek nem illeszkednek így marad az 1.1-es csempében.

Elvárt kimenet:

```
-----
EmptyTile    FullTile1
FullTile2    FullTile3
-----
```

```
-----
FullTile2    FullTile1
EmptyTile    FullTile3
-----
```

FigureA mozgásiránya beállítva E irányba.

FigureA maradt a(z) 1 1 csempében a(z) 250 118 koordinatan.

FigureB maradt a(z) 2 2 csempében a(z) 450 218 koordinatan.

FigureA mozgásiránya beállítva E irányba.

FigureA a(z) 1 1 csempében a(z) 350 118 koordinatara kerult.

FigureB maradt a(z) 2 2 csempében a(z) 450 218 koordinatan.

FigureA mozgásiránya beállítva E irányba.

FigureA kulcsot talalt.

FigureA a(z) 1 1 csempében a(z) 450 118 koordinatara kerult.

FigureB maradt a(z) 2 2 csempében a(z) 450 218 koordinatan.

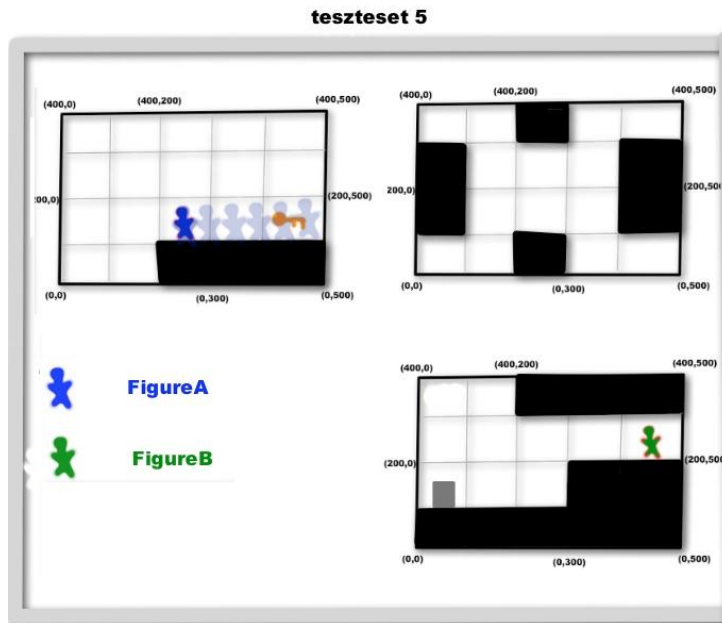
FigureA mozgásiránya beállítva E irányba.

FigureA Kelet irányban nem tudja elhagyni a csempet!

FigureA a(z) 1 1 csempeben a(z) 495 118 koordinatara kerult.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

Ábra:



Tesztese#t6:

Leírás: FigureA elindul balra, de elfogy alatta a talaj és az aktuális csempéje alatt sincsen másik illeszkedő csempe, ezért meghal és a kezdőkoordinátán újraéled.

Elvárt kimenet:

FigureA mozgásiránya beállítva W irányba.

FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

FigureA elkezdett zuhanni.

FigureA a(z) 2 1 csempeben a(z) 145 93 koordinatara kerult.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

FigureA a(z) 2 1 csempeben a(z) 191 22 koordinatara kerult.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

FigureA meghalt, majd a 250 118 koordinatan ujraeledt.

FigureA a(z) 2 1 csempeben a(z) 250 118 koordinatara kerult.

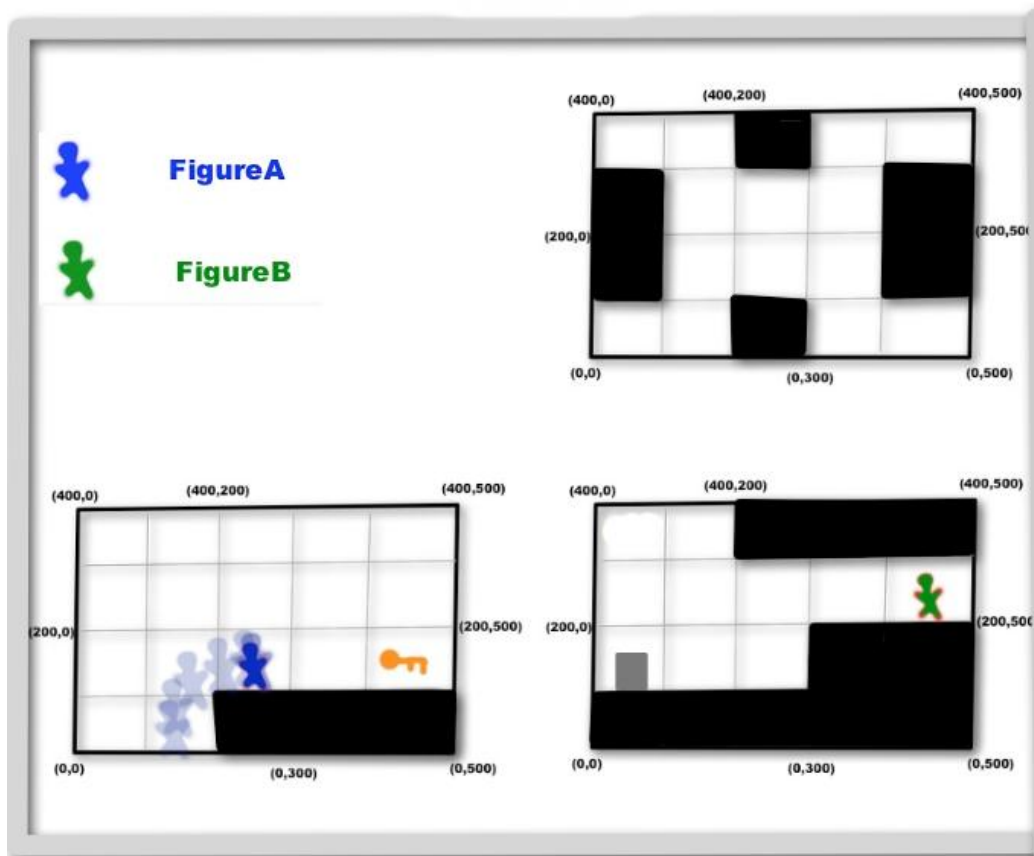
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

Ábra:

teszteset 6



Teszteset#7:

Leírás: FigureA elindul jobbra majd talál egy kulcsot, ezután átmegy a 2.1-es csempéből a 2.2-esbe ahol megtalálja a kijáratot. Nála van a szükséges kulcs, így teljesíti az 1. szintet és betöltődik a 2. szint.

Elvárt kimenet:

FigureA mozgásiránya beállítva E irányba.
 FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara került.
 FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiránya beállítva E irányba.
 FigureA kulcsot talalt.
 FigureA a(z) 2 1 csempeben a(z) 450 118 koordinatara került.
 FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiránya beállítva E irányba.
 FigureA atmegy Kelet irányban egy másik csempebe.
 FigureA a(z) 2 2 csempeben a(z) 5 118 koordinatara került.
 FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiránya beállítva E irányba.
 FigureA kijarathoz ert, es van eleg kulcsa. A szint teljesítve!
 level2 szint létrehozasa sikeres.
 Teli csempe hozzáadva.
 Új terepalakzat szerkesztése.
 Kontrollpont 0 100 hozzáadva.
 Kontrollpont 100 100 hozzáadva.

Kontrollpont 100 300 hozzáadva.
Kontrollpont 0 300 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 200 0 hozzáadva.
Kontrollpont 300 0 hozzáadva.
Kontrollpont 300 100 hozzáadva.
Kontrollpont 200 100 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 400 100 hozzáadva.
Kontrollpont 500 100 hozzáadva.
Kontrollpont 500 300 hozzáadva.
Kontrollpont 400 300 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 200 300 hozzáadva.
Kontrollpont 300 300 hozzáadva.
Kontrollpont 300 400 hozzáadva.
Kontrollpont 200 400 hozzáadva.
Terepalakzat szerkesztése befejezve.
Teli csempe szerkesztése befejezve.
Teli csempe hozzáadva.
Új terepalakzat szerkesztése.
Kontrollpont 0 100 hozzáadva.
Kontrollpont 100 100 hozzáadva.
Kontrollpont 100 300 hozzáadva.
Kontrollpont 0 300 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 200 0 hozzáadva.
Kontrollpont 300 0 hozzáadva.
Kontrollpont 300 100 hozzáadva.
Kontrollpont 200 100 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 400 100 hozzáadva.
Kontrollpont 500 100 hozzáadva.
Kontrollpont 500 300 hozzáadva.
Kontrollpont 400 300 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 200 300 hozzáadva.
Kontrollpont 300 300 hozzáadva.
Kontrollpont 300 400 hozzáadva.
Kontrollpont 200 400 hozzáadva.
Terepalakzat szerkesztése befejezve.
Teli csempe szerkesztése befejezve.
Teli csempe hozzáadva.
Új terepalakzat szerkesztése.

Kontrollpont 200 0 hozzáadva.
Kontrollpont 500 0 hozzáadva.
Kontrollpont 500 100 hozzáadva.
Kontrollpont 200 100 hozzáadva.
Terepalakzat szerkesztése befejezve.
Kulcs 450 113 helyen hozzáadva.
Teli csempe szerkesztése befejezve.
Teli csempe hozzáadva.
Új terepalakzat szerkesztése.
Kontrollpont 0 0 hozzáadva.
Kontrollpont 500 0 hozzáadva.
Kontrollpont 500 200 hozzáadva.
Kontrollpont 300 200 hozzáadva.
Kontrollpont 300 100 hozzáadva.
Kontrollpont 0 100 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 200 300 hozzáadva.
Kontrollpont 500 300 hozzáadva.
Kontrollpont 500 400 hozzáadva.
Kontrollpont 200 400 hozzáadva.
Terepalakzat szerkesztése befejezve.
Kulcs 50 350 helyen hozzáadva.
2 kulcsos kijarat 50 123 helyen hozzáadva.
Teli csempe szerkesztése befejezve.
Teli csempe hozzáadva.
Új terepalakzat szerkesztése.
Kontrollpont 0 100 hozzáadva.
Kontrollpont 100 100 hozzáadva.
Kontrollpont 100 300 hozzáadva.
Kontrollpont 0 300 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 200 0 hozzáadva.
Kontrollpont 300 0 hozzáadva.
Kontrollpont 300 100 hozzáadva.
Kontrollpont 200 100 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 400 100 hozzáadva.
Kontrollpont 500 100 hozzáadva.
Kontrollpont 500 300 hozzáadva.
Kontrollpont 400 300 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 200 300 hozzáadva.
Kontrollpont 300 300 hozzáadva.
Kontrollpont 300 400 hozzáadva.
Kontrollpont 200 400 hozzáadva.
Terepalakzat szerkesztése befejezve.

Teli csempe szerkesztése befejezve.

FigureA a(z) 2 1 pozícióju teli csempeben 250 118 pozícióba hozzáadva

FigureB a(z) 2 2 pozícióju teli csempeben 450 218 pozícióba hozzáadva

Szint szerkesztése befejezve.

FigureA a(z) 2 2 csempeben a(z) 25 118 koordinatara kerult.

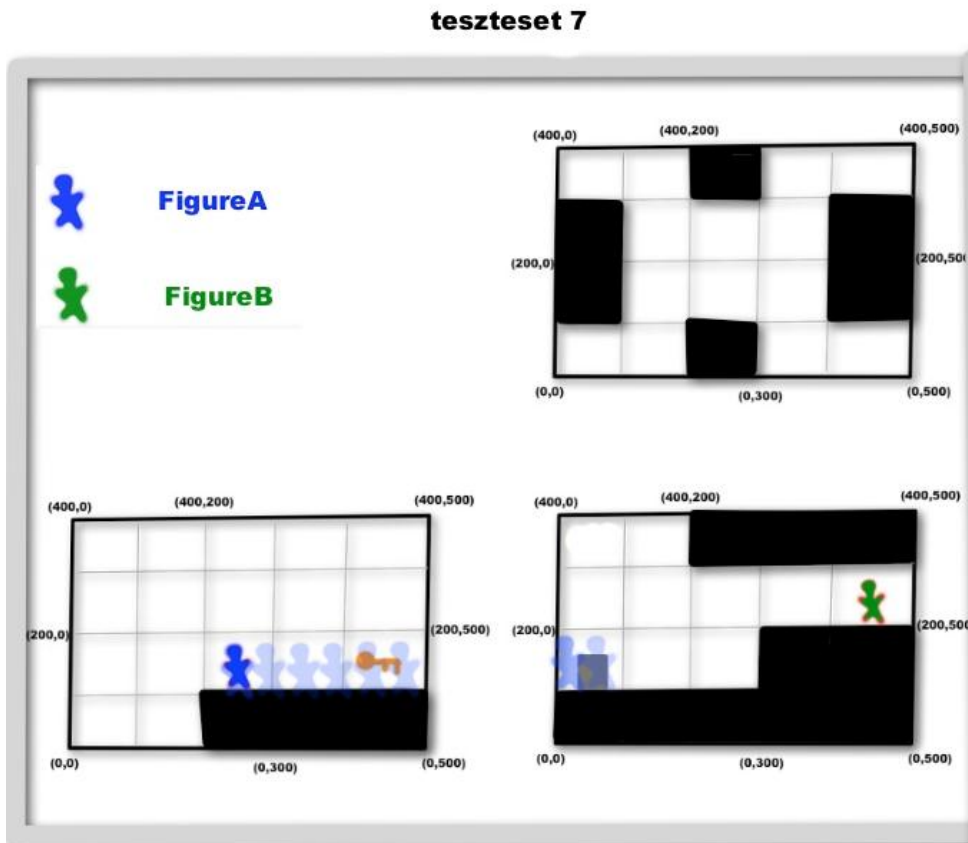
FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

FigureA mozgásiranya beallitva E irányba.

FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara kerult.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

Ábra:



Teszteset#8:

Leírás: FigureA elindul jobbra, talál egy kulcsot és tovább megy a 2.1-es csempéből a 2.2-esbe, ahol talál egy kijáratot is, de nincs nála elég kulcs (kettő kell a szint teljesítéséhez).

Elvárt kimenet:

FigureA mozgásiranya beallitva E irányba.

FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara kerult.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

FigureB mozgásiranya beallitva W irányba.

FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.

FigureB a(z) 2 2 csempeben a(z) 350 218 koordinatara kerult.

FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.

FigureB maradt a(z) 2 2 csempeben a(z) 350 218 koordinatan.

FigureB mozgásiranya beallitva W irányba.

FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
 FigureB a(z) 2 2 csempeben a(z) 250 218 koordinatara kerult.
 FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
 FigureB elkezdett zuhanni.
 FigureB a(z) 2 2 csempeben a(z) 250 172 koordinatara kerult.
 FigureB mozgásiranya beallitva W irányba.
 FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
 FigureB talajnak utkozott!
 FigureB a(z) 2 2 csempeben a(z) 218 118 koordinatara kerult.
 FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
 FigureB falnak utkozott!
 FigureB a(z) 2 2 csempeben a(z) 294 118 koordinatara kerult.
 FigureB mozgásiranya beallitva W irányba.
 FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
 FigureB a(z) 2 2 csempeben a(z) 194 118 koordinatara kerult.
 FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
 FigureB maradt a(z) 2 2 csempeben a(z) 194 118 koordinatan.
 FigureB mozgásiranya beallitva W irányba.
 FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
 FigureB a(z) 2 2 csempeben a(z) 94 118 koordinatara kerult.
 FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
 FigureB maradt a(z) 2 2 csempeben a(z) 94 118 koordinatan.
 FigureA mozgásiranya beallitva W irányba.
 FigureB mozgásiranya beallitva W irányba.
 FigureA a(z) 2 1 csempeben a(z) 250 118 koordinatara kerult.
 FigureB kijarathoz ert, de nincs eleg kulcsa.
 FigureB atmegy Nyugat irányban egy masik csempebe.
 FigureB a(z) 2 1 csempeben a(z) 495 118 koordinatara kerult.
 FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.
 FigureB maradt a(z) 2 1 csempeben a(z) 495 118 koordinatan.
 FigureB mozgásiranya beallitva W irányba.
 FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.
 FigureB kulcsot talalt.
 FigureB a(z) 2 1 csempeben a(z) 395 118 koordinatara kerult.
 FigureB mozgásiranya beallitva W irányba.
 FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.
 FigureB a(z) 2 1 csempeben a(z) 295 118 koordinatara kerult.
 FigureA mozgásiranya beallitva E irányba.
 FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara kerult.
 FigureB maradt a(z) 2 1 csempeben a(z) 295 118 koordinatan.

Ábra:

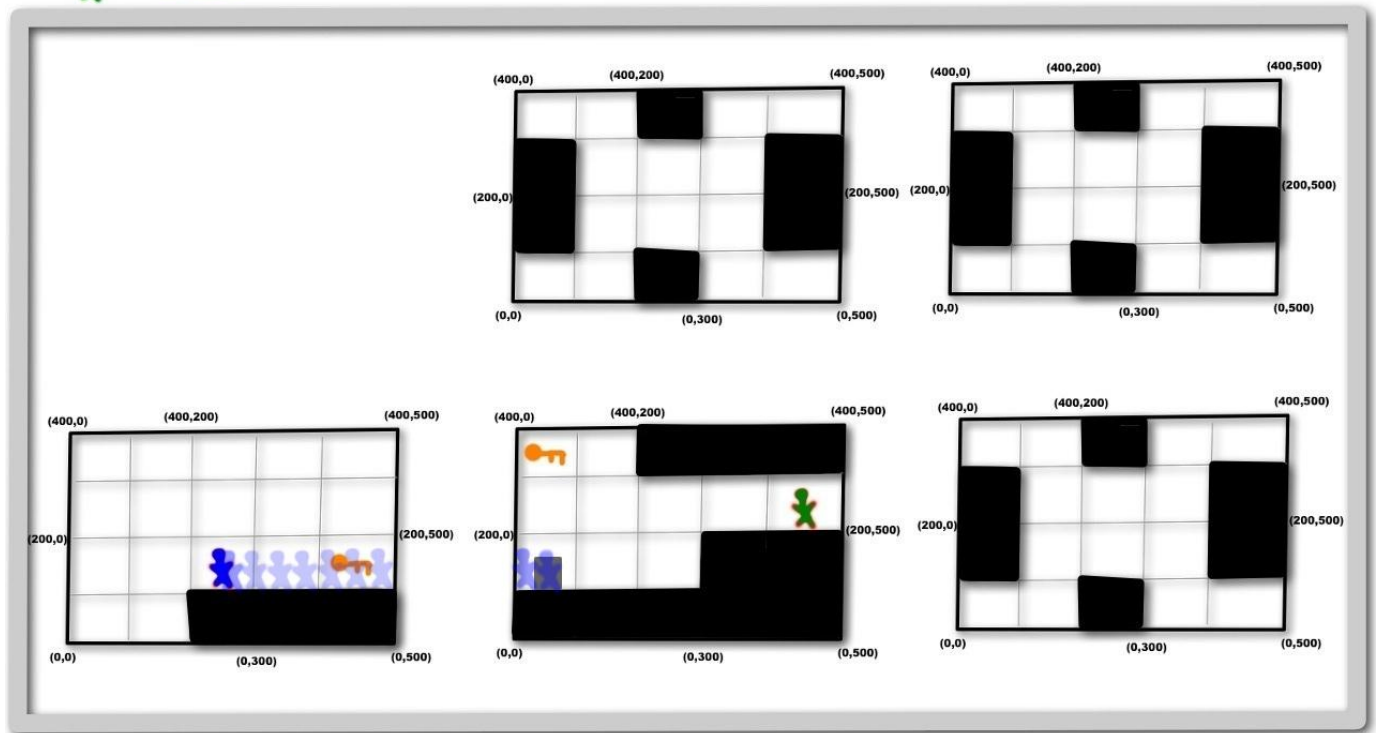


FigureA



FigureB

teszteset 8



Teszteset#9:

Leírás: FigureB elindul jobbra ahol talál egy kijáratot, de nem történik semmi, mert nincs nála kulcs, majd átmegy a 2.2-es csempeből a 2.1-esbe, ahol talál egy kulcsot is. Ezután elhalad az álló FigureA előtt, közben nem történik semmi sem (nincs reakció a két bábu között).

Elvárt kimenet:

- FigureA mozgásiránya beállítva E irányba.
- FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara került.
- FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
- FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
- FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.
- FigureB mozgásiránya beállítva W irányba.
- FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
- FigureB a(z) 2 2 csempeben a(z) 350 218 koordinatara került.
- FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
- FigureB maradt a(z) 2 2 csempeben a(z) 350 218 koordinatan.
- FigureB mozgásiránya beállítva W irányba.
- FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
- FigureB a(z) 2 2 csempeben a(z) 250 218 koordinatara került.
- FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.
- FigureB elkezdett zuhanni.
- FigureB a(z) 2 2 csempeben a(z) 250 172 koordinatara került.
- FigureB mozgásiránya beállítva W irányba.
- FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.

FigureB talajnak utkozott!

FigureB a(z) 2 2 csempeben a(z) 218 118 koordinatara kerult.

FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.

FigureB falnak utkozott!

FigureB a(z) 2 2 csempeben a(z) 294 118 koordinatara kerult.

FigureB mozgásiranya beallitva W irányba.

FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.

FigureB a(z) 2 2 csempeben a(z) 194 118 koordinatara kerult.

FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.

FigureB maradt a(z) 2 2 csempeben a(z) 194 118 koordinatan.

FigureB mozgásiranya beallitva W irányba.

FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.

FigureB a(z) 2 2 csempeben a(z) 94 118 koordinatara kerult.

FigureA maradt a(z) 2 1 csempeben a(z) 350 118 koordinatan.

FigureB maradt a(z) 2 2 csempeben a(z) 94 118 koordinatan.

FigureA mozgásiranya beallitva W irányba.

FigureB mozgásiranya beallitva W irányba.

FigureA a(z) 2 1 csempeben a(z) 250 118 koordinatara kerult.

FigureB kijarathoz ert, de nincs eleg kulcsa.

FigureB atmegy Nyugat irányban egy masik csempebe.

FigureB a(z) 2 1 csempeben a(z) 495 118 koordinatara kerult.

FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.

FigureB maradt a(z) 2 1 csempeben a(z) 495 118 koordinatan.

FigureB mozgásiranya beallitva W irányba.

FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.

FigureB kulcsot talalt.

FigureB a(z) 2 1 csempeben a(z) 395 118 koordinatara kerult.

FigureB mozgásiranya beallitva W irányba.

FigureA maradt a(z) 2 1 csempeben a(z) 250 118 koordinatan.

FigureB a(z) 2 1 csempeben a(z) 295 118 koordinatara kerult.

FigureA mozgásiranya beallitva E irányba.

FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara kerult.

FigureB maradt a(z) 2 1 csempeben a(z) 295 118 koordinatan.

Ábra:

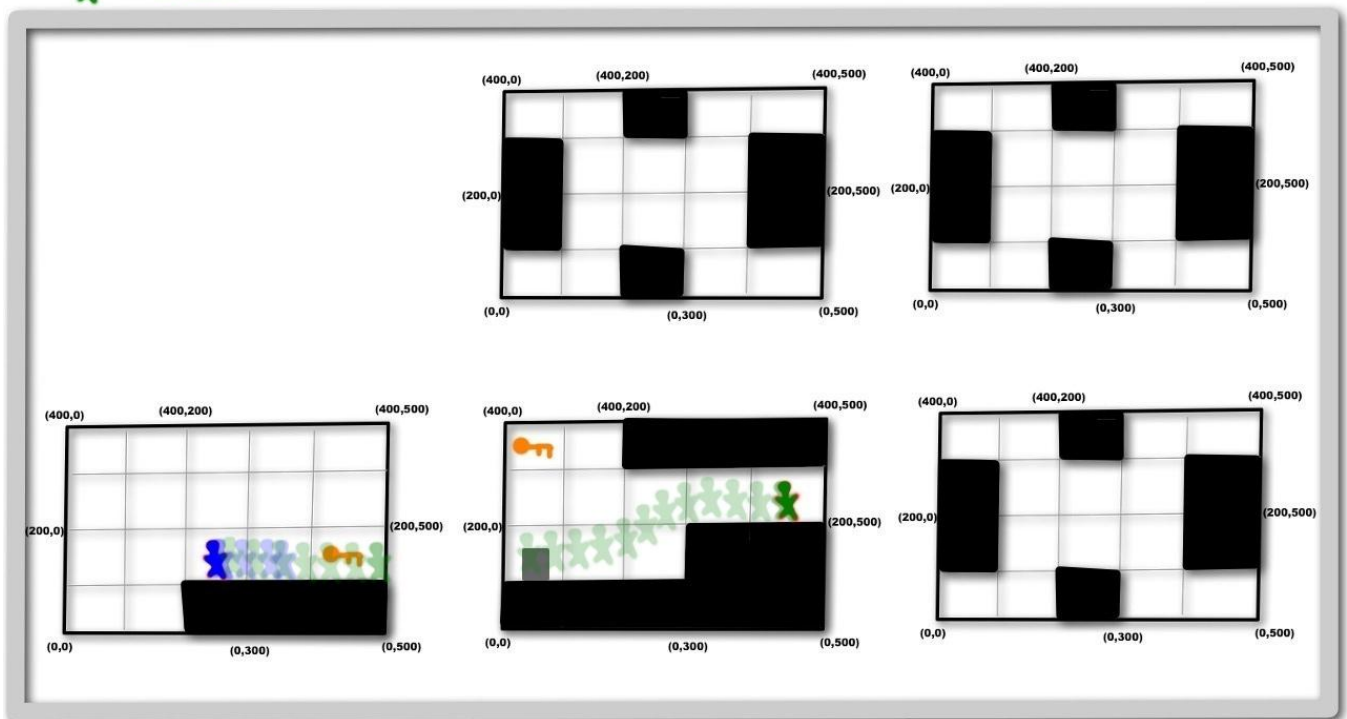


FigureA



FigureB

teszteset 9



Teszteset#10:

Leírás: Először az 1.1-es és a 2.1-es majd az új 2.1-es és a 2.2-es csempe cserélődik meg. Ezután FigureA elindul jobbra, felvesz egy kucsot majd balra megy ahol beleugrik az alatta lévő csempebe. Zuhanás közben: felvesz még egy kulcsot és eléri a kijáratot, ezzel teljesíti az adott pálya és betöltődik a következő.

Elvárt kimenet:

FigureA mozgásiránya beállítva E irányba.

FigureA a(z) 2 1 csempeben a(z) 350 118 koordinatara került.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

FigureA mozgásiránya beállítva E irányba.

FigureA kulcsot talalt.

FigureA a(z) 2 1 csempeben a(z) 450 118 koordinatara került.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

FigureA maradt a(z) 2 1 csempeben a(z) 450 118 koordinatan.

FigureB maradt a(z) 2 2 csempeben a(z) 450 218 koordinatan.

```
-----
EmptyTile    FullTile1    FullTile2
FullTile3    FullTile4    FullTile5
-----
```

```
-----
FullTile3    FullTile1    FullTile2
EmptyTile    FullTile4    FullTile5
-----
```

```
-----
FullTile3    FullTile1    FullTile2
FullTile4    EmptyTile    FullTile5
-----
```

FigureA mozgásiranya beallitva W irányba.
 FigureA a(z) 1 1 csempeben a(z) 350 118 koordinatara kerult.
 FigureB maradt a(z) 2 1 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiranya beallitva W irányba.
 FigureA a(z) 1 1 csempeben a(z) 250 118 koordinatara kerult.
 FigureB maradt a(z) 2 1 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiranya beallitva W irányba.
 FigureA elkezdett zuhanni.
 FigureA a(z) 1 1 csempeben a(z) 145 93 koordinatara kerult.
 FigureB maradt a(z) 2 1 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiranya beallitva W irányba.
 FigureA a(z) 1 1 csempeben a(z) 36 22 koordinatara kerult.
 FigureB maradt a(z) 2 1 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiranya beallitva W irányba.
 FigureA atmegy Del irányban egy másik csempebe.
 FigureA a(z) 2 1 csempeben a(z) 30 381 koordinatara kerult.
 FigureB maradt a(z) 2 1 csempeben a(z) 450 218 koordinatan.
 FigureA kulcsot talalt.
 FigureA Nyugat irányban nem tudja elhagyni a csempet!
 FigureA a(z) 2 1 csempeben a(z) 5 316 koordinatara kerult.
 FigureB maradt a(z) 2 1 csempeben a(z) 450 218 koordinatan.
 FigureA talajnak utkozott!
 FigureA a(z) 2 1 csempeben a(z) 5 118 koordinatara kerult.
 FigureB maradt a(z) 2 1 csempeben a(z) 450 218 koordinatan.
 FigureA maradt a(z) 2 1 csempeben a(z) 5 118 koordinatan.
 FigureB maradt a(z) 2 1 csempeben a(z) 450 218 koordinatan.
 FigureA mozgásiranya beallitva E irányba.
 FigureA kijarathoz ert, es van eleg kulcsa. A szint teljesitve!
 level2 szint leterhozasa sikeres.
 Teli csempe hozzaadva.
 Uj terepalakzat szerkesztese.
 Kontrollpont 0 100 hozzaadva.
 Kontrollpont 100 100 hozzaadva.
 Kontrollpont 100 300 hozzaadva.
 Kontrollpont 0 300 hozzaadva.
 Terepalakzat szerkesztese befejezve.
 Uj terepalakzat szerkesztese.
 Kontrollpont 200 0 hozzaadva.
 Kontrollpont 300 0 hozzaadva.
 Kontrollpont 300 100 hozzaadva.
 Kontrollpont 200 100 hozzaadva.
 Terepalakzat szerkesztese befejezve.
 Uj terepalakzat szerkesztese.
 Kontrollpont 400 100 hozzaadva.
 Kontrollpont 500 100 hozzaadva.
 Kontrollpont 500 300 hozzaadva.
 Kontrollpont 400 300 hozzaadva.
 Terepalakzat szerkesztese befejezve.
 Uj terepalakzat szerkesztese.

Kontrollpont 200 300 hozzáadva.
Kontrollpont 300 300 hozzáadva.
Kontrollpont 300 400 hozzáadva.
Kontrollpont 200 400 hozzáadva.
Terepalakzat szerkesztése befejezve.
Teli csempe szerkesztése befejezve.
Teli csempe hozzáadva.
Új terepalakzat szerkesztése.
Kontrollpont 0 100 hozzáadva.
Kontrollpont 100 100 hozzáadva.
Kontrollpont 100 300 hozzáadva.
Kontrollpont 0 300 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 200 0 hozzáadva.
Kontrollpont 300 0 hozzáadva.
Kontrollpont 300 100 hozzáadva.
Kontrollpont 200 100 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 400 100 hozzáadva.
Kontrollpont 500 100 hozzáadva.
Kontrollpont 500 300 hozzáadva.
Kontrollpont 400 300 hozzáadva.
Terepalakzat szerkesztése befejezve.
Új terepalakzat szerkesztése.
Kontrollpont 200 300 hozzáadva.
Kontrollpont 300 300 hozzáadva.
Kontrollpont 300 400 hozzáadva.
Kontrollpont 200 400 hozzáadva.
Terepalakzat szerkesztése befejezve.
Teli csempe szerkesztése befejezve.
Teli csempe hozzáadva.
Új terepalakzat szerkesztése.
Kontrollpont 200 0 hozzáadva.
Kontrollpont 500 0 hozzáadva.
Kontrollpont 500 100 hozzáadva.
Kontrollpont 200 100 hozzáadva.
Terepalakzat szerkesztése befejezve.
Kulcs 450 113 helyen hozzáadva.
Teli csempe szerkesztése befejezve.
Teli csempe hozzáadva.
Új terepalakzat szerkesztése.
Kontrollpont 0 0 hozzáadva.
Kontrollpont 500 0 hozzáadva.
Kontrollpont 500 200 hozzáadva.
Kontrollpont 300 200 hozzáadva.
Kontrollpont 300 100 hozzáadva.
Kontrollpont 0 100 hozzáadva.
Terepalakzat szerkesztése befejezve.

Uj terepalakzat szerkesztése.
Kontrollpont 200 300 hozzáadva.
Kontrollpont 500 300 hozzáadva.
Kontrollpont 500 400 hozzáadva.
Kontrollpont 200 400 hozzáadva.
Terepalakzat szerkesztése befejezve.
Kulcs 50 350 helyen hozzáadva.
2 kulcsos kijarat 50 123 helyen hozzáadva.
Teli csempe szerkesztése befejezve.
Teli csempe hozzáadva.
Uj terepalakzat szerkesztése.
Kontrollpont 0 100 hozzáadva.
Kontrollpont 100 100 hozzáadva.
Kontrollpont 100 300 hozzáadva.
Kontrollpont 0 300 hozzáadva.
Terepalakzat szerkesztése befejezve.
Uj terepalakzat szerkesztése.
Kontrollpont 200 0 hozzáadva.
Kontrollpont 300 0 hozzáadva.
Kontrollpont 300 100 hozzáadva.
Kontrollpont 200 100 hozzáadva.
Terepalakzat szerkesztése befejezve.
Uj terepalakzat szerkesztése.
Kontrollpont 400 100 hozzáadva.
Kontrollpont 500 100 hozzáadva.
Kontrollpont 500 300 hozzáadva.
Kontrollpont 400 300 hozzáadva.
Terepalakzat szerkesztése befejezve.
Uj terepalakzat szerkesztése.
Kontrollpont 200 300 hozzáadva.
Kontrollpont 300 300 hozzáadva.
Kontrollpont 300 400 hozzáadva.
Kontrollpont 200 400 hozzáadva.
Terepalakzat szerkesztése befejezve.
Teli csempe szerkesztése befejezve.
FigureA a(z) 2 1 pozicioju teli csempeben 250 118 pozicioba hozzáadva
FigureB a(z) 2 2 pozicioju teli csempeben 450 218 pozicioba hozzáadva
Szint szerkesztése befejezve.
FigureA a(z) 2 1 csempeben a(z) 25 118 koordinatara kerult.
FigureB maradt a(z) 2 1 csempeben a(z) 450 218 koordinatan.

Ábra:

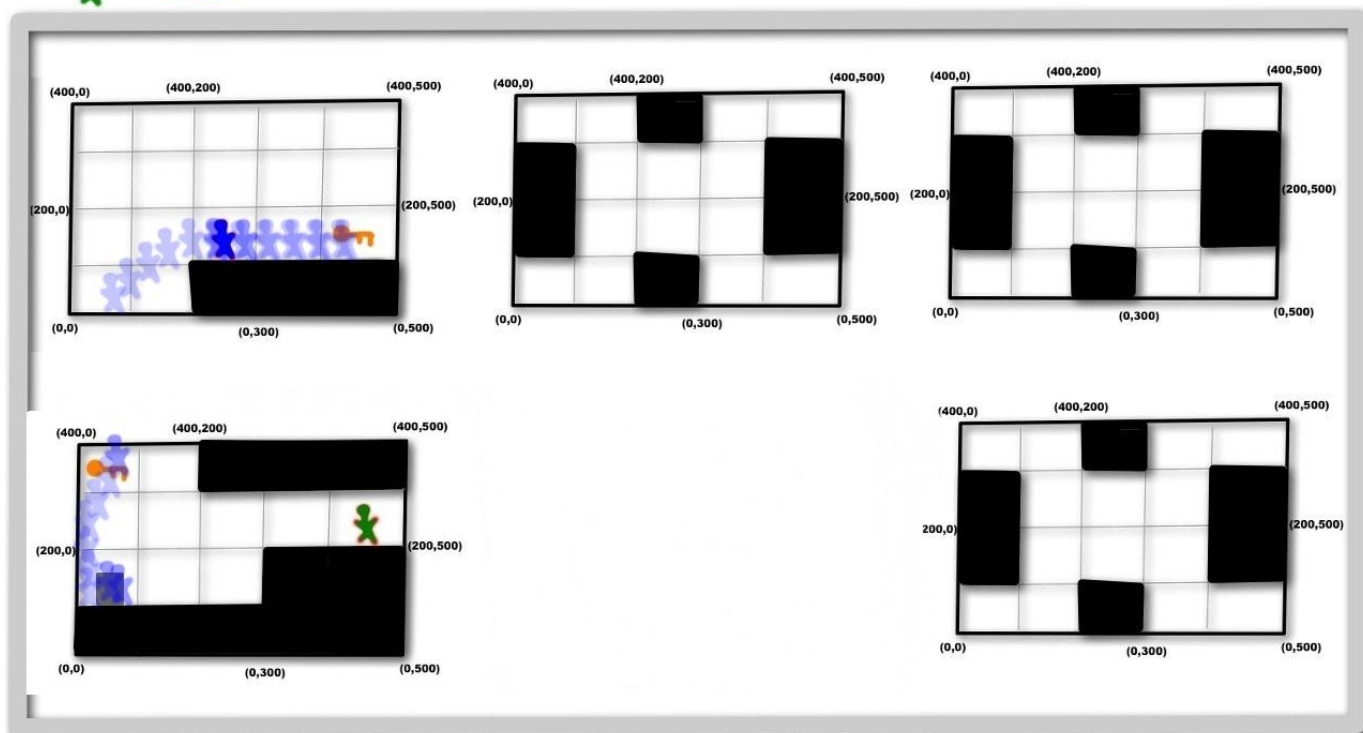


FigureA



FigureB

teszteset 10



11. Grafikus felület specifikációja

11.1 A grafikus interfész

A grafikus környezet megtervezésekor két dolgot vettünk figyelembe: egyrészt törekszünk arra, hogy a felület a felhasználók számára minél átláthatóbb legyen, másrészt pedig funkciókban legyen gazdag.

A program főmenüjében négy gomb található, amelyek a következő funkciókat valósítják meg: *Új játék*, *Játék folytatása*, *Súgó* és a *Kilépés*. A *Súgó* gombra kattintva egy általános leírást kapunk a kezelőfelület használatáról.



A csapat logójáról is készítettünk két látványtervet:



11.1.1 A felület működési elve

A felületi működése a Model View Controller elven alapszik, ahol teljesen elkülönül egymástól az alkalmazás modellje, a megjelenítés és a vezérlő logika.

- modell: az applikáció objektum, az adatok tárolásáért, kezeléséért felel
- view: a megjelenítésért felel
- controller: definiálja, hogy a felhasználói felület hogyan reagál a felhasználói bemenetre

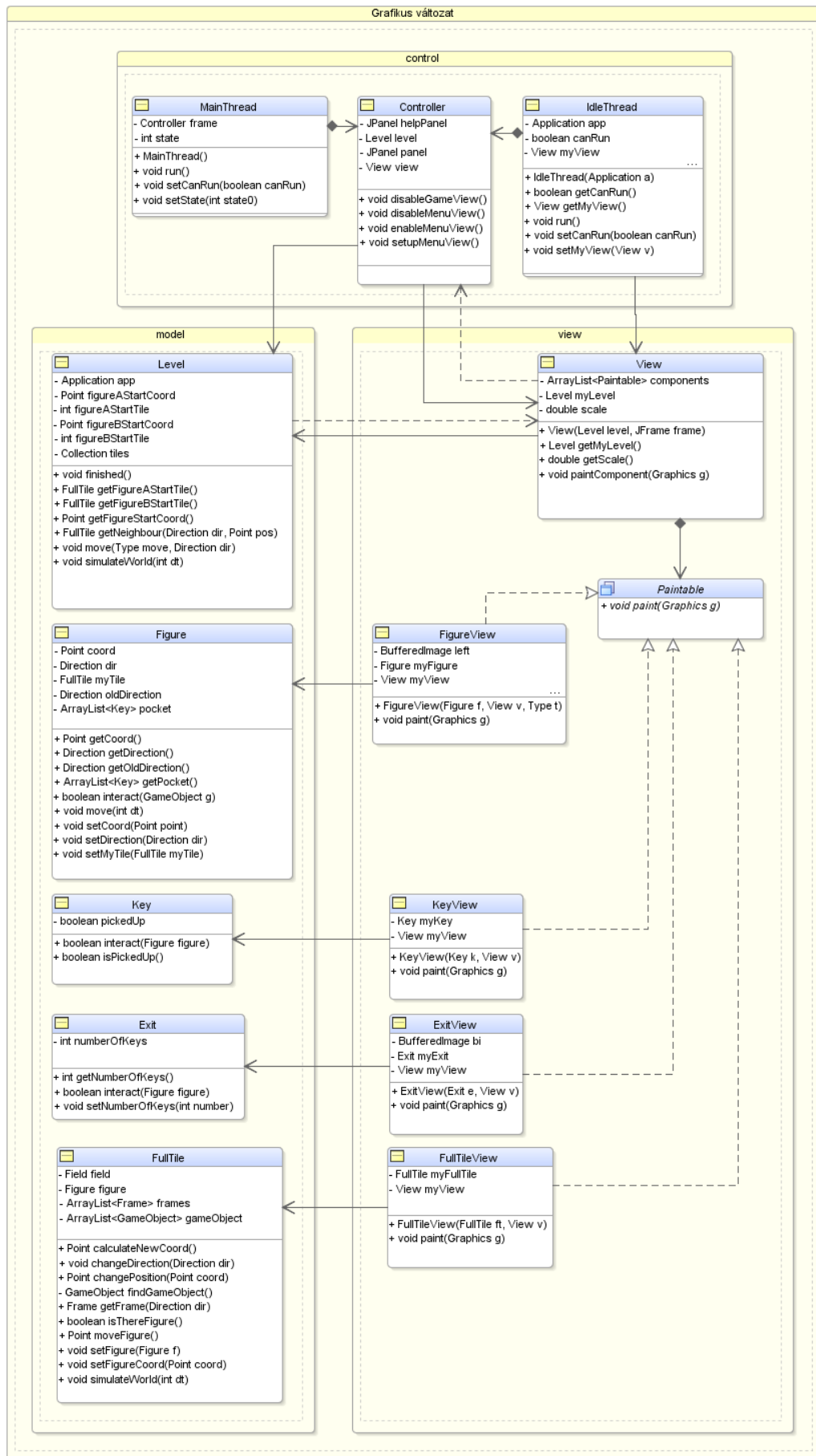
Ezen belül a push alapú megoldást tartottuk célszerűnek: vagyis a modell értesíti a felületet, hogy változás történt a felület pedig újrakirajolja önmagát a változásnak megfelelően.

Esetünkben: a Controller osztály értesíti a View-et, hogy megváltozott, az meghívja az egyes kirajzolandó objektumok paint() függvényét, amik lekérdezik az objektumok új helyzetét és kirajzolják őket.

11.1.2 A felület osztály-struktúrája

Annak érdekében, hogy modell minél jobban elkülönüljön a kirajzolástól szükségesnek láttuk néhány új osztály felvételét. A játék során öt osztály van, amelynek bizonyos elemet szeretnénk kirajzoltatni, ezek: Exit, Field, Figure, FullTile, Key. Ezeken kívül még a Level osztály áll kapcsolatban a felülettel. Ezeknek az osztályokhoz kapcsolódó osztályokat hoztunk létre (a Controllerhez csak új attribútumot és metódust adtunk).

Az új osztályok és kapcsolataik a régiekkel az alábbi osztálydiagramon láthatóak:



11.2 A grafikus objektumok felsorolása

11.2.1 ExitView

- **Felelősség**
A kijárat kirajzolásáért felelős.
- **Ősosztályok**
Nincs.
- **Interfészek**
Paintable
- **Attribútumok**
 - - **bi: BufferedImage**: A kijárat képét tároló objektum.
 - - **myExit: Exit**: Referencia a kirajzolandó kijáratra.
 - - **myView: View**: Referencia a View objektumra.
- **Metódusok**
 - + **ExitView(e:Exit, v:View)**: - Konstruktor.
 - + **paint(g:Graphics): void** Az adott komponenes kirajzolásáért felelős.

11.2.2 FigureView

- **Felelősség**
A bábu kirajzolásáért felelős.
- **Ősosztályok**
Nincs.
- **Interfészek**
Paintable.
- **Attribútumok**
 - - **left: BufferedImage**: A balra mozgó bábu képét tároló objektum.
 - - **myFigure: Figure**: Referencia a kirajzolandó bábura.
 - - **my View: View**: Referencia a View objektumra
 - - **right: BufferedImage**: A jobbra mozgó bábu képét tároló objektum.
 - - **stop: BufferedImage**: Az álló bábu képét tároló objektum.
- **Metódusok**
 - + **FigureView(f:Figure, v:View, t:Type)**: - Konstruktor.
 - + **paint(g:Graphics): void** - Az adott komponenes kirajzolásáért felelős.

11.2.3 FullTileView

- **Felelősség**
Egy teli csempe kirajzolásáért felelős
- **Ősosztályok**
Nincs.

- **Interfészek**
Paintable.
- **Attribútumok**
 - - **myFullTile: FullTile**: Referencia a kirajzolandó teli csempére.
 - - **myView: View**: Referencia a View objektumra.
- **Metódusok**
 - + **FullTileView(ft:FullTile, v:View)**: - Konstruktor.
 - + **paint(g:Graphics): void** - Az adott komponenes kirajzolásáért felelős.

11.2.4 IdleThread

- **Felelősség**
A játék futásának szálát megvalósító osztály. Vizsgálja a játék közbeni felhasználói bemenetet és vezérli a modellt.
- **Ősosztályok**
Thread
- **Interfészek**
Nincs.
- **Attribútumok**
 - - **app: Application**: Referencia a modellre.
 - - **canRun: boolean**: logikai változó, ha igaz, futhat a modell szimulációja.
 - - **myView: View**: Referencia az éppen betöltött szint View-jára.
 - - **time: long**: az előző szimuláció idejét tárolja.
- **Metódusok**
 - + **getCanRun():boolean**: Referenciát ad a canRun változóra.
 - + **getMyView():View**: Referenciát ad a myView változóra.
 - + **IdleThread(Application a)**: Konstruktor, ami létrehoz egy IdleThread objektumot egy modell alapján.
 - + **run(): void**: Az ősosztály override-olt metódusa, a szál futását tartalmazó kód.
 - + **setCanRun(boolean canRun):void**: Beállítja a canRun változót.
 - + **setMyView(View v):void**: Beállítja a myView változót.

11.2.5 KeyView

- **Felelősség**
A kulcs kirajzolásáért felelős.
- **Ősosztályok**
Nincs.
- **Interfészek**
Paintable.
- **Attribútumok**
 - - **found : BufferedImage**: A már felvett kulcs képét tároló objektum.

- - **myKey: Key**: Referencia a kirajzolandó kulcsra.
- - **myView: View**: Referencia a View objektumra.
- - **notFound : BufferedImage**: A még nem felvett kulcs képét tároló objektum.
- **Metódusok**
 - + **KeyView(k:Key, v:View)**: - Konstruktor.
 - + **paint(g:Graphics): void** - Az adott komponenes kirajzolásáért felelős.

11.2.6 MainThread

- **Felelősség**
Az ablakkezelést segítő osztály. Inicializálja az ablakot, szabályozza, hogy a játék vagy a menü látszódjon.
- **Ősosztályok**
Thread
- **Interfészek**
Nincs.
- **Attribútumok**
 - - **frame: Controller**: Referencia a játék ablakára.
 - - **state: int**: Értéke tükrözi, hogy menüben vagy játékban vagyunk-e (0=menü, 1=játék).
- **Metódusok**
 - + **MainThread()**: Konstruktor, ami létrehoz és inicializál egy MainThread objektumot.
 - + **run(): void**: Az őosztály override-olt metódusa, a szál futását tartalmazó kód.
 - + **setCanRun(boolean canRun):void**: Beállítja a canRun változót.
 - + **setState(int state0):void**: Beállítja a state változót.

11.2.7 Paintable Interface

- **Felelősség**
Interface mely összefogja a kirajzolandó objektumok kezelését.
- **Ősosztályok**
Nincs.
- **Interfészek**
Nincs.
- **Attribútumok**
Nincs.
- **Metódusok**
 - + **paint(g:Graphics): void** - Elvégzi az adott objektum kirajzolását.

11.2.8 View

- **Felelősség**
A View osztály felelőssége, hogy biztosítsa a megjelenítési környezetet a tartalmazott komponensek számára. Ez az osztály gondoskodik arról, hogy megfelelő időpontban és sorrendben meghívja a játék összes megjelenítéssel kapcsolatos objektum kirajzoló függvényét.
- **Ősosztályok**
JPanel
- **Interfészek**
Nincs.
- **Attribútumok**
 - - **components: ArrayList <IPaintable>** - Tartalmazza a kirajzolandó objektumokat.
 - - **myLevel: Level** - A kirajzolandó szint objektumra mutató referencia.
 - - **scale: double** - A megfelelő méretű kirajzoláshoz szükséges skálázási szám.
- **Metódusok**
 - + **getMyLevel()**: Referenciát ad vissza a myLevel objektumra.
 - + **getScale():double**: Visszaadja a skálázási számot.
 - + **paintComponent(g: Graphics): void** Ez a metódus végzi a felület tényleges újrarajzolását. Az összes tartalmazott objektum paint() metódusát meghívja a megfelelő sorrendben és időben.
 - + **View (level: Level, frame: JFrame)**: Konstuktor, ami szint alapján inicializálja a kirajzolandó objektumok tömbjét.

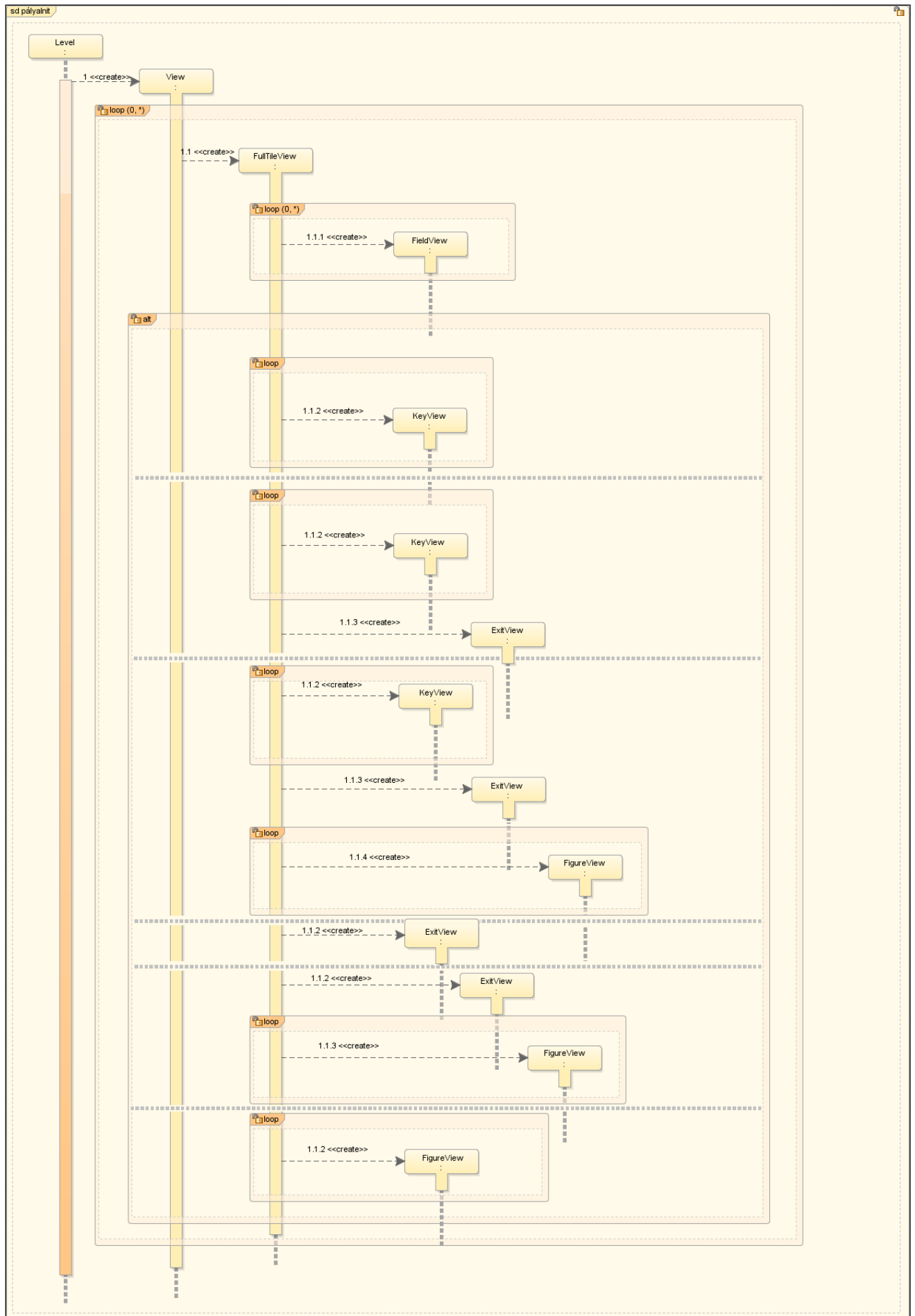
11.2.9 Controller (bővítés)

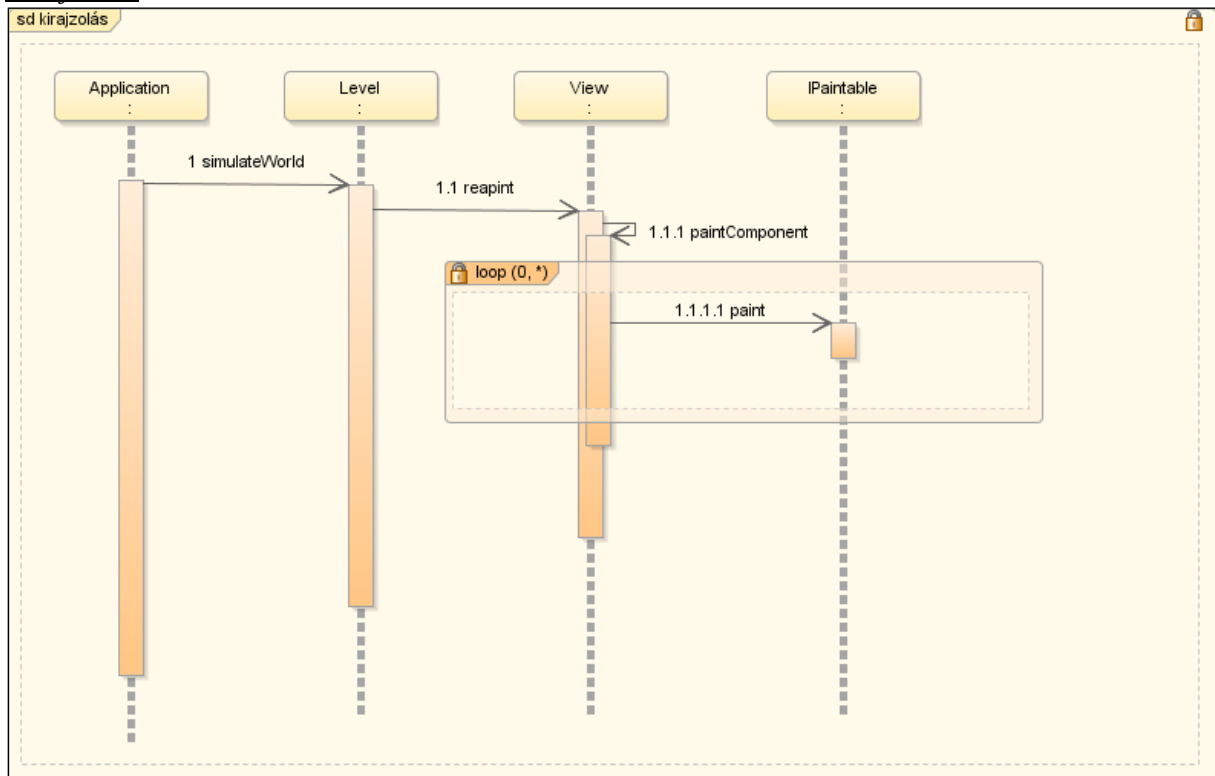
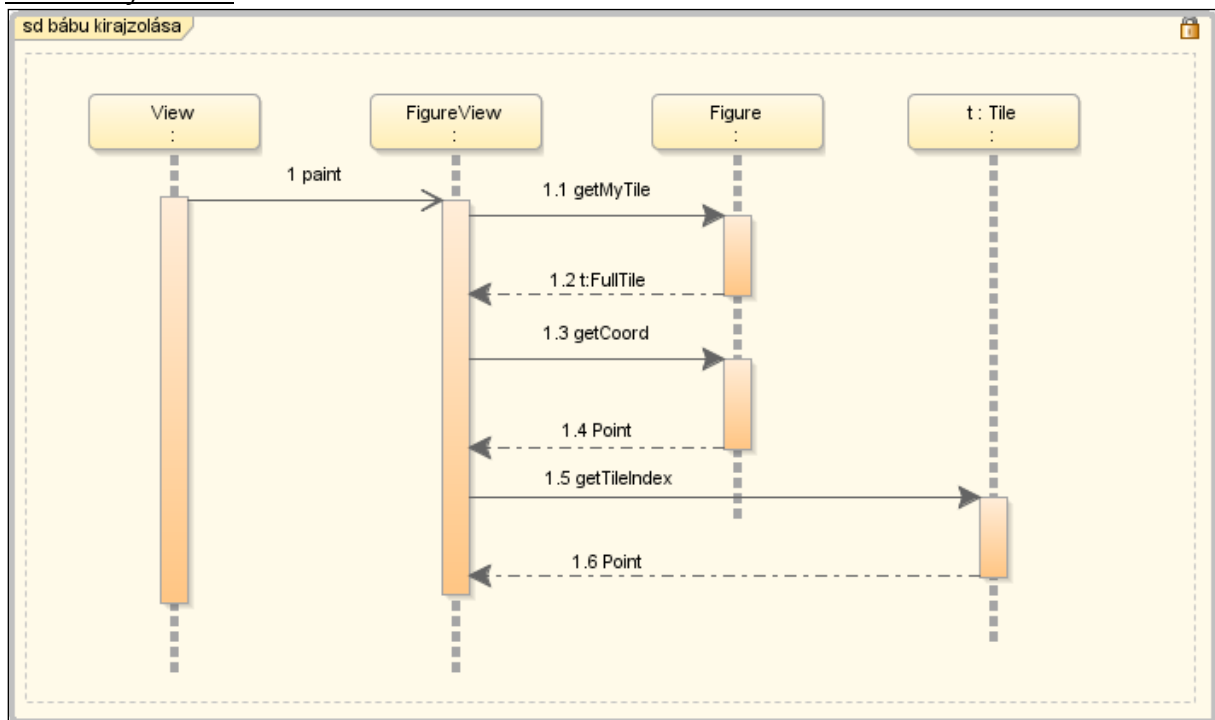
- **Felelősség**
Program indításakor jön létre. Ez az osztály tartalmazza a main() metódust és vezérli a menüt és a
- **Ősosztályok**
JFrame
- **Interfészek**
Nincs.
- **Attribútumok**
 - - **helpPanel: JPanel**: Súgó elemeit tartalmazó konténer
 - - **panel: JPanel**: Menü elemeit tartalmazó konténer
 - - **view: View**: Játékot megjelenítő view
- **Metódusok**
 - + **disableGameView():void**: Letiltja az éppen futó szint láthatóságát.
 - + **disableMenuView():void**: Letiltja a menü láthatóságát.
 - + **enableMenuView():void**: Engedélyezi a menü láthatóságát.
 - + **setupMenuView():void**: Felépíti a menüt.

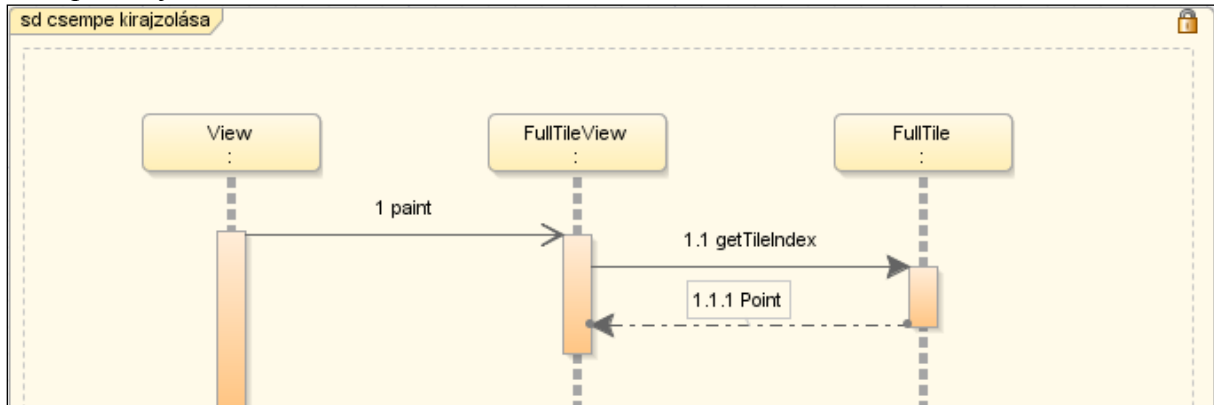
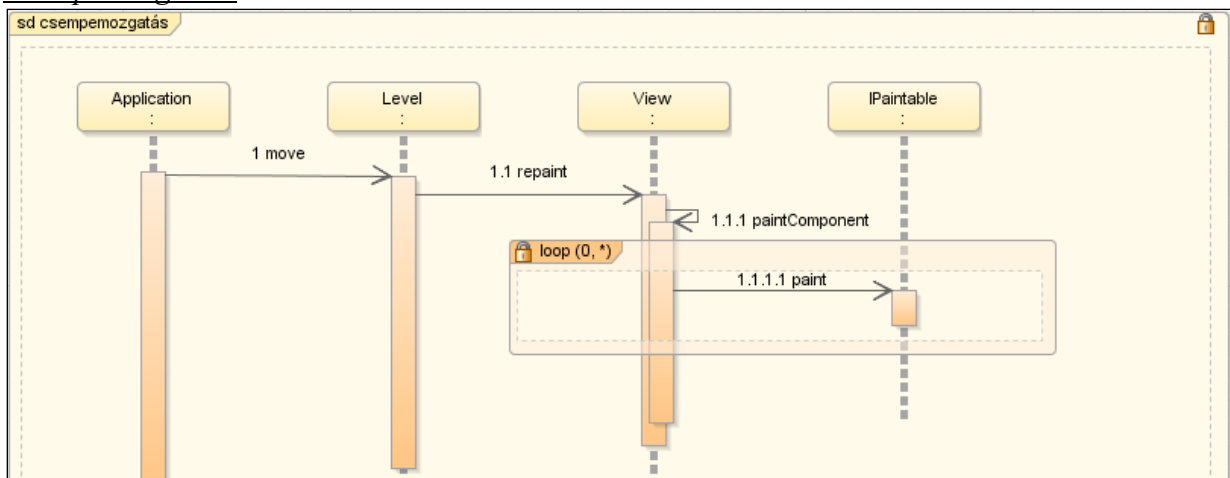
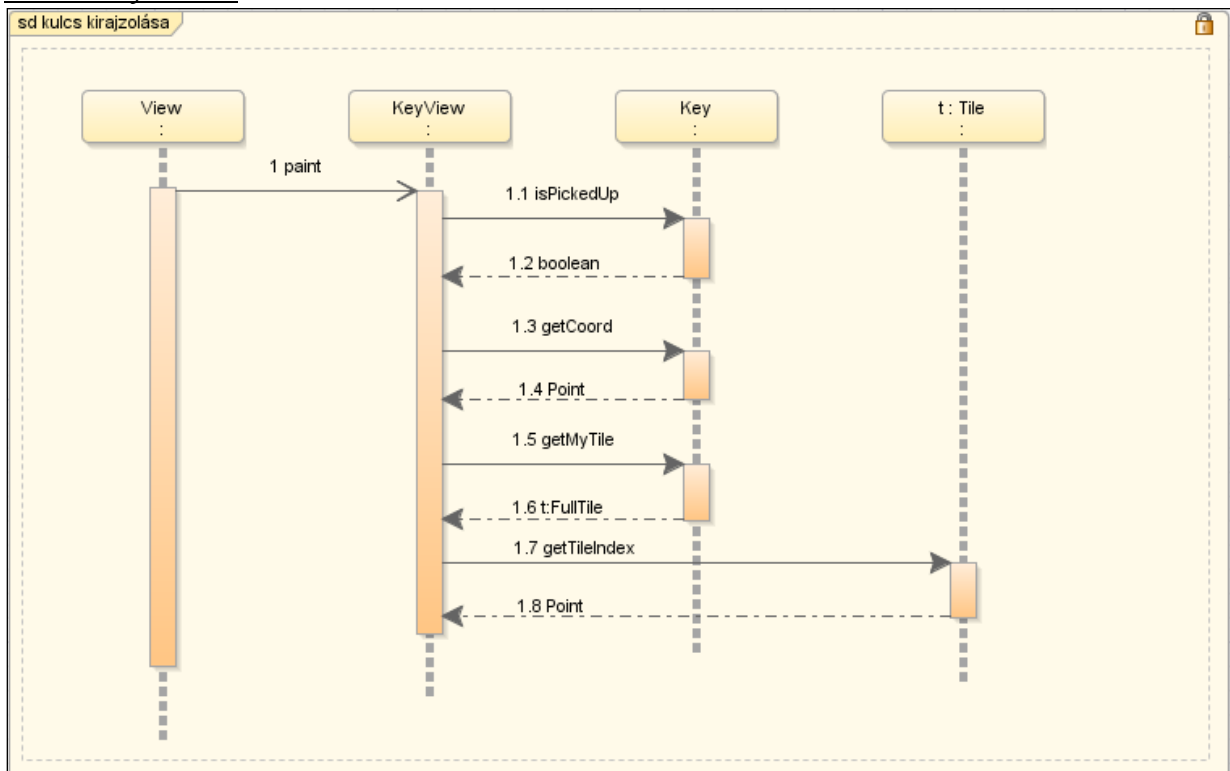
11.3 Kapcsolat az alkalmazói rendszerrel

A modell és a view közötti kapcsolatot szekvencia diagramokkal is illusztráltjuk. Látható, hogyan épül fel a kirajzolás, és hogyan kezeli le a view a modell változásait. A diagramok konzisztensek a korábbi tervekkel, az objektum katalógussal és az osztálydiagrammal.

Szekvencia diagramok:

Pálya kirajzolása játékindításkor:

KirajzolásBábu kirajzolása:

Csempe kirajzolása:Csempemozgatás:Kulcs kirajzolása:

13. Grafikus változat beadása

13.1 Fordítási és futtatási útmutató

13.1.1 Fájllista

Fájl neve	Méret [KB]	Keletkezés ideje	Tartalom (mi kerül megvalósításra)
compile.bat	1	2012.04.17.	Lefordítja a forráskódot.
run.bat	1	2012.04.17.	Futtatja a programot.
Setup.bat	1	2012.04.17.	Beállítja, a compilert és a futtatókörnyezetet.
src\Control\Controller.java	18	2012.05.07.	Forráskód.
src\Control\InputCommands.java	2	2012.04.12.	Forráskód.
src\Control\IdleThread.java	1	2012.05.07.	Forráskód.
src\Control\MainThread.java	2	2012.05.07.	Forráskód.
src\Model\Application.java	24	2012.05.07.	Forráskód.
src\Model\Direction.java	6	2012.05.07..	Forráskód.
src\Model\EmptyTile.java	2	2012.04.16.	Forráskód.
src\Model\Exit.java	2	2012.05.07.	Forráskód.
src\Model\Field.java	2	2012.05.07..	Forráskód.
src\Model\Figure.java	5	2012.05.07..	Forráskód.
src\Model\Frame.java	3	2012.04.17.	Forráskód.
src\Model\FullTile.java	24	2012.05.07.	Forráskód.
src\Model\GameObject.java	2	2012.05.07.	Forráskód.
src\Model\Key.java	2	2012.05.07.	Forráskód.
src\Model\Level.java	8	2012.04.17.	Forráskód.
src\Model\LevelInputCommands.java	1	2012.04.12.	Forráskód.
src\Model\Tile.java	2	2012.04.176	Forráskód.
src\Model\Type.java	1	2012.05.03..	Forráskód.
szint1.txt	1	2012.05.07.	A szint forrása.
szint2.txt	2	2012.05.07.	A szint forrása.
szint1_output.txt	2	2012.05.07.	A modell elvárt kimenete a szint1-re.
src\View\ExitView.java	2	2012.05.07.	Forráskód.
src\View\FigureView.java	4	2012.05.07.	Forráskód.
src\View\FullTileView.java	3	2012.05.07.	Forráskód.
src\View\KeyView.java	3	2012.05.07.	Forráskód.
src\View\IPaintable.java	1	2012.05.07.	Forráskód.
src\View\View.java	3	2012.05.07.	Forráskód.
exit.png	2	2012.05.03.	A grafikus felület forrása.
figureA_go_left.png	5	2012.05.03.	A grafikus felület forrása.
figureA_go_right.png	5	2012.05.03.	A grafikus felület forrása.
figureA_stop.png	5	2012.05.03.	A grafikus felület forrása.
figureB_go_left.png	5	2012.05.03.	A grafikus felület forrása.
figureB_go_right.png	5	2012.05.03.	A grafikus felület forrása.
figureB_stop.png	5	2012.05.03.	A grafikus felület forrása.
key.png	5	2012.05.03.	A grafikus felület forrása.
key_pickedup.png	5	2012.05.03.	A grafikus felület forrása.
gameState	1	2012.05.07.	Az aktuális pályát tárolja.

13.1.2 Fordítás és telepítés

Az egyszerűsítés, és a hibák elkerülésének érdekében a fordítást a *compile.bat* nevű batch fájl végzi. A java útvonal beállítása után (pl. set PATH=„c:\Program Files\Java\jdk1.6.0_18\bin”) célszerű végrehajtani. A fordítás eredménye a */bin* könyvtárban lesz látható: elkészülnek a *.class fájlok.

Az előállításhoz szükséges feltétel a legalább 1.6-os verziójú Java JDK kit megléte.

13.1.3 Futtatás

A futtatás egy batch fájl elindításával lehetséges (run.bat), természetesen csak az után, hogy az alkalmazás hiba nélkül lefordításra került.

Először egy menü jelenik meg a képernyőn négy gomb található. Az első az Új játék, a második a mentett Játék folytatása a harmadik a Súly, a negyedik pedig a Kilépés.

A játékban két mód van, az egyikben a csempeket tudjuk mozgatni, a másikban pedig a bábukat, ezen módok között a *Space* illetve a *Control* billentyűvel lehet váltani. Bábu módban *Le*, *Fel*, *Jobb* és *Bal* nyilakkal lehet a kék bábút irányítani a pirosat pedig az *S* (le), *W* (fel), *D* (jobb), *A* (bal) nyilakkal lehet. Csempe módban ugyan ezekkel a billentyűkkel lehet mozgatni a csempeket az üres csempehelyre.

A billentyűparancsokat és a játék menetét a Súlyban el lehet olvasni.

13.2 Értékelés

Tag neve	Munka százalékban
Berki Ramóna	15 %
Gerlei Balázs	20 %
Halász Antal István	20 %
Kiss Beáta	20 %
Szabó Csaba	25 %

14. Összefoglalás

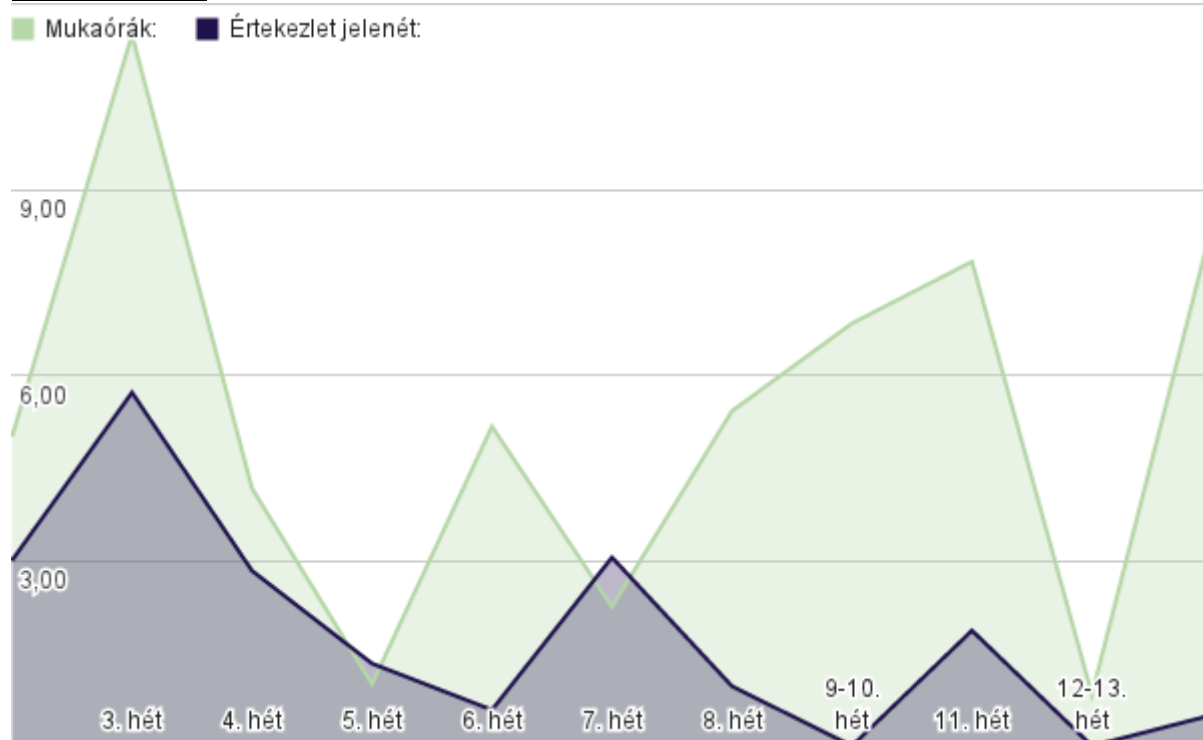
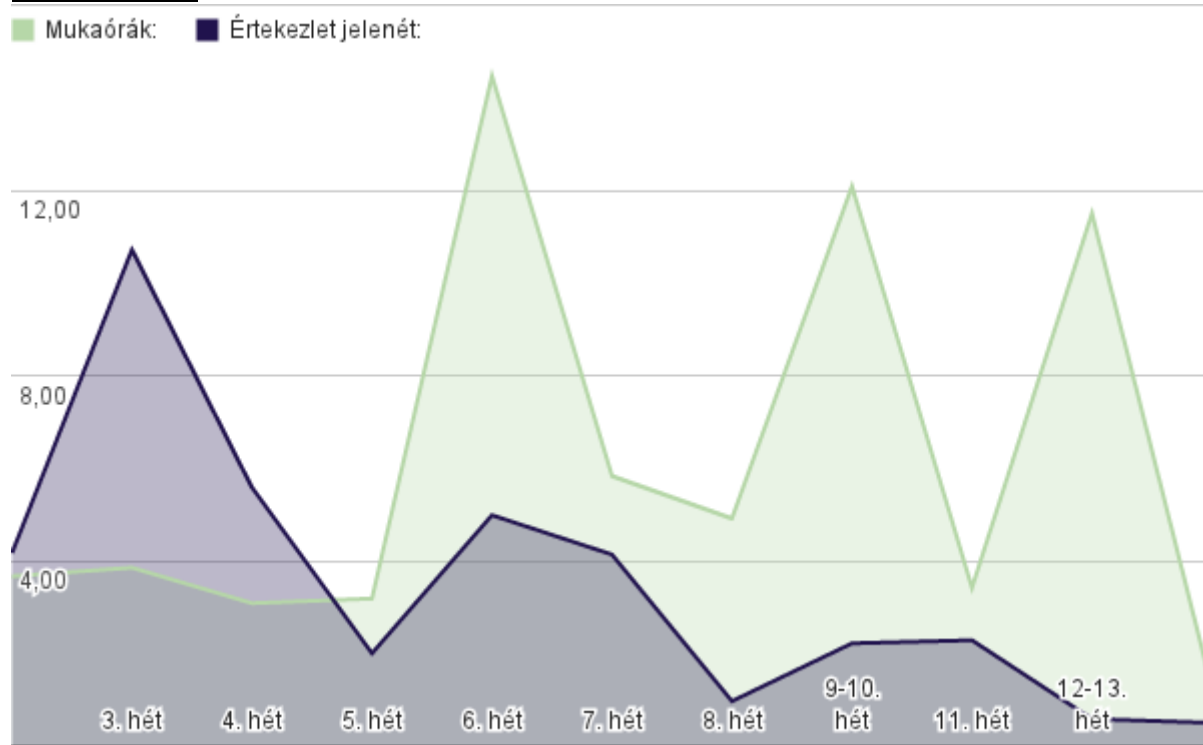
14.1. A projektre fordított összes munkaidő

A projektre fordított össze munka (percben):

Név	Munkaórák		Összesen
	<i>Értekezletek</i>	<i>Egyéni munka</i>	
Berki Ramóna	1191	3505	4696
Gerlei Balázs	2288	4035	6323
Halász Antal István	1790	5615	7585
Kiss Beáta	2059	4406	6465
Szabó Csaba	2431	6787	9218
Összesen	2650	24348	34287

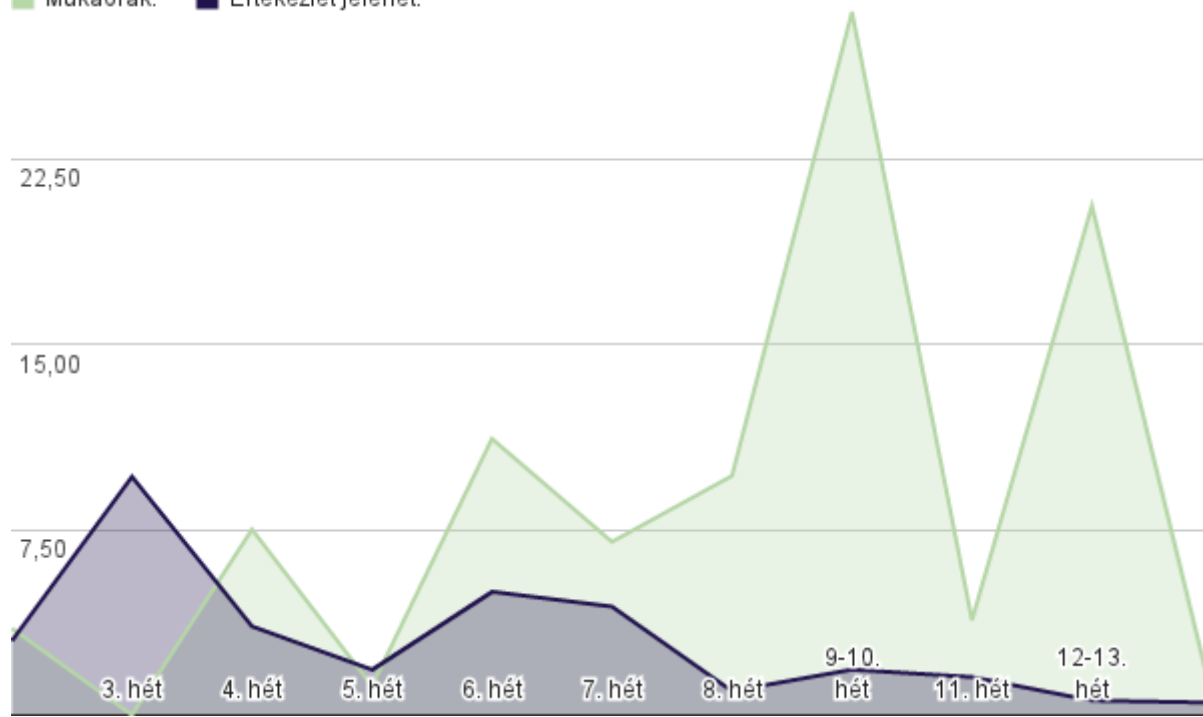
A projektre fordított össze munka (órában):

Név	Munkaórák		Összesen
	<i>Értekezletek</i>	<i>Egyéni munka</i>	
Berki Ramóna	20	58	78
Gerlei Balázs	38	67	105
Halász Antal István	33	94	126
Kiss Beáta	34	73	108
Szabó Csaba	41	113	153
Összesen	44	406	571

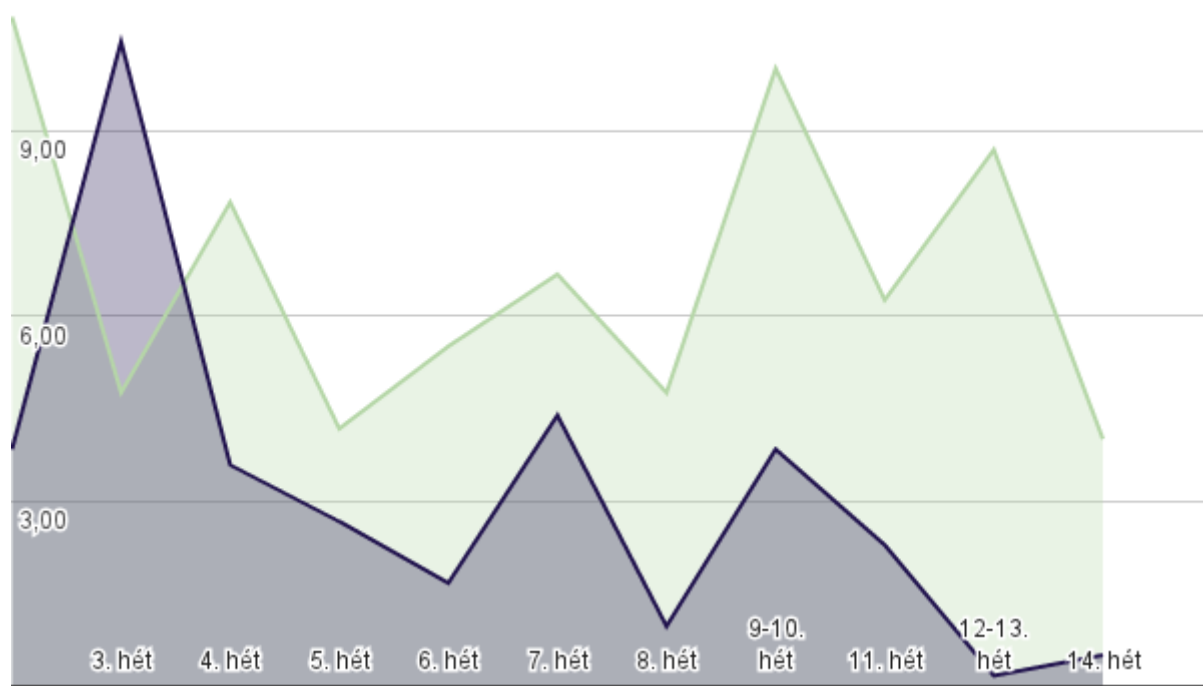
Munkaóra diagramok:Berki Ramóna:Gerlei Balázs:

Halász Antal István:

Mukaórák: Értekezlet jelenét:

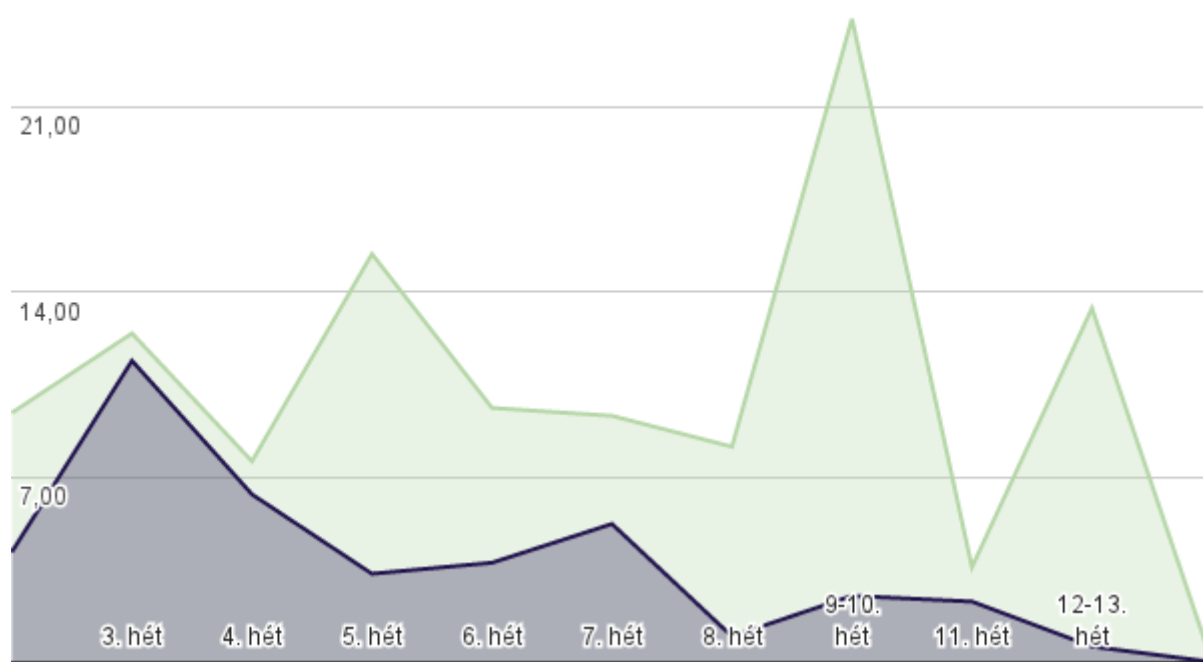
Kiss Beáta

Mukaórák: Értekezlet jelenét:

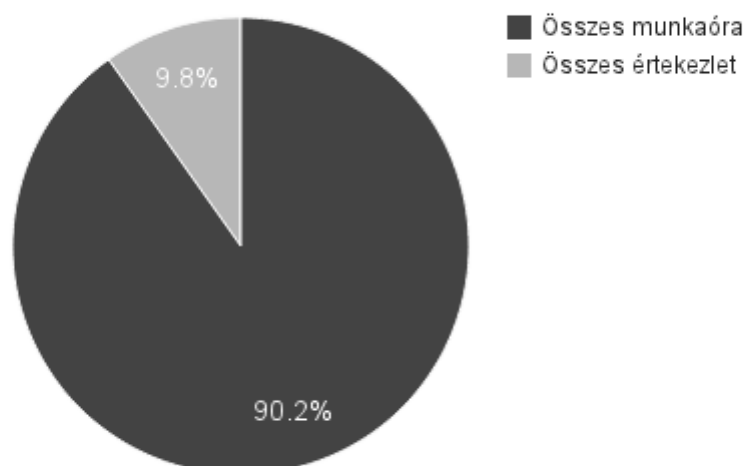


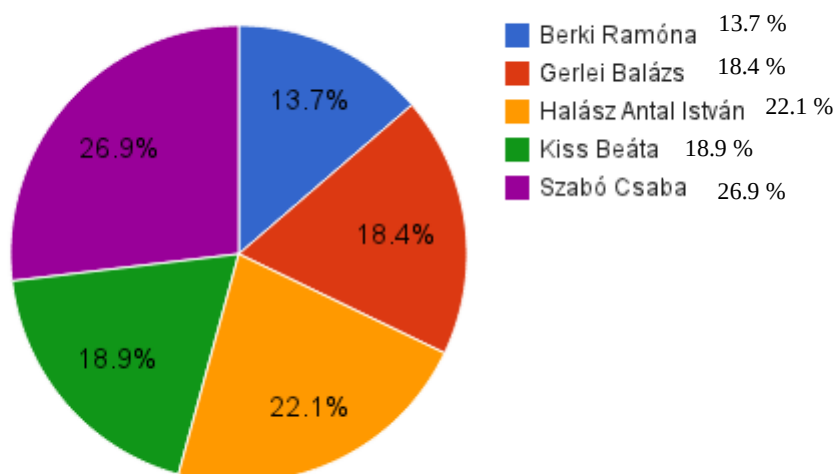
Szabó Csaba:

Munkaóra: Értekezlet jelenét:

*Értekezletek és egyéni munkaórák megoszlása:*

Munkaóra/értekezlet



Összesített munkaóra diagram:**Munkaórák értekezletekkel együtt összesen****Végső értékelés:**

Berki Ramóna	15%
Gerlei Balázs	20%
Halász Antal István	20%
Kiss Beáta	20%
Szabó Csaba	25%

14.2 A projekt során beadott forrásfájlok és azok sorainak a száma

14.2.1. Skeleton

Fájl neve	Méret	Keletkezés ideje	Tartalom	Sorok száma
Controller.java	4 KB	2012.03.19.	A Controller osztály függvényei és változói.	148
Application.java	3 KB	2012.03.19.	Az Application osztály függvényei és változói.	112
Direction.java	1 KB	2012.03.19.	A Direction osztály függvényei és változói	61
EmptyTile.java	2 KB	2012.03.19.	Az üres csempe osztály függvényei és változói.	59
Exit.java	3 KB	2012.03.19.	A kijárat osztály függvényei és változói.	115
Field.java	3 KB	2012.03.19.	A mező osztály függvényei és változói,	99
Figure.java	7 KB	2012.03.19.	A bábu osztály függvényei és változói.	288
Frame.java	4 KB	2012.03.19.	A keret osztály függvényei és változói.	144
FullTile.java	12KB	2012.03.19.	A teli csempe osztály függvényei és változói.	445
GameObject.java	2 KB	2012.03.19.	A alakzat ösosztály függvényei és változói.	91
Key.java	2 KB	2012.03.19.	A kulcs osztály függvényei és változói.	73
Level.java	5 KB	2012.03.19.	A szint osztály függvényei és változói.	198
Tile.java	2 KB	2012.03.19.	A csempe ösosztály függvényei és változói.	98
Type.java	1 KB	2012.03.19.	A típus enum függvényei és változói.	14
Continuity.jar	19 KB	2012.03.19.	A projekt futtatható formában.	
Összesen:				1945

14.2.2. Prototípus

Fájl neve	Méret [byte]	Keletkezés ideje	Tartalom (mi kerül megvalósításra)	Sorok száma
compile.bat	214	2012.04.17.	Lefordítja a forráskódot.	5
run.bat	181	2012.04.17.	Futtatja a prototípust.	5

Setup.bat	84	2012.04.17.	Beállítja, milyen java compilert és futtatókörnyezetet használunk.	2
src\Control\Controller.java	9 325	2012.04.17.	Forráskód.	239
src\Control\InputCommands.java	174	2012.04.17.	Forráskód.	13
src\Model\Application.java	22 395	2012.04.17.	Forráskód.	723
src\Model\Direction.java	4 538	2012.04.17.	Forráskód.	208
src\Model\EmptyTile.java	1 696	2012.04.17.	Forráskód.	65
src\Model\Exit.java	1 839	2012.04.17.	Forráskód.	74
src\Model\Field.java	1 294	2012.04.17.	Forráskód.	59
src\Model\Figure.java	3 625	2012.04.17.	Forráskód.	212
src\Model\Frame.java	1 928	2012.04.17.	Forráskód.	94
src\Model\FullTile.java	16 041	2012.04.17.	Forráskód.	691
src\Model\GameObject.java	1 096	2012.04.17.	Forráskód.	57
src\Model\Key.java	1 214	2012.04.17.	Forráskód.	63
src\Model\Level.java	7 382	2012.04.17.	Forráskód.	347
src\Model\LevelInputCommands.java	204	2012.04.17.	Forráskód.	16
src\Model\Tile.java	1 577	2012.04.17.	Forráskód.	93
src\Model\Type.java	367	2012.04.17.	Forráskód.	21
szint1.txt	1022	2012.04.17.	Előre megírt szint a tesztesetekhez.	56
szint2.txt	1 988	2012.04.17.	Előre megírt szint a tesztesetekhez.	109
teszteset1.txt	89	2012.04.17.	Bemenet.	4
teszteset1_output.txt	1 921	2012.04.17.	A teszt eredménye.	57
teszteset2.txt	616	2012.04.17.	Bemenet.	52
teszteset2_output.txt	5 334	2012.04.17.	A teszt eredménye.	107
teszteset3.txt	160	2012.04.17.	Bemenet.	10
teszteset3_output.txt	746	2012.04.17.	A teszt eredménye.	26
teszteset4.txt	185	2012.04.17.	Bemenet.	12
teszteset4_output.txt	557	2012.04.17.	A teszt eredménye.	15
teszteset5.txt	208	2012.04.17.	Bemenet.	14
teszteset5_output.txt	773	2012.04.17.	A teszt eredménye.	23
teszteset6.txt	141	2012.04.17.	Bemenet.	10
teszteset6_output.txt	528	2012.04.17.	A teszt eredménye.	14
teszteset7.txt	185	2012.04.17.	Bemenet.	14
teszteset7_output.txt	557	2012.04.17.	A teszt eredménye.	128
teszteset8.txt	185	2012.04.17.	Bemenet.	14
teszteset8_output.txt	557	2012.04.17.	A teszt eredménye.	19
teszteset9.txt	335	2012.04.17.	Bemenet.	32
teszteset9_output.txt	1 472	2012.04.17.	A teszt eredménye.	52
teszteset10.txt	380	2012.04.17.	Bemenet.	27
teszteset10_output.txt	2 451	2012.04.17.	A teszt eredménye.	161
teszteset1_elvart.txt	1 921	2012.04.17.	A teszteset kimenete.	57
teszteset2_elvart.txt	5 334	2012.04.17.	A teszteset kimenete.	107
teszteset3_elvart.txt	746	2012.04.17.	A teszteset kimenete.	26

teszteset4_elvart.txt	557	2012.04.17	A teszteset kimenete.	15
teszteset5_elvart.txt	773	2012.04.17	A teszteset kimenete.	23
teszteset6_elvart.txt	528	2012.04.17	A teszteset kimenete.	14
teszteset7_elvart.txt	557	2012.04.17	A teszteset kimenete.	128
teszteset8_elvart.txt	557	2012.04.17	A teszteset kimenete.	19
teszteset9_elvart.txt	1472	2012.04.17	A teszteset kimenete.	52
teszteset10_elvart.txt	2 334	2012.04.17	A teszteset kimenete.	161
Összesen:				4545

14.2.3. Garfikus változat

Fájl neve	Méret [KB]	Keletkezés ideje	Tartalom (mi kerül megvalósításra)	Sorok száma
compile.bat	1	2012.04.17.	Lefordítja a forráskódot.	5
run.bat	1	2012.04.17.	Futtatja a programot.	5
Setup.bat	1	2012.04.17.	Beállítja, a compilert és a futtatókörnyezetet.	2
src\Control\Controller.java	18	2012.05.07.	Forráskód.	587
src\Control\InputCommands.java	2	2012.04.12.	Forráskód.	13
src\Control\IdleThread.java	1	2012.05.07.	Forráskód.	62
src\Control\MainThread.java	2	2012.05.07.	Forráskód.	60
src\Model\Application.java	24	2012.05.07.	Forráskód.	793
src\Model\Direction.java	6	2012.05.07..	Forráskód.	269
src\Model\EmptyTile.java	2	2012.04.16.	Forráskód.	65
src\Model\Exit.java	2	2012.05.07.	Forráskód.	79
src\Model\Field.java	2	2012.05.07..	Forráskód.	78
src\Model\Figure.java	5	2012.05.07..	Forráskód.	226
src\Model\Frame.java	3	2012.04.17.	Forráskód.	94
src\Model\FullTile.java	24	2012.05.07.	Forráskód.	824
src\Model\GameObject.java	2	2012.05.07.	Forráskód.	93
src\Model\Key.java	2	2012.05.07.	Forráskód.	69
src\Model\Level.java	8	2012.04.17.	Forráskód.	347
src\Model\LevelInputCommands.java	1	2012.04.12.	Forráskód.	16
src\Model\Tile.java	2	2012.04.176	Forráskód.	93
src\Model\Type.java	1	2012.05.03..	Forráskód.	22
szint1.txt	1	2012.05.07.	A szint forrása.	56
szint2.txt	2	2012.05.07.	A szint forrása.	109
szint1_output.txt	2	2012.05.07.	A modell elvárt kimenete a szint1-re.	56
src\View\ExitView.java	2	2012.05.07.	Forráskód.	77
src\View\FigureView.java	4	2012.05.07.	Forráskód.	117

src\View\FullTileView.java	3	2012.05.07.	Forráskód.	85
src\View\KeyView.java	3	2012.05.07.	Forráskód.	86
src\View\IPaintable.java	1	2012.05.07.	Forráskód.	19
src\View\View.java	3	2012.05.07.	Forráskód.	124
actualGame			Az aktuális pályát tárolja. (Nem használjuk.)	31
gameState	1	2012.05.07.	Az aktuális pályát tárolja.	2
Összesen:				4564

Az összes kódsor: 11 054

14.3 A projekt összegzése

14.3.1. Mit tanultunk?

A félév során nagyon sok tapasztalatot szereztünk a csapatmunka és a másokkal való együtt dolgozás területén. Az élményeink többsége pozitív, persze akadtak néha nehéz pillanatok, amikor nem tudtuk, hogyan is tovább. Vészesebb helyzetekben tudtunk egymásra támaszkodni és egymást biztatni, hogy közel van már a cél.

Néhány dolog, amit szeretnénk kiemelni a projekt alatt tanultak közül:

- Nehéz volt olyan közös időpontot találni értekezletekhez, ami mindenkinek megfelelt. (Mindenkinek más volt az órarendje, időbeosztása.)
- A vége felé elfogyott a lelkesedés, nem osztottuk be jól az energiáinkat.
- Élveztük a projektet, szeretnénk volna minél jobbra megcsinálni, de egyetem mellett nehéz volt plusz időt rászánni.
- Nagyon fontos a lényegre törő, érthető kommunikáció.
- Fontos, hogy mindenki mindenről tudjon, és jól ledokumentáljuk a döntéseket; ne legyen többszálú kommunikáció.
- Lényeges, hogy le tudjuk úgy kommunikálni a többiek felé az ötleteinket és a javaslatainkat, hogy az ne legyen sértő vagy bántó.
- Fontos, hogy tervezzük meg a heti munkamenetet, és jól osszuk be az időnket.
- Inkább becsüljük túl a projektet, mint alul. Főleg implementálásnál, ha egy feladat könnyűnek és rögtön működőnek látszik, ott is léphetnek fel váratlan hibák.
- Fontos, hogy akkor is próbáljunk meg időt szakítani a csapattagokkal való gyűlésekre mikor kevés időnk van, hisz ez egy projekt és fontosak a rendszerek gyűlések, hogy valóban csapatnak érezzük magunkat.

14.3.2. Mi volt a legnehezebb?

A csapat nagy része a projektmunka első felét érezte a legnehezebbnek, körülbelül azt az időszakot, amíg eljutottunk a skeleton megvalósításáig:

- Nehezebb volt a tervezés, hisz ilyen típusú munkában még egyikünknek sem volt tapasztalata.
- A csapat egy részének nehezen ment az ilyen magas absztrakciós szinten való gondolkodás.
- Sokszor nehéz volt összeszervezni az embereket. Az implementációs időszak hosszú hétvégéi csak nehezítették a dolgot, hisz akkor mindenki hazament, pihenni szeretett

volna és a családdal lenni. (Egyikünk sem budapesti.) Ilyenkor a skypos értekezleteken sem tudott mindenki részt venni technikai okok miatt.

- Sokszor nem sikerült jól ütemezni a beadásokat, így az utolsó pillanatban átvirrasztott éjszakákat vagy a más tárgyakra való készületlenséget eredményezett és még így is csúszva sikerült beadni a feladatot. Ez emberileg megterhelő, idegörlő volt.

14.3.3. Mi volt a legkönnyebb?

Két olyan dolog volt, amit meglepően könnyűnek éreztünk:

- Az értekezleteken való együttes munkát, a közös tervezést és gondolkozást.
- A tesztelést, ennek is azon részét élveztük különösen, amikor más csapatok protóját vagy skeletonját kellett tesztelni. Érdekes volt belelátni mások kódjába, megnézni ők hogyan oldották meg ugyan azt a feladatot.

14.3.4. Pontszám és feladatok összhangja

Úgy érezzük, hogy a megkapható pontszámok arányban álltak a feladatok nehézségével, viszont a tárgy kreditértéke nem áll arányban a ráfordított idővel és energiával, illetve a feladat komplexségével.

14.3.5. Javasolt változások

Véleményünk a tárgy kreditértéke nincs arányban a befektetett munkával. Mi például körülbelül 120 óra/fő –t dolgoztunk vele a félév alatt 2 kreditért, míg a tanterv szerint egy kredit 30 munkaórával teljesíthető.

14.3.6. Projekt ajánlatok

UML diagramkészítő programot képzeltünk el, amely mindenképpen tudjon C, C++ és Java nyelven, osztálydiagramot, szekvencia diagramot rajzolni. Lehesse vele projektet létrehozni, ahol egy áttekintő nézetben meg lehessen nézni az adott projekthez tartozó diagramok listáját. A cél, hogy az eszközt hatékonyan lehessen használni egy, a tárgy keretein belül is zajló szoftverfejlesztési feladathoz.

Egyéb ajánlatok:

- Egy vállalatirányítási rendszer (pl. SAP vagy Navision) modellezése.
- Neptun szerű adatbázis-kezelő rendszer.

Napló

2. hét:

Kezdet	Időtartam (perc)	Résztevők	Leírás
2011.02.15 18:30	120	Berki Gerlei Halász Kiss Szabó	Értekezlet. Döntés: - Szabó lett a csapatkoordinátor - mindenki elkészíti a szótárt és a feladatleírást 02.16 24:00-ig - Kiss elvégzi az adminisztratív teendőket - következő értekezlet: 02.18. 10:00
2011.02.18. 10:15	80	Berki Gerlei Halász Kiss Szabó	Értekezlet. Döntés: - Szabó letisztázza a 2.1 és 2.2 pontokat, Kiss letisztázza a 2.3 és 2.4 pontokat és a use case-eket 02.19. 12:00-ig - mindenki megismeri a programokat, amiket elvállalt 02.22. 18:00-ig - Gerlei beadja a feladatot 02.20-án
2011. 02. 19. 19:34	30	Gerlei Kiss Szabó	Értekezlet. Döntések: - egységes formázást használunk - Gerlei, Kiss, Szabó átnézik egymás dokumentumait - Gerlei nyomtatja, Szabó leadja a dokumentumot
2011.02.19. 21:15	20	Gerlei Szabó	Értekezlet. Döntések: - Dokumentumok véglegesítésének határideje 23:30 - Órák átszámítása percre a munkaóra táblázatban
2011.02.19. 23:40	20	Gerlei Kiss Szabó	Értekezlet. Döntések: - Csapattagok értékelésének megjelenítése kördiagram formában (munkaórák alapján) - Dokumentumok nyomtathatóak
2012.02.16. 23:00:00	45	Berki	Megírtam a Feladatleírást (2.3).
2012.02.16. 23:55:00	15	Berki	Megírtam a Szótárt (2.4).
2012.02.18. 20:00:00	60	Berki	Letöltöttem és kipróbáltam az OpenAmeost.
2012.02.18. 21:00:00	180	Berki	Letöltöttem és kipróbáltam az Oracle JDevelopert.
2012.02.16. 14:00:00	60	Gerlei	Megírtam a Feladatleírást (2.3).
2012.02.16. 21:30:00	30	Gerlei	Megírtam a Szótárt (2.4)
2012.02.19.	60	Gerlei	Átnéztem és korrektúráztam a Kiss által megírt

16:00:00			végleges szótárt (2.4)
2012.02.19. 17:15:00	30	Gerlei	Átnéztem és korrektúráztam a Kiss által megírt végleges feladatleírást (2.4)
2012.02.19. 20:45:00	40	Gerlei	Átnéztem és korrektúráztam a Szabó által megírt követelmény definíciót (2.1) és projekt tervet (2.2), valamint a Kiss által megírt essential usecase-eket (2.5)
2012.02.16. 22:30:00	120	Halász	Megírtam a Feladatleírást (2.3).
2012.02.17. '00:00:00	30	Halász	Megírtam a Szótárt (2.4)
2012.02.19. 19:15:00	60	Halász	SDK és verziókezelőprogram-lehetőségek vizsgálata
2012.02.15. 21:45:00	30	Kiss	Az ülésjegyzőkönyvet megszerkesztettem, feltöltöttem.
2012.02.17. 18:10:00	140	Kiss	A jegyzőkönyvet a többiek tanácsai alapján átszerkesztettem, use-case diagramot és az use case leírásokat digitalizáltam.
2012.02.18. 15:00:00	120	Kiss	A 2.3. és 2.4. részeket elkészítettem.
2012.02.18. 18:15:00	150	Kiss	A 2.3. és 2.4. részeket az ülésen megbeszéltek alapján összeszerkesztettem az 5 különbözőből.
2012.02.18. 20:50:00	40	Kiss	A 2.5. és a 2.6. részt a többiek javaslataival kiegészítettem.
2012.02.19. 11:30:00	60	Kiss	A 2.3-at és 2.4-et helyesírási, nyelvhelyességi, a mások által talált pontatlanságokat kijavítottam, a naplót elkezdtem szerkeszteni.
2012.02.19. 17:00:00	30	Kiss	2.1 és 2.2 dokumentumokat ellenőriztem, dokumentum sablon készítettem.
2012.02.19. 22:30:00	80	Kiss	A beadandó dokumentumokat végeges formára szerkesztettem, a naplót elkészítettem.
2012.02.16. 14:00:00	150	Szabó	Megírtam a Feladatleírást (2.3).
2012.02.16. 20:30:00	60	Szabó	Megírtam a Szótárt (2.4)
2012.02.16. 23:00:00	30	Szabó	Megírtam az egységes névkonvenciót.
2012.02.17. 0:20:00	60	Szabó	Átolvastam a többiek által elkészített feladatleírásokat és egy összesítést készítettem belőle mely felhasználható a megbeszélésen + megbeszélés egyéb témáinak megírása.
2012.02.18. 16:20:00	120	Szabó	Dropboxra átköltöttem a dokumentumokat a Google Docsról + befejeztem a 2.1 pont (Követelmény def.) leírását.
2012.02.18. 18:30:00	60	Szabó	2.2 pont (projekt terv) leírásának befejezése.
2012.02.19. 20:00:00	30	Szabó	Átnéztem és javítottam a Kiss által megírt végleges feladatleírást (2.3, 2.4)

2012.02.19. 20:30:00	15	Szabó	Átnéztem és javítottam a Kiss által megírt végleges use-case leírást és diagramot (2.5, 2.6)
2012.02.19. 22:00:00	10	Szabó	2.1 2.2 2.3 2.4 véglegesítettem, Dropboxba feltöltöttem.
2012.02.19. 22:10:00	30	Szabó	2.5 2.6 véglegesítésítettem, Dropboxba feltöltöttem.

3. hét:

Kezdet	Időtartam	Résztvevők	Leírás
2012.02.24. 18:15:00	120	Berki Gerlei Halász Kiss Szabó	Döntések: - Mindenki magyar szavakkal leírja, hogy néz ki, amikor a program feláll. - Holnap 10-re feltölteni és ekkor gyűlés SCH-ban.
2012.02.25. 10:00:00	110	Gerlei Halász Kiss Szabó	Döntések: - Az osztálydiagramokat véglegesek. - Ebédszünet, utána a szekvenciadiagramokat kidolgozzuk közösen.
2012.02.24. 16:00:00	120	Gerlei Halász Kiss Szabó	Döntések: - Szabó osztálydiagramok és a szekvencia diagramokat letisztázza ma 21:00-ig - Gerlei és Halász a 3.1 és a 3.2-t megcsinálja 02.25 éjfélig. (Nem volt idejük rá, a másnapi értekezletre halsztottuk.)
2012.02.24. 08:30:00	258	Berki Gerlei Halász Kiss Szabó	Az értekezleten nem mindenki volt végig jelen. Döntések: - Gerlei 22:00-ig átnézni, hogy a 3.1 és a 3.2 konzisztens-e a szótárral. - Szabó 16:00-ig átnézni, hogy a 3.1 és a 3.2 konzisztens-e a konzultáción elhangzottakkal. - Berki feltölti a kész osztály diagramokat 16:00-ig. - Berki feltölti a kész szekvencia diagramokat 17:00-ig. - Kiss véglegesíti a dokumentációt 02.27. 6:00-ig.
2012.02.25. 22:15:00	330	Berki	Class diagram rajzolás JDeveloper-ben

2012.02.26. 10:20:00	160	Berki	Class diagram befejezése, javítása és szekvencia diagramok rajzolása.
2012.02.26. 16:30:00	200	Berki	Szekvencia diagramok folytatása, befejezése.
2012.02.22. 16:45:00	75	Gerlei	Javadoc konenciók átnézése.
2012.02.23. 22:03:00	120	Gerlei	Az elkészítendő program Modell-View-Control tervezési minta alapján történő megvalósíthatóságának elemzése.
2012.05.25. 18:30:00	90	Gerlei	Végleges dokumentum szerkesztése.
2012.02.24. 21:00:00	45	Kiss	Osztálydiagramok átgondolása
2012.02.25. 23:50:00	90	Kiss	A meglévő osztálydiagramban hibák keresése, azokra megoldási terv készítése.
2012.05.25. 18:30:00	90	Kiss	Dokumentáció átnézése, szerkesztése
2012.05.25. 32:20:00	60	Kiss	A dokumentáció véglegesítése.
2012.02.22. 8:15:00	45	Szabó	Konzultáció.
2012.02.22. 13:40:00	60	Szabó	Dropbox rendszerezés, 3. heti template, 3. heti mankó1.
2012.02.24. 22:30:00	90	Szabó	Osztálydiagram, szekvenciadiagramok tervezése.
2012.02.25. 7:20:00	75	Szabó	Osztálydiagram, szekvenciadiagramok tervezése.
2012.02.25. 18:18:00	60	Szabó	Osztálydiagram megrajzolása.
2012.02.25. 20:45:00	110	Szabó	Szekvencia diagramok rajzolása.
2012.02.25. 23:00:00	180	Szabó	Osztálydiagram javítása és szekvenciadiagram rajzolása.

4. hét:

Kezdet	Időtartam	Résztvevők	Leírás
2012.02.29. 12:35:00	40	Halász Kiss Szabó	Döntések: - A megbeszéltek alapján Szabó újrarajzolja az osztálydiagramot 02.29. 18:00-ig. - Kiss és Halász megrajzolja a szekvenciadiagramokat 02.29. 18:00-ig.
2012.02.29. 18:25:00	170	Gerlei Halász Kiss Szabó	Döntések: - 03.01. 09:00 Szabó, Halász és Gerlei befejezik a szekvenciadiagramokat. - 03.01. 06:00-ig mindeki

			átgondolja a modellt.
2012.02.24. 09:15:00	165	Gerlei Szabó	Döntések: - Szabó befejezi az utolsó (vízszintesen fut és nekimegy a falnak) szekvencia diagrammot - Berki bedigitalizálja a szekvencia diagramot - Kiss és Halász megírja a 3.1-et és 3.2-t a szekvencia diagramok alapján - Szabó a kész 3.1 és 3.2 alapján megcsinálja a class diagramot - Határidőket később egyeztetünk
2012.03.03. 20:30:00	130	Berki	4.1 és 4.2 készítése..
2012.03.04. 15:30:00	120	Berki	Osztály diagram készítése
2012.02.29. 8:15:00	45	Gerlei	Konzultáció
2012.03.03. 10:30:00	105	Gerlei	Szekvenciadiagramok átnézése.
2012.03.04. 11:50:00	35	Gerlei	4.1, 4.2 átolvasása.
2012.02.29. 13:00:00	120	Halász	A modell átgondolása.
2012.03.01. 14:10:00	90	Halász	Szekvenciadiagramok átnézése.
2012.03.03. 11:10:00	45	Halász	Szekvenciadiagramok átnézése.
2012.03.03. 17:15:00	90	Halász	Szekvenciadiagramok átnézése.
2012.03.03. 20:30:00	90	Halász	4.1 és 4.2 készítése
2012.02.29. 8:15:00	45	Kiss	Konzultáció
2012.02.29. 13:00:00	100	Kiss	A modell átgondolása. Szekvenciadiagramok rajzolása.
2012.03.01. 14:10:00	130	Kiss	Szekvenciadiagramok átnézése.
2012.03.03. 12:00:00	30	Kiss	Szekvenciadiagramok átnézése.
2012.03.03. 17:15:00	60	Kiss	Szekvenciadiagramok átnézése.
2012.03.04. 18:30:00	105	Kiss	Végleges dokumentum formázása, egyéni munkanaplók összesítése.
2012.02.29. 8:15:00	45	Szabó	Konzultáció
2012.02.29.	45	Szabó	A modell átgondolása.

15:00:00			
2012.03.01. 15:45:00	15	Szabó	Szekvenciadiagram és osztálydiagram rajzolása.
2012.03.01. 18:20:00	25	Szabó	Osztálydiagram készítés google docs-ban.
2012.03.02. 18:00:00	60	Szabó	Szekvenciadiagramok átnézése.
2012.03.02. 20:10:00	100	Szabó	Szekvenciadiagramok átnézése/készítése.
2012.03.02. 22:10:00	40	Szabó	Szekvenciadiagramok készítése.
2012.03.02. 23:05:00	15	Szabó	Szekvenciadiagramok alapján osztálydiagram frissítése.
2012.03.03. 13:35:00	75	Szabó	Szekvenciadiagramok véglegesítése, dropboxba feltöltése, osztálydiagram véglegesítése ez alapján

5. hét:

Kezdet	Időtartam	Résztevők	Leírás
2012.03.07. 8:15:00	60	Berki	Konzultáció
2012.03.07. 8:15:00	60	Gerlei	Konzultáció
2012.03.09. 20:30:00	46	Gerlei	Eddig elkészült anyagok átnézése, beadandó dokumentumok finomítása
2012.03.11. 14:40:00	40	Gerlei	Szótár felvitele google docs táblázatba a későbbi bővíthetőség kedvéért
2012.03.11. 15:20:00	30	Gerlei	Beadandó dokumentum finomítása
2012.03.11. 15:50:00	15	Gerlei	Munkaórák számolását végző függvények, grafikonok működésének leellenőrzése
2012.03.08. 14:20:00	70	Halász	Use-case készítés.
2012.03.08. 14:20:00	70	Kiss	Use-case készítés.
2012.03.11. 19:50:00	120	Kiss	Use-case-k átgondolása, tesztesetek átgondolása.
2012.03.11. 21:45:00	50	Kiss	Dokumentálás.
2012.03.07. 8:15:00	60	Szabó	Konzultáció
2012.03.07. 15:35:00	30	Szabó	5. heti template, 5. heti mankó, régebbi beadott dokumentumok átnézése.
2012.03.08. 0:00:01	120	Szabó	Szekvenciadiagramok javítása.
2012.03.08. 9:15:00	90	Szabó	JDeveloper telepítés, osztálydiagram rajzolás (félkész), szekvenciadiagram ellenőrzés.
2012.03.08.	50	Szabó	Szekvenciadiagramok javítása, véglegesítése.

12:05:00			
2012.03.08. 14:20:00	70	Szabó	Use-case készítés.
2012.03.08. 21:50:00	140	Szabó	Class diagram véglegesítése. Use-case diagram digitalizálása, átnézése. Use-case tablazat.
2012.03.09. 16:50:00	135	Szabó	Use-case diagram átnézése, use-case táblázat átnézése. Architektúra átgondolása, elkészítése. Architektúra tesztesetek megírása. A szkeleton kezelői felületének terve, dialógusok.
2012.03.10. 11:45:00	50	Szabó	Dokumentum-írás folytatása, hibák keresése, javítása. Szekvenciadiagramok beillesztése. Szekvenciadiagramokról a "websequencediagram.com" felirat eltávolítása.
2012.03.10. 17:00:00	30	Szabó	Játék indítás szekvenciadiagram készítése, szekvenciadiagramos rész javítása, véglegesítése.
2012.03.11. 16:24:00	20	Szabó	Use-case diagram korrigálása.
2012.03.11. 20:00:00	130	Szabó	Dokumentum átnézése.

6. hét:

Kezdet	Időtartam (perc)	Résztevők	Leírás
2012.03.14. 18:15	25	Berki Gerlei Halász Kiss Szabó	Értekezlet. Döntés: - Halász és Gerlei GIT-et tesztel csütörtök délelőtt - Szabó JDeveloperrel kódot generál az osztálydiagramból csütörtök délelőtt
2012.03.15. 10:30	74	Gerlei Halász	Értekezlet. Döntés: - Projekt neve „Continuity” lesz. - GIT verziókövetőt nem sikerült működésre bírni, további tesztelés szükséges, Halász és Kiss ezzel fog foglalkozni a héten. - Gerlei kódol a héten, ha kódmegosztásra van szükség, Dropboxot használunk - 19:00-kor kódolási értekezlet Skype-on.
2012. 03.15. 19:10	35	Gerlei Halász Kiss Szabó	Értekezlet. Döntések: - Gerlei megcsinálja a Java.doc tutorialt, összerakja projektbe a generált vázat és holnap elkezd a kódolást. - Halász és Kiss továbbra is a GIT-tel foglalkozik.

2012.03.16. 19:00	100	Gerlei Halász Szabó	Értekezlet. Döntések: - Mostantól engedélyezve van a „.docx” fájlformátum a csapat által használt dokumentumok között. A régebbi dokumentumokat természetesen nem írjuk át, és továbbra is lehet „.doc”-t használni, de az értekezlet naplókat mindig „.docx”-ben mentjük! - A projekt (program) főverziójában és az egyes osztályok, interfészek verziószámaiban is a „major . minor . buildnumber . revisionnumber” formátumot használjuk. A főverzió jelenleg 0.0.0.1, az első, működő szkeleton lesz az 1.0. - Ha beleszerkesztünk egy osztályba, vagy interfészbe, kötelezően ki kell tölteni az „@author” tag-et, kronológiai sorrendben, de csak egyszer, az első belenyúláskor. - Gerlei még ma elkezdi a kódolást, de előtte egy újabb értekezletet tartunk 21:30-kor, hogy megbeszéljük a munka elkezdéséhez szükséges dolgokat.
2012.03.16. 21:40	25	Gerlei Halász Szabó	Értekezlet. Döntések: - Gerlei ma este lekódolja a szkeleton kezdőképernyőjét, illetve a behúzásokat kezelő részeket. - A behúzást végző tab-okat külön private tabs változóban tároljuk. - A Controllerben lesz az egyszerű kezelőfelület
2011.03.18. 12:20	30	Berki Gerlei Halász Kiss	Értekezlet. Döntések: - Berki megírja a 6.1 6.1.2 és a 6.1.3-at - Halász és Szabó 03.18. 19:00-ig megpróbálja lekódolni a hiányzó részeket, ha nem tudja befejezni Gerlei befejezi 03.18. 24:00-ig.
2012.03.16. 21:05:00	135	Berki	Javadoc és Git tutorial átolvasása, Git repó tanulmányozása.
2012.03.18. 14:50:00	75	Berki	4. heti dokumentáció (Analízis modell kidolgozása II) javítása a legfrissebb osztály és szekvenciadiagramok alapján.

2012.03.19. 5:20:00	100	Berki	6.1.2 és 6.1.3 és 6.2 elkészítése.
2012.03.14. 8:15:00	30	Gerlei	Konzultáció.
2012.03.15. 19:45:00	30	Gerlei	Új értekezlet napló template elkészítése.
2012.03.15. 20:15:00	90	Gerlei	Javadoc tutorial megírása.
2012.03.16. 15:35:00	40	Gerlei	Skeleton projekt létrehozása Eclipse-ben, Javadoc kommentek írása.
2012.03.16. 22:05:00	90	Gerlei	Skeleton kezelőfelületének, gettereknek, a függvények önkírás formájának implementálása
2012.03.17. 13:40:00	40	Gerlei	Függvény önkírás formájának megváltoztatása, setterek implementálása.
2012.03.17. 20:40:00	90	Gerlei	Függvények önkírásának, és visszatérési értékek kiírásának implementálása, Halász kódolásának átnézése.
2012.03.17. 23:18:00	60	Gerlei	Konstruktorok implementálása.
2012.03.18. 0:40:00	20	Gerlei	1. Tesztetesehez GameObject, Key, Exit, Figure, Level osztályok implementálása.
2012.03.18. 10:06:00	76	Gerlei	Konstruktorok önkíratásának beállítása, 1. tesztetese tesztelése.
2012.03.18. 17:50:00	135	Gerlei	2. Tesztetese felhasználóval kommunikációjának implementálása.
2012.03.19. 1:03:00	137	Gerlei	Minden osztályt érintő változtatás a kiíratásban, 4. tesztetese implementálása
2012.03.19. 09:00:00	30	Gerlei	7. tesztetetet érintő hiba javítása.
2012.03.13. 20:15:00	120	Halász	GIT repó tanulmányozás és inicializálás.
2012.03.14. 12:20:00	70	Halász	GIT tutorial készítés.
2012.03.15. 19:45:00	200	Halász	GIT repó tanulmányozás, tutorial befejezése
2012.03.17. 22:40:00	40	Halász	Controller, Application implementálása.
2012.03.18. 0:40:00	20	Halász	1. Tesztetesehez Controller, Application, Direction és Type osztályok implementálása.
2012.03.18. 13:30:00	60	Halász	2. Tesztetese implementálása.
2012.03.19. 10:30:00	100	Halász	JAR fájl készítése, tesztelése, 6.1.2 és 6.1.3 újraírása.
2012.03.14. 8:15:00	30	Kiss	Konzultáció
2012.03.15. 19:45:00	150	Kiss	GIT repó tanulmányozás, tutorial befejezése.
2012.03.19. 8:00:00	60	Kiss	Dokumentum átolvasása, skeleton tesztelése.

2012. 03. 19. 10:00:00	90	Kiss	Dokumentum véglegesítése, skeleon tesztelése.
2012.03.14. 8:15:00	30	Szabó	Konzultáció.
2012.03.14. 16:10:00	20	Szabó	6. heti template, régebbi beadott dokumentumok átnézése.
2012.03.15. 11:00:00	60	Szabó	JDeveloperes osztálydiagramból kód generálás.
2012.03.18. 0:40:00	20	Szabó	1. teszteset implementálása.
2012.03.18. 0:40:00	20	Szabó	1. Tesztetsethez Tile, FullTile, EmptyTile, Field, Frame osztályok implementálása.
2012.03.18. 2:05:00	30	Szabó	Szekvenciadiagramok átnézése, javítása.
2012.03.18. 10:06:00	76	Szabó	FullTile függvényeinek írása.
2012.03.18. 22:10:00	170	Szabó	Git telepítés, kódolás megkezdése (tesztesetek írása: 3,4).
2012.03.19. 3:28:00	120	Szabó	5. teszteset, 6. teszteset, 7. teszteset
2012.03.19. 5:28:00	30	Szabó	Ellenőrzés.

7. hét:

Kezdet	Időtartam	Résztevők	Leírás
2012.03.20. 12:30:00	80	Gerlei Halász Kiss Szabó Berki (30 percet volt jelen)	Értekezletdöntések: - Holnap reggel 8:15-kor mindenki bemegy konzultációra. - Szabó 03.20. 24:00-ig kijavítja a szekvencia diagramokat. - Elvisszük a szekvencia diagramokat kinyomtatva apró változtatásokkal a konzultációra. - Berki fogja megcsinálni a dokumentációs aktualizálását.
2012.03.21. 18:10:00	25	Berki Gerlei Halász Kiss Szabó	Értekezletdöntések: - Use-case-ek megbeszélése péntek délután 16:00-tól 18:00-ig. - Szabó és Gerlei kidolgozza a pénteki értekezletre a 7.1-es (prototípus interfész definíciója) - A többi feladatot a pénteki értekezleten osztjuk ki
2012.03.22. 20:25:00	50	Gerlei Szabó	Értekezletdöntések: - Értekezleten átbeszéljük a bemeneti nyelvet, addig legalább két csapattag, aki nem dolgozott rajta, átnézi. - Kimeneti nyelv elkészítését a pénteki értekezleten kiosztjuk.

			- 03.23-i értekezlet időpontja 16:20
2012.03.23. 15:35:00	75	Berki Halász Kiss Szabó	Értekezletdöntések: - Kimeneti nyelvet megírja Halász szombat 16:00-ra. - Kiss megcsinálja a 7.1.-et, a 7.1.1-et és a 7.3-at szombat 12:00-ra.
2012.03.25. 11:30:00	90	Berki Gerlei Halász Kiss Szabó	Értekezletdöntések: - Halász konzisztenssé teszi a 7.1.1-et és a 7.1.2-t 03.25. 18:00-ig. - A program sorosan fogja kezelni a pályaépítést! - Kiss 03.25. 24:00-ra megcsinálja a végső dokumentumot. - Szabó 15:00-ig elkészít, egy példa bementi fájlt. - Berki és Gerlei átnézi a korábbi javítanivalókat 03.27. 06:00-ig.
2012.03.21. 8:15:00	90	Berki	Konzultáció
2012.03.26. 0:10:00	45	Berki	7. heti dokumentum átolvasása, javítása
2012.03.19. 13:10:00	20	Gerlei	Munkaórákat számoló függvények létrehozása a 7. hét naplójába
2012.03.19. 13:35:00	80	Gerlei	4. heti dokumentáció (Analízis modell kidolgozása II) javítása a skeleon beadás alapján
2012.03.20. 14:00:00	70	Gerlei	Skeleton kód és objektum katalógus, osztály leírás konzisztenssé tétele (függvények, attribútumok sorrendje)
2012.03.21. 8:15:00	90	Gerlei	Konzultáció.
2012.03.24. 16:40:00	55	Gerlei	7.1.2 készítése.
2012.03.24. 21:30:00	35	Gerlei	Eddig elkészült anyagok átnézése
2012.03.21. 8:15:00	90	Halász	Konzultáció.
2012.03.23. 15:20:00	40	Halász	7.2 és 7.3 átnézése.
2012.03.24. 14:40:00	170	Halász	7.1.2 elkészítése.
2012.03.25. 19:00:00	120	Halász	7.1 átolvasása, javítása, kiegészítése.
2012.03.21. 8:15:00	90	Kiss	Konzultáció.
2012.03.23. 12:15:00	150	Kiss	Konzultáció.
2012.03.24. 12:25:00	50	Kiss	7.2 és 7.3 kidolgozása, 7.1.2 átolvasása.
2012.03.24.	30	Kiss	7.1 és 7.4 készítése.

15:00:00			
2012.03.25. 23:20:00	80	Kiss	7.1.2 készítése.
2012.03.20. 20:56:00	220	Szabó	Szkeleton beadáshoz szekvenciadiagramok javítása, JDeveloperbe való átírás.
2012.03.21. 8:15:00	90	Szabó	Konzultáció.
2012.03.21. 1:24:00	68	Szabó	Szekvenciadiagramok befejezése, ellenőrzése, dokumentum elkészítése.
2012.03.24. 18:05:00	80	Szabó	Heti dokumentum összeszerkesztése, átnézése.
2012.03.25. 14:10:00	50	Szabó	palya_pelda.txt megírása, 7.1.2 változtatása/kommentezése
2012.03.26 00:16:00	30	Szabó	Végleges dokumentáció átnézése.

8. hét:

Kezdet	Időtartam	Résztevők	Leírás
2012.03.28 18:10:00	58	Berki Gerlei Halász Kiss Szabó	Értekezlet. Döntések: - Berki megcsinálja a 8.1-et 03.30. 12:00-ig. - Szabó megcsinálja 03.29. 10:00-ig a bemeneti és a kimeneti nyelv javítását. - Gerlei megtervezi a pályát 03.28. 24:00-ig, ami jó a legtöbb tesztesetre. - Halász, Gerlei és Szabó, Kiss megcsinálják a 8.2-t 03.31. 24:00-ig.
2012.03.27. 15:00:00	30	Berki	Munkaóra táblázat, és Javadoc javításának megtervezése.
2012.03.30. 9:00:00	190	Berki	8.1 megírása az objektumkatalógus és az osztálydiagram alapján.
2012.04.01. 20:15:00	105	Berki	8.1 befejezése, átnézése a javított osztálydiagram alapján.
2012.03.27. 15:00:00	30	Gerlei	Munkaóra táblázat, és Javadoc javításának megtervezése
2012.03.28. 8:15:00	45	Gerlei	Konzultáció.
2012.03.29. 22:30:00	90	Gerlei	Dokumentum rendezése, 10. teszteset kidolgozása.
2012.03.31. 14:20:00	95	Gerlei	Maradék tesztesetek kidolgozása
2012.03.31. 20:45:00	35	Gerlei	Dokumentum átnézése.
2012.03.29. 15:00:00	90	Halász	Szint méretek meghatározása, teszteset1 megírása, teszteset2 megírásának elkezdése.
2012.03.29. 22:30:00	90	Halász	Dokumentum rendezése, 10. teszteset kidolgozása
2012.03.30.	100	Halász	Szintek véglegesítése, 2. teszteset

10:50:00			bemenetének elkészítése, vektor geometrikus rész tervezése.
2012.04.02. 00:15:00	300	Halász	Fizikai modell véglegesítése, tesztesetek átnézése, esetleges javítása, 8. heti dokumentum szerkesztése.
2012.03.28. 8:15:00	45	Kiss	Konzultáció.
2012.03.28. 22:00:00	150	Kiss	Heti dokumentáció átnézése, formailag végleges dokumentum szerkesztése.
2012.04.02. 0:40:00	30	Kiss	8. heti dokumentum szerkesztése.
2012.04.02. 03:30:00	60	Kiss	8. heti dokumentum szerkesztése, kimeneti nyelv pontosítása.
2012.03.28. 8:15:00	45	Szabó	Konzultáció.
2012.03.28. 20:55:00	40	Szabó	7. heti beadandó javítása, átnézése: szglab4_07_mod, Prototípus példa megírása
2012.03.29. 15:00:00	100	Szabó	Palya méretek meghatározása, teszteset1 megírása, teszteset2 megírásának elkezdése.
2012.03.31. 20:45:00	35	Szabó	Dokumentum átnézése.
2012.04.01. 0:40:00	60	Szabó	Class diagram javítása, dokumentum átnézése
2012.04.01. 16:00:00	120	Szabó	Pályák véglegesítése (függelék), fizika meghatározása, 2es teszteset megírása.
2012.04.01. 18:44:00	68	Szabó	Tesztesetek átnézése, 8.3 megírása.
2012.04.01. 20:54:00	20	Szabó	Ellenőrzött funkcionalitás, várható hibahelyek megírása.

10. hét:

Kezdet	Időtartam	Résztvevők	Leírás
2012.04.03. 16:20	30	Gerlei Halász Kiss Szabó	Értekezlet, Döntések: - Új Google Docs lap döntések számontartására - Halász ezen a héten ráér elkezdni a kódolást, a txt betöltés lehetőleg kész lesz ezen a héten, valamint a konstruktorok. - Gerlei, Halász hétvégén tud vele foglalkozni.
2012.04.08. 16:40	120	Gerlei Kiss Szabó	Értekezlet, Döntések: - Int levelsPathIndex formátumú lett a szint betöltő függvénye. - Csak egy darab koordináta rendszer lesz, nem lesz külön a

			pályának és a csempének. - Diszkrét koordinátarendszerben dolgozunk. - A clearActualLevel helyett loadNextLevel lesz, ami kinullázza az aktuális szintet és betölti a következőt. - Ha nem találunk más, egyszerűbb megoldást (pl. egyszálút) a SimulateWorld implementálására 04.10. 12:00-ig akkor Szabó javaslatát valósítjuk meg.
2012.04.08. 23:20	60	Halász Kiss	Értekezlet, Döntések: - Olyan vektorgeometriai modellt választunk, ami 3 vektort összegez (gravitáció, irány és előző elmozdulás) ez alapján számoljuk ki a bábu következő koordinátáját.
2012.04.02. 20:30:00	60	Berki	Objektum katalógus(4.1 és 4.2) módosítása a 8.1 alapján
2012.04.03. 17:15:00	160	Berki	A kód kommentjeit és verziószámait javítottam
2012.04.04. 8:15:00	15	Berki	Konzultáció
2012.04.17. 0:01:00	45	Berki	batch fájlok készítése, azok helyes működésének tesztelése
2012.04.17. 10:45:00		Berki	10.1 10.3 és 10.4 megírása
2012.04.07. 17:30:00	35	Gerlei	8. heti dokumentáció, és a kód állásának átnézése
2012.04.07. 19:45:00	45	Gerlei	Kód javítása
2012.04.08. 22:05:00	100	Gerlei	Az értekezleten megbeszélte dolgok javítása, néhány függvény implementálása a kódban.
2012.04.12. 23:00:00	35	Gerlei	Projekt állásának átnézése, kisebb hibák javítása a kódban
2012.04.13. 10:25:00	170	Gerlei	Kód és javadoc javítása, setFramePoint metódus implementálása, endFullTile bemeneti parancs hatásainak implementálása
2012.04.13. 15:25:00	150	Gerlei	Kulcs, Kijárat, Bábu hozzáadásához szükséges metódusok implementálása, hiányzó getterek, setterek elkészítése
2012.04.13. 22:00:00	35	Gerlei	Field osztály módosítása, Polygon osztály használatával a terep tárolásának és kezelésének újra implementálása
2012.04.15.	47	Gerlei	Bábu B hozzáadásnak megírása, Polygon

13:28:00			osztály használata miatt felmerülő hibák javítása
2012.04.15. 23:10:00	30	Gerlei	Application loadCommand metódusának endFullTile, addFigureA és addFigureB bemeneti parancsok feldolgozásában lévő hiba javítása
2012.04.16. 22:00:00	45	Gerlei	Application loadLevel metódusban felmerült hibák felderítése és javítása
2012.04.05. 18:00:00	120	Halász	8. heti dokumentum és a kód összeegyeztetése, konstruktorok átírása, kód előkészítése az implementáláshoz Applicationtól a Figure osztályig.
2012.04.06. 18:45:00	70	Halász	Összeegyeztetés folytatása, FullTile osztályig kész.
2012.04.06. 22:00:00	50	Halász	Összeegyeztetés befejezve.
2012.04.07. 13:15:00	130	Halász	Direction, EmptyTile, Exit osztályok implementálása, Fieldben, Frame-ben, Figure-ban és FullTile-ban csak bizonyos függvények.
2012.04.09. 18:40:00	20	Halász	Simulateworld átgondolása.
2012.04.10. 0:30:00	30	Halász	Modell átgondolása, dokumentumok javítása.
2012.04.12. 23:15:00	240	Halász	Direction osztály kibővítése, FullTile.calculateNewCoord implementálása.
2012.04.15. 22:30:00	330	Halász	FullTile.calculateNewCoord korrigálása, FullTile.moveFigure implementálása. Egyéb osztályok kódjainka kis mértékű módosítása, majd debuggolás.
2012.04.16. 19:10:00	180	Halász	CalculateNewCoord korrigálása, debuggolás.
2012.04.16. 23:30:00	220	Halász	Korrekciók a FullTile.calculateNewCoordban és a FullTile.moveFigure-ben. Bemeneti nyelv parancsfeldolgozásában módosítások, Key és Exit osztályok kiíratásainak implementálása.
2012.04.17. 11:05:00	331	Halász	Hibajavítás-
2012.04.04. 8:15:00	15	Kiss	Konzultáció
2012.04.05. 19:30:00	30	Kiss	8. heti dokumentáció átnézése
2012.04.07. 17:00:00	45	Kiss	Eddigi kód átgondolása, értelmezése.
2012.04.07. 18:40:00	20	Kiss	Simulateworld átgondolása.
2012.04.07. 21:00:00	30	Kiss	Értekezletnaplók szerkesztése.

2012.04.08. 2:12:00	30	Kiss	Eddigi kódok átgondolása, átnézése.
2012.04.10. 0:30:00	30	Kiss	Modell átgondolása, dokumentumok javítása.
2012.04.15. 19:15:00	30	Kiss	A végső modell és a prototípus közti különbségek átgondolása, értekezletnaplók rendbetétele. Még implementálatlan függvények feltérképezése, tesztek futtatása az eddigi programon.
2012.04.15. 20:00:00	70	Kiss	Hibakeresés.
2012.04.04. 8:15:00	15	Szabó	Konzultáció
2012.04.08. 10:05:00	60	Szabó	Eddigi kódok átnézése, 8. heti változtatott dokumentáció átnézése.
2012.04.07. 18:40:00	20	Szabó	Simulateworld átgondolása.
2012.04.08. 10:50:00	30	Szabó	SimulateWorld szálakkal való megvalósítása. Verzio: 1.0
2012.04.08. 13:40:00	10	Szabó	SimulateWorld szálakkal való megvalósítása. Verzio: 2.0, ami mégse kell. A 2.0-ás verzió nem készült el.
2012.04.08. 22:04:00	170	Szabó	Kódolás: bemeneti nyelv beolvasása.
2012.04.10. 12:47:00	133	Szabó	Kódolás folytatása.
2012.04.11. 0:54:00	60	Szabó	Kódolás, bemeneti/kimeneti nyelv megírása.
2012.04.12. 11:15:00	160	Szabó	Kódolás: 3. tesztet kész, pályaépítés egy kis része kész.
2012.04.13. 20:30:00	100	Szabó	Kódolás: pályaépítés átnézése, addFieldPoint megírása.
2012.04.15. 22:00:00	30	Szabó	Kimeneti fájl ellenőrző program megírása.
2012.04.16. 1:30:00	210	Szabó	Kódolás, debuggolás.

11. hét:

Kezdet	Időtartam	Résztvevők	Leírás
2012.04.18. 18:35:00	35	Berki Gerlei Halász Kiss Szabó	Értekezlet, döntések: - Berki 04.21. 24:00-ig elkészíti a 11.1-et. - 04.19. 14:15-kor az SCH-ban a következő értekezleten kidolgozzuk az 11.2. 11.3. 11.4-et. Ami feladat marad azt majd a 04.19-i értekezlet végén kiosztjuk.
2012.04.19. 14:30:00	102	Berki Gerlei	Értekezlet, döntések: - Berki digitalizálja az osztály diagramot

		Halász Kiss Szabó	04.20. 20:00-ig. - Halász és Kiss 04.21. 12:00-ig megcsinálja az új objektumkatalógust. - Gerlei a szekvencia diagramokat 04.21. 20:00-ig. Berki digitalizálja őket 04.22. 12:00-ig.
2012.04.17. 21:45:00	20	Berki	Munkaórákat számoló függvények létrehozása a 11. hét naplójába.
2012.04.21. 17:00:00	30	Berki	Osztálydiagram digitalizálása part1.
2012.04.22. 14:15:00	15	Berki	11.1 grafikus felület tervének elkészítése.
2012.04.22. 15:10:00	240	Berki	Szekvenciadiagramok digitalizálása.
2012.04.23. 0:10:00	15	Berki	11.1 szöveges rész megírása.
2012.04.21. 20:45:00	120	Gerlei	Model - View kapcsolatát bemutató szekvencia diagramok rajzolása.
2012.04.22. 16:10:00	25	Gerlei	Bedigitalizált szekvencia diagramok ellenőrzése.
2012.04.22. 16:35:00	35	Gerlei	Csapatlogó megtervezése.
2012.04.22. 21:45:00	25	Gerlei	11.4 (Kapcsolat az alkalmazói rendszerrel) megírása, beadandó dokumentum átnézése
2012.04.18. 8:15:00	105	Halász	Konzultáció.
2012.04.21. 10:40:00	110	Halász	Objektumkatalógus felépítése.
2012.04.22. 17:15:00	15	Halász	Szekvenciadiagramok átnézése, javítási javaslatok tétele.
2012.04.18. 8:15:00	105	Kiss	Konzultáció.
2012.04.19. 11:30:00	15	Kiss	Értekezletnapló digitalizálása.

13. hét:

Kezdet	Időtartam(perc)	Résztevők	Lejírás
2012.04.25. 10:35:00	35	Gerlei Halász Kiss Szabó	Értekezlet, döntések: - Halász pénteken a Direction osztály átírásával, a fizika finomításával kezdi. - Kiss péntekre segédletet készít a Java grafikáról.
2012.04.23. 16:15:00	25	Berki	Munkaórakat számoló függvények létrehozása a 12. hét naplójába

2012.04.25. 9:05:00	25	Berki	Konzultáció.
2012.04.25. 9:05:00	25	Gerlei	Konzultáció.
2012.04.28. 17:40:00	30	Gerlei	Grafikus kirajzolás átgondolása, Kiss tutorialja alapján.
2012.04.28. 21:21:00	100	Gerlei	Grafikus kirajzolás további tervezése, implementálás átgondolása.
2012.04.29. 21:45:00	80	Gerlei	Grafikus kirajzolás példaprogram készítése.
2012.04.30. 9:45:00	105	Gerlei	Grafikus kirajzolás példaprogram készítése, ez alapján a további munkálatok felvázolása.
2012.05.01. 21:40:00	160	Gerlei	Grafikus osztályok implementálása.
2012.05.01. 14:30:00	110	Gerlei	Végleges méretarányú példa képek rajzolása, programhoz használandó grafikák megrajzolása.
2012.05.01. 17:40:00	35	Gerlei	Grafikához szükséges képek feltöltése, végleges elnevezése, kódba illesztése.
2012.05.01. 15:40:00	45	Gerlei	FigureView, View paintComponent metódusának implementálása, hibajavítás a kódban, értekezlet napló javítása.
2012.04.25. 9:05:00	25	Halász	Konzultáció
2012.04.30. 17:30:00	180	Halász	Fizika módosítása, Direction osztály fejlesztése, a kirajzoláshoz szükséges koordináta-rendszer-váltó függvények implementálása.
2012.05.01. 21:40:00	160	Halász	Grafikus osztályok implementálása.
2012.05.05. 17:45:00	100	Halász	Paint metódus implementálása az ExitView, KeyView, FullTileView és FigureView osztályokban.
2012.05.05. 21:50:00	40	Halász	View osztály metódusainak implementálása.

2012.05.06. 15:00:00	420	Halász	View-k és fizika debuggolása.
2012.05.07. 10:15:00	130	Halász	Kétjátékosos illentyűzetkezelés implementálása.
2012.05.07. 13:50:00	180	Halász	Fizika végső debug, grafikus osztályok hibajavítása.
2012.04.25. 21:57:00	121	Kiss	Kutatómunka: java graphics osztály.
2012.04.26. 0:45:00	15	Kiss	Értekezletnaplók és Értekezletek táblázat összegegyeztetése.
2012.04.26. 12:30:00	50	Kiss	Kutatómunka: Java graphics osztály - shapes kirajzolás.

2012.04.26. 18:00:00	60	Kiss	Kutatómunka: Java graphics osztály2D - shapes kirajzolás.
2012.04.26. 21:55:00	110	Kiss	Segédlet készítése a java grafikus interface-ről.
2012.04.29. 23:00:00	60	Kiss	Grafikus tervek.
2012.05.06. 20:15:00	55	Kiss	Dokumentumszerkesztés.
2012.05.06. 15:00:00	30	Kiss	Heti dokumentum javítása, Sógó megírása.
2012.05.07. 18:15:00	20	Kiss	Fájllista javítása, diagramok javítása.
2012.05.03. 18:18:00	22	Szabó	Controller osztály változtatása.
2012.05.03. 20:04:00	210	Szabó	Controller osztályban szálak és keyboardlistener megírása.
2012.05.05. 19:50:00	100	Szabó	Controller javítása, alap menü készítés.
2012.05.05. 22:50:00	30	Szabó	Controller javítása, új játék gomb működik, betölti az első pályát.
2012.05.06. 15:00:00	210	Szabó	View-k debuggolása.
2012.05.07. 12:38:00	230	Szabó	Menu, Controller kódolása.

Tartalom

2. Követelmény, projekt, funkcionalitás	2
2.1 Követelmény definíció	2
2.2 Projekt terv	3
2.3 Feladatleírás.....	5
2.4 Szótár.....	6
2.5 Essential use-case-ek	7
2.5.1 Use-case diagram.....	7
2.5.2 Use-case leírások	7
3. Analízis modell kidolgozása	9
3.1 Objektum katalógus.....	9
3.1.1. Application	9
3.1.2. Controller	9
3.1.3. EmptyTile.....	9
3.1.4. Exit	9
3.1.5. Field.....	9
3.1.6. Figure	9
3.1.7. FullTile	9
3.1.8. Key	9
3.1.9. Level.....	9
3.2 Osztályok leírása	9
3.2.1. Application	9
3.2.2. Controller	10
3.2.3. EmptyTile.....	10
3.2.4. Exit	11
3.2.5. Field.....	11
3.2.6. Figure	11
3.2.7. FullTile	12
3.2.8. GameObject.....	12
3.2.9. IMoveable.....	12
3.2.10. Key	13
3.2.11. Level.....	13
3.2.12. Tile	13
3.3 Statikus struktúra diagramok.....	15
3.4 Szekvencia diagramok.....	16
3.5 State-chartok.....	20
4. Analízis modell kidolgozása II.....	21

4.1	Objektum katalógus.....	21
4.1.1	Application	21
4.1.2	Controller.....	21
4.1.3	Direction	21
4.1.4	EmptyTile.....	21
4.1.5	Exit	21
4.1.6	Field.....	21
4.1.7	Figure	21
4.1.8	Frame.....	21
4.1.9	FullTile	21
4.1.10	GameObject.....	21
4.1.11	Key	21
4.1.12	Level.....	22
4.1.13	Tile	22
4.1.14	Type.....	22
4.2	Osztályok leírása	22
4.2.1	Application	22
4.2.2	Controller	22
4.2.3	Direction.....	23
4.2.4	EmptyTile.....	23
4.2.5	Exit	23
4.2.6	Field.....	24
4.2.7	Figure	24
4.2.8	Frame.....	24
4.2.9	FullTile	25
4.2.10	GameObject	26
4.2.11	Key.....	26
4.2.12	Level.....	26
4.2.13	Tile	27
4.2.14	Type	27
4.3	Statikus struktúra diagramok.....	28
4.4	Szekvencia diagramok.....	29
4.5	State-chartok.....	49
5.	Szkeleton tervezése	50
5.1	A szkeleton modell valóságos use-case-ei	50
5.1.1	Use-case diagram	50
5.1.2	Use-case leírások.....	52

5.2	Architektúra.....	54
5.2.1	Szkeleton architektúrális leírása.....	54
5.2.2	Teszteset #1	54
5.2.3	Teszteset #2	54
5.2.4	Teszteset #3	54
5.2.5	Teszteset #4	55
5.2.6	Teszteset #5	55
5.2.7	Teszteset #6	55
5.2.8	Teszteset #7	55
5.3	A szkeleton kezelői felületének terve, dialógusok	55
5.4	Szekvencia diagramok a belső működésre	57
5.4.1	Teszteset #1	57
5.4.2	Teszteset #2	59
5.4.3	Teszteset #3	60
5.4.4	Teszteset #4	65
5.4.5	Teszteset #5	69
5.4.6	Teszteset #6	70
5.4.7	Teszteset #7	72
6.	Szkeleton beadás	78
6.1	Fordítási és futtatási útmutató	78
6.1.1	Fájllista	78
6.1.2	Fordítás	78
6.1.3	Futtatás	78
6.2	Értékelés	79
7.	Prototípus koncepciója	80
7.1	Prototípus interface-definíciója	80
7.1.1	Az interfész általános leírása	80
7.1.2	Bemeneti nyelv.....	80
7.1.3	Kimeneti nyelv	82
7.2	Összes részletes use-case	84
7.3	Tesztelési terv.....	86
7.4	Tesztelést támogató segéd- és fordítóprogramok specifikálása	87
8.	Részletes tervek	88
8.1	Osztályok és metódusok tervei.	88
8.1.1	Application	88
8.1.2	Controller	88
8.1.3	Direction.....	88

8.1.4 EmptyTile	89
8.1.5 Exit	89
8.1.6 Field.....	89
8.1.7 Figure	90
8.1.8 Frame.....	90
8.1.9 FullTile	91
8.1.10 GameObject.....	91
8.1.11 Key	92
8.1.12 Level.....	92
8.1.13 Tile	92
8.1.14 Type.....	93
8.2 A tesztek részletes tervei, leírásuk a teszt nyelvén.....	94
8.2.1 Teszteset1: Pálya betöltése.....	94
8.2.2 Teszteset2: A bábu mozgatása	95
8.2.3 Teszteset3: Csempe mozgatása	97
8.2.4 Teszteset4: A bábu át tud haladni a csempén.....	98
8.2.5 Teszteset5: A bábu nem tud áthaladni a csempén.....	99
8.2.6 Teszteset6: A bábu meghal.....	100
8.2.7 Teszteset7: Bábu kulcsot talál, majd eléri a kijáratot és kimegy rajta.	100
8.2.8 Teszteset8: Bábu áthalad egy felszedett kulcson, kijáratot talál, de nincs nála elég kulcs.	103
8.2.9 Teszteset9: Két bábu találkozik.....	104
8.2.10 Teszteset10: A bábu felszedi az első kulcsot, zuhanás közben éri el a második kulcsot, majd a kijáratot.	105
8.3 A tesztelést támogató programok tervei	106
10. Prototípus beadása.....	107
10.1 Fordítási és futtatási útmutató	107
10.1.1 Fájllista	107
10.1.2 Fordítás.....	108
10.1.3 Futtatás	108
10.2 Tesztek jegyzőkönyvei	108
10.2.1 Teszteset#1	108
10.2.2 Teszteset#2	108
10.2.3 Teszteset#3	108
10.2.4 Teszteset#4	108
10.2.5 Teszteset#5	109
10.2.6 Teszteset#6	109
10.2.7 Teszteset#7	109

10.2.8 Teszteset#8	109
10.2.9 Teszteset#9	109
10.2.10 Teszteset#10	109
10.3 Értékelés	110
10. Részletes tervek kiegészítő.....	111
11. Grafikus felület specifikációja.....	132
11.1 A grafikus interfész	132
11.1.1 A felület működési elve	133
11.1.2 A felület osztály-struktúrája	133
11.2 A grafikus objektumok felsorolása.....	135
11.2.1 ExitView.....	135
11.2.2 FigureView	135
11.2.3 FullTileView	135
11.2.4 IdleThread	136
11.2.5 KeyView.....	136
11.2.6 MainThread	137
11.2.7 Paintable Interface	137
11.2.8 View	137
11.2.9 Controller (bővítés)	138
11.3 Kapcsolat az alkalmazói rendszerrel	139
13. Grafikus változat beadása.....	143
13.1 Fordítási és futtatási útmutató	143
13.1.1 Fájllista	143
13.1.2 Fordítás és telepítés	144
13.1.3 Futtatás	144
13.2 Értékelés	144
14. Összefoglalás.....	145
14.1. A projektre fordított összes munkaidő	145
A projektre fordított össze munka (percben):.....	145
A projektre fordított össze munka (órán):	145
Munkaóra diagramok:	146
Értekezletek és egyéni munkaórák megoszlása:	148
Összesített munkaóra diagram:	149
Végső értékelés:	149
14.2 A projekt során beadott forrásfájlok és azok sorainak a száma	150
14.2.1. Skeleton.....	150
14.2.2. Prototípus	150

14.2.3. Garfikus változat	152
14.3 A projekt összegzése	153
14.3.1. Mit tanultunk?	153
14.3.2. Mi volt a legnehezebb?	153
14.3.3. Mi volt a legkönnyebb?	154
14.3.4. Pontszám és feladatok összhangja.....	154
14.3.5. Javasolt változások	154
14.3.6. Projekt ajánlatok.....	154
Napló	155
2. hét:	155
3. hét:	157
4. hét:	158
5. hét:	160
6. hét:	161
7. hét:	164
8. hét:	166
10. hét:	167
11. hét:	170
13. hét:	171