**Analízis modell kidolgozása I.**

35 – *pupakok*

Konzulens:

**Koczó Attila**

Csapattagok:

|  |  |  |
| --- | --- | --- |
| Tarsoly Levente | E1IK75 | tarsolyl@edu.bme.hu |
| Németh Gergely | TA9G0R | gergo.h.nemeth@gmail.com |
| Schulcz Gábor | EP3I9R | schulcz.gabor6@gmail.com |
| Czotter Benedek | TFB4FY | czottibeni@gmail.com |
| Hermann Máté Béla | T83K5I | mate3829@gmail.com |

2024.03.04.

# **3.** **Analízis modell kidolgozása**

## **3.1** **Objektum katalógus**

### **3.1.1 Ajtó**

A szoba része, ezen keresztül lehet a szobák között mozogni.

### **3.1.2 Camembert** Egy tárgy, ami szobákból felvehető. Használatával elgázosítható egy szoba.

### **3.1.3 Egyirányú ajtó**

Az ajtó egy speciális fajtája, ami csak egyik irányba engedi a közlekedést.

### **3.1.4 Elátkozott szoba**

Az szoba egy olyan fajtája aminek vannak eltűnő, majd előtűnő ajtói.

### **3.1.5 Gázos Szoba**

A szoba egy olyan fajtája, ami gázzal van megtelítve ami elkábítja a karaktereket belépéskor.

### **3.1.6 Hallgató**

Egy irányítható karakter. Célja a logarléc nevű tárgy megszerzése.

### **3.1.7 Logarléc** Egy tárgy, ami szobákból felvehető. Ennek felvételével a hallgatók megnyerik a játékot.

### **3.1.8 Maszk** Egy tárgy, ami szobákból felvehető. Ennek segítségével nem vesztik eszméletüket a játékosok

### **3.1.9 Oktató**

A játék által irányított karakter. Célja a hallgatók akadályozása.

### **3.1.10 Rongy** Egy tárgy, ami szobákból felvehető. Ennek használatával megbénulnak az oktatók.

### **3.1.11 Söröspohár** Egy tárgy, ami szobákból felvehető. Ez védelmet nyújt az oktatók ellen.

### **3.1.12 Szoba**

A pálya része, ahol a játék történik. Ebben lehet felvenni a tárgyakat, és itt mozognak a karakterek.

### **3.1.13 Tranzisztor** Egy tárgy, ami szobákból felvehető. Ennek használatával képesek hallgatók szobák közt ajtó nélkül közlekedni

### **3.1.14 TVSZ** Egy tárgy, ami szobákból felvehető. Ez védelmet nyújt az oktatók ellen

# **3.2** **Statikus struktúra diagramok**

# 

## **3.3** **Osztályok leírása**

### **3.3.1 Camembert**

#### · **Felelősség**

A feladat leírásában lévő Dobozolt káposztás camembert tárgyat reprezentálja. Ennek használatával egy szoba átalakítható gázos szobává.

#### · **Ősosztályok**

* Tárgy

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

#### · **Metódusok**

* void use(): A paraméterben átadott szobában elhasználódik, ennek köszönhetően az a szoba gázos lesz. Ez után a tárgy eltűnik.

### **3.3.2 Elátkozott\_szoba**

#### · **Felelősség**

A szobának egy olyan fajtájáért felel aminek vannak a szobán túlmutató tulajdnságai, ugyanis lehetnek eltűnő és előtűnő ajtói.

#### · **Ősosztályok**

Szoba

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

* ArrayList<Szoba> eltuntajto : eltároljuk az eltűnt ajtókat, hogy később vissza tudjuk állítani.

#### · **Metódusok**

* void eltunik() : eltüntet egy ajtót, így azon nem lehet átmenni amíg nem kerül elő újra.
* void elotunik() : előtűntet egy korábban eltűnt ajtót, így azon újra át lehet menni.

### **3.3.3 Game**

#### · **Felelősség**

Ez felel a program indulásáért és megállásáért.

#### · **Ősosztályok**

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

* Pálya palya : a kezdeti pálya
* int szamlalo : ez adja meg milyen hosszú a játék, ha a leteltéig nem sikerül megtalálni a logarlécet akkor a hallgatók vesztettek.

#### · **Metódusok**

* void start() : a függvény ami indítja és zárja a játékot.

### **3.3.4 Hallgató**

#### · **Felelősség**

Ez az osztály felel az irányítható karakterért a hallgató tulajdonságaiért.

#### · **Ősosztályok**

Karakter

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

#### · **Metódusok**

* void eldob(Tárgy) : a hallgató saját akaratból is dobhat el tárgyat, ennek a függvénynek a segítségével.
* void kibukik() : ha a hallgató kibukik, akkor megszűnik létezni és a tárgyai megsemmisülnek.

### **3.3.5 *Karakter***

#### · **Felelősség**

Ez az absztrakt osztály felel a karakterek a működéséért. Ez kezeli azokat az eseményeket amik egy karakterrel történhetnek, a tárgyfelvétel és letétel, mozgás és eszméletvesztés.

#### · **Ősosztályok**

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

* Szoba szoba : a szoba ahol a karakter tartózkodik.
* ArrayList<Tárgy> taska : tárgyak listája ami a karakternél van.

#### · **Metódusok**

* void mozog(Szoba) : a karakter mozgását végzi
* void felvesz(Tárgy) : felvesz egy tárgyat és eltárolja a táskában.
* void eszméletvesztés() : ez végzi az eszméletvesztéskor beteljesülő eseményeket, a tárgy és kimaradást.

### **3.3.6** **Logarléc**

#### · **Felelősség**

Ennek a tárgynak felvételével a hallgatók megnyerik a játékot.

#### · **Ősosztályok**

Tárgy

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

#### · **Metódusok**

* void use(): Ez automatikusan lefut, amikor a hallgató felveszi ezt a tárgyat, és ezzel megnyerik a játékot.

### **3.3.7 Maszk**

#### · **Felelősség**

A feladat leírásában lévő FFP2es maszk tárgyat reprezentálja. Megvédi ezt a tárgyat birtokló karaktereket az eszméletvesztéstől a gázos szobákban.

· **Ősosztályok**

Tárgy

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

* int töltet: Ez számlálja azt, hogy az adott maszk hányszor használható még.Ha ez letelik, akkor a tárgy már nem funkcionál.

#### · **Metódusok**

* void use(): Ha a játékos gázos szobába lép, akkor használja ezt a tárgyat, ezzel a töltete eggyel csökken, viszont nem veszti eszméletét és folytathatja a játékot.

### **3.3.8 Oktató**

#### · **Felelősség**

Ez az osztály felel a játék által irányított karakter, az oktató tulajdonságaiért.

#### · **Ősosztályok**

Karakter

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

#### · **Metódusok**

* void megbenul() : ha van megfelelő tárgy a hallgatónál és egy szobába kerülnek akkor az oktató megbénul és nem tud kibuktatni hallgatót.

### **3.3.9 Pálya**

#### · **Felelősség**

Ez az osztály felel a játék generálásáért, valamint az oktatók mozgatásáért.

#### · **Ősosztályok**

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

* ArrayList<Hallgató> hallgatok: összes hallgató listája
* ArrayList<Oktató> oktatok: összes oktató listája
* ArrayList<Szoba> szobak: összes szoba listája

#### · **Metódusok**

* void general() : legenerálja a pályát, vagyis a szobákat, azoknak a különböző típusait, valamint a játékosokat és tárgyakat.
* void leptet() : lép az oktatókkal, valamint a véghezviszi a pályabeli változtatásokat.

### **3.3.10 Rongy**

#### · **Felelősség**

A feladat leírásában lévő Nedves táblatörlő rongy tárgyat reprezentálja. Ennek segítségével megbéníthatók az oktatók, ha egy szobában tartózkodnak egy olyan hallgatóval, amelynél van egy ilyen tárgy.

#### · **Ősosztályok**

Tárgy

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

* int töltet: Ez számlálja azt, hogy az adott rongy mennyi ideig használható még. Ha ez letelik, akkor a tárgy már nem funkcionál.

#### · **Metódusok**

* void use(): Ha egy hallgató egy szobába kerül egy oktatóval, akkor azt megbénítja egy adott időre.

### **3.3.11 Söröspohár**

#### · **Felelősség**

A feladat leírásában lévő Szent söröspohár tárgyat reprezentálja. Ez egy ideig képes védelmet nyújtani az oktatók ellen a hallgatóknak.

#### · **Ősosztályok**

Tárgy

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

* int töltet: Ez számlálja azt, hogy az adott söröspohár mennyi ideig használható még. Ha ez letelik, akkor a tárgy már nem funkcionál.

#### · **Metódusok**

* void use(): Amíg egy hallgatónál van ez a tárgy, ez fogja a töltetet csökkenteni

### **3.3.12 Szoba**

#### · **Felelősség**

Ez az osztály felel a szobának a működéséért. Itt fog történni a játék, itt fognak mozogni a karakterek.

#### · **Ősosztályok**

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

* bool gazos : megmutatja, hogy gázos-e a szoba
* int befogadokepesseg : hány fő lehet a szobában
* ArrayList<Tárgy> targyak : a szobában található tárgyak listája
* ArrayList<Szoba> regiszobak : tárolja a szobákat amiből összeolvadt egy szoba, hogy az alapján szét is tudjon osztódni.
* ArrayList<Karakter> karakterek : a szobában tartózkodó karakterek
* ArrayList<Szoba> szomszedok : szomszédos szobák listája

#### · **Metódusok**

* void targy\_elhelyezese() : elhelyez egy tárgyat a szobában amit fel lehet venni.
* void targy\_eltuntetese(Tárgy) : eltűntet egy tárgyat, hogy ne lehessen felvenni többször, ha egy játékos felveszi.

### **3.3.13 *Tárgy***

#### · **Felelősség**

Egy absztrakt osztály, amely a tárgyakat reprezentálja.

#### · **Ősosztályok**

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

* Szoba hol: Ez azt írja, le, hogy a tárgy melyik szobában van, ha ez egy karakternél van ez a tárgy, akkor is változik a karaktertől függően.

#### · **Metódusok**

* void use(): A tárgyat használja a karakter.

### **3.3.14 Tranzisztor**

#### · **Felelősség**

A feladat leírásában lévő Tranzisztor tárgyat reprezentálja. Ennek segítségével képesek a hallgatók két tranzisztor összekapcsolása után egyik nem szomszédos szobából átmenni a másikba, ajtók használata nélkül.

#### · **Ősosztályok**

Tárgy

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

* bool aktív: Ez azt írja le, hogy a tranzisztor aktív-e
* Szoba hova: Ez azt tárolja, hogy ha össze van kötve két tranzisztor, akkor melyikbe visz át a használatakor

#### · **Metódusok**

* void use(): Első használatkor aktívvá teszi a tranzisztorokat, és az elsőt leteszi. Második használatkor leteszi a másikat a jelenlegi szobába és átmegy az első letett tranzisztorhoz, és a második aktivitását kikapcsolja. Ez után további használatokkor ha aktív a tranzisztor átmegy a karakter a párjához, és a kiindulót be, a másikat pedig beaktiválja.

### **3.3.15 TVSZ**

#### · **Felelősség**

A feladat leírásában levő TVSZ denevérbőrre nyomtatott példányát reprezentálja. Megvédi a hallgatót a vele egy szobában lévő oktató általi kibuktatástól.

#### · **Ősosztályok**

Tárgy

#### · **Interfészek**

#### · **Asszociációk**

#### · **Attribútumok**

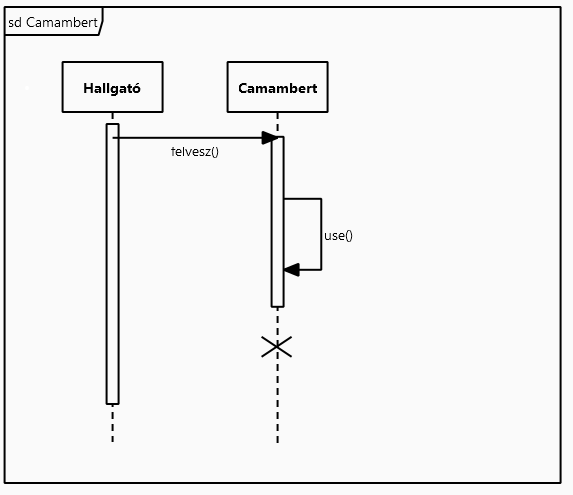
* int töltet: Ez számlálja azt, hogy az adott TVSZ hányszor használható még.

#### · **Metódusok**

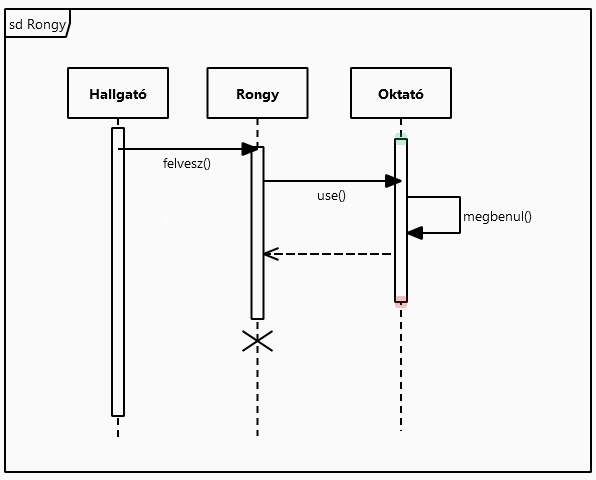
* void use(): Ha a játékos egy szobába lép, amelyben oktató is tartózkodik, akkor használja ezt a tárgyat, ezzel a töltete eggyel csökken, viszont folytathatja a játékot.

**3.4** **Szekvencia diagramok**

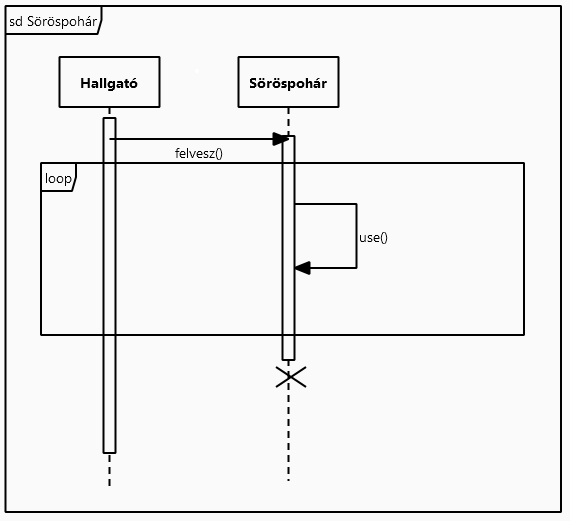
**3.4.1 Káposztás camembert használata**



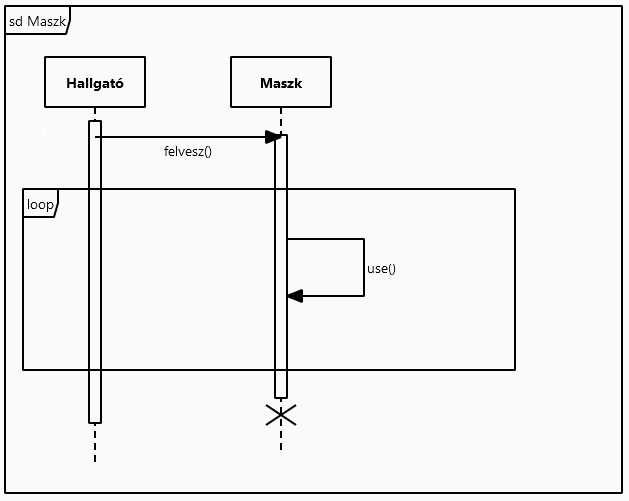
# **3.4.2 Rongy használata**



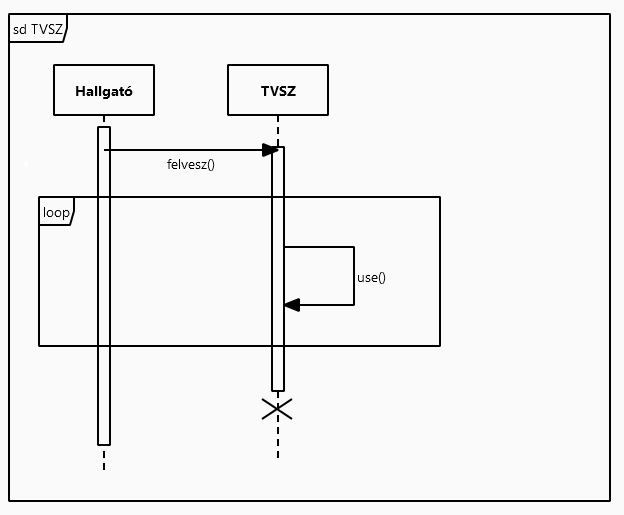
**3.4.3 Söröspohár használata**



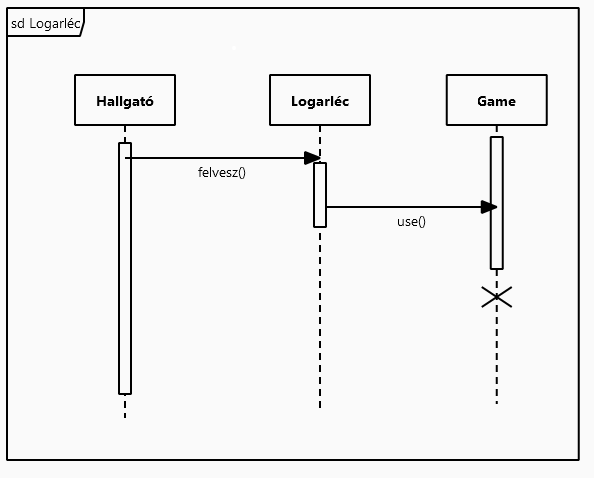
**3.4.4 FFP2-es maszk használata**



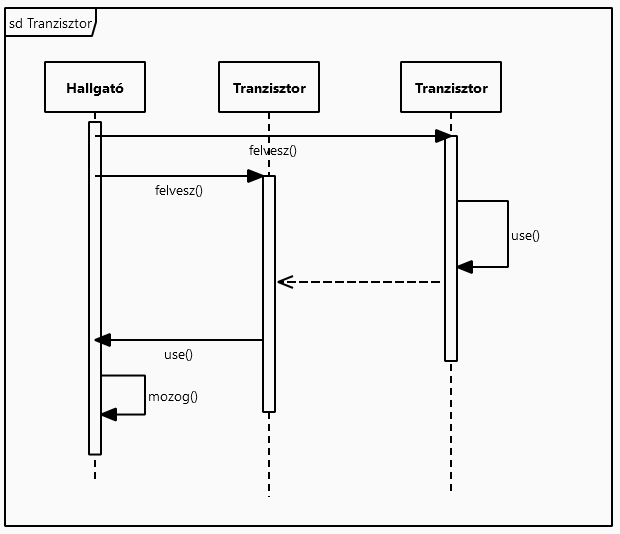
**3.4.5 TVSZ használata**



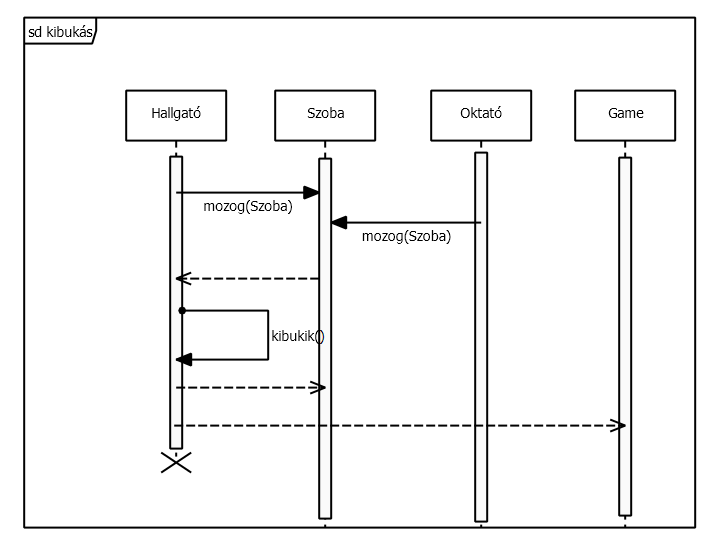
**3.4.6 Logarléc**



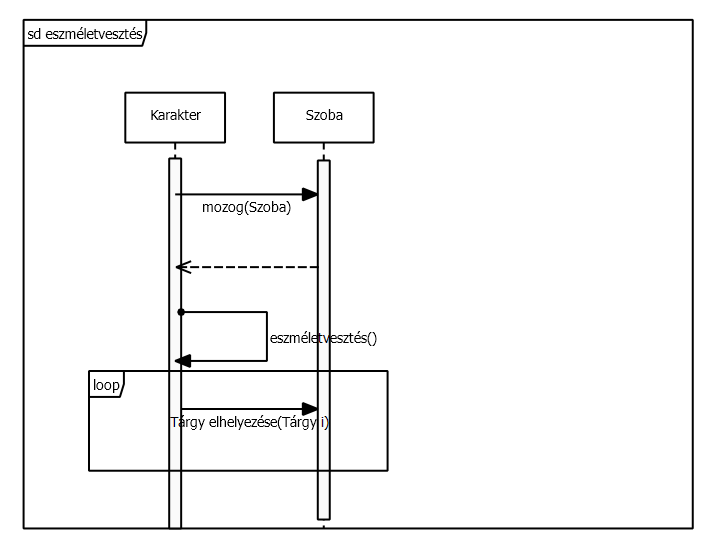
**3.4.7 Tranzisztor használata**



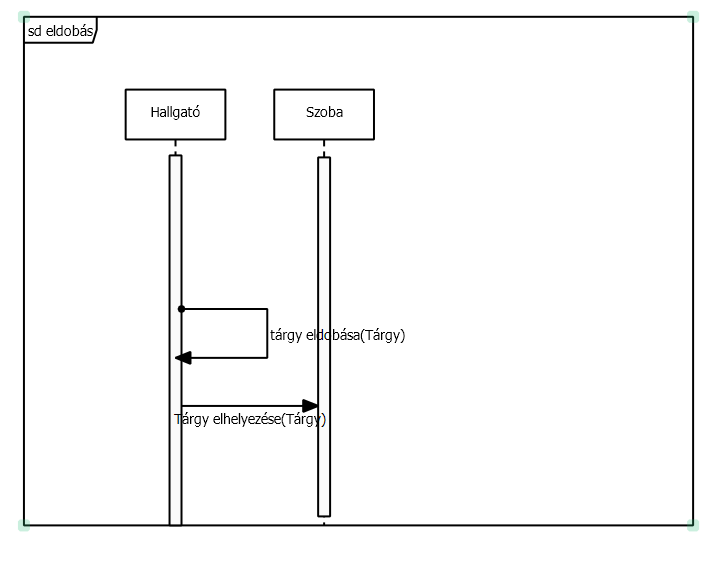
**3.4.8 Hallgató kibukása**



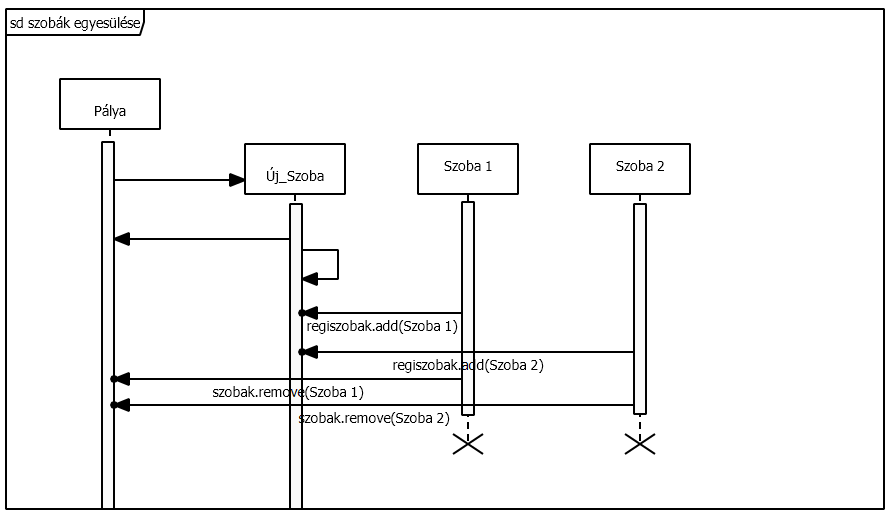
**3.4.9 Karakterek eszméletvesztése**

****

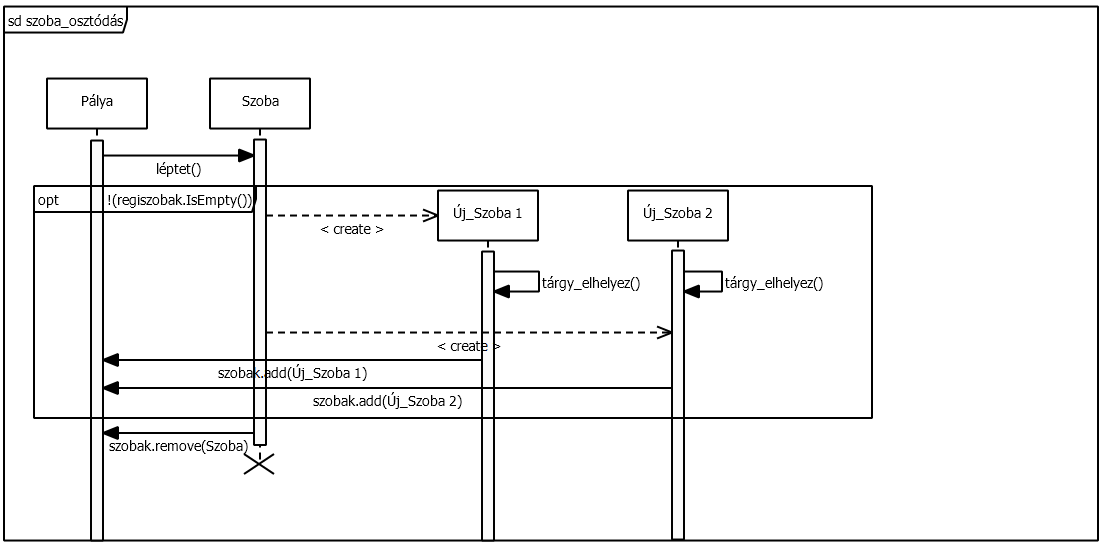
**3.4.10 Eldobás**

****

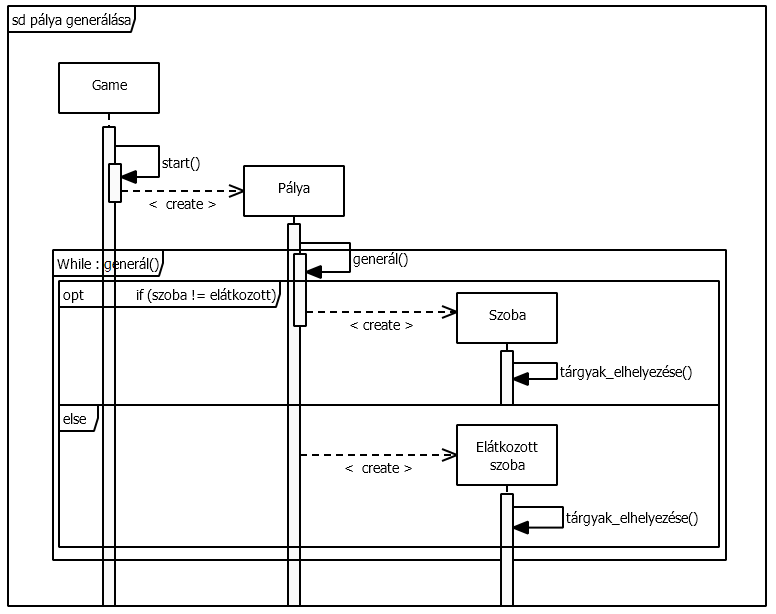
**3.4.11 Szobák egyesülése**



**3.4.12 Szobák osztódása**

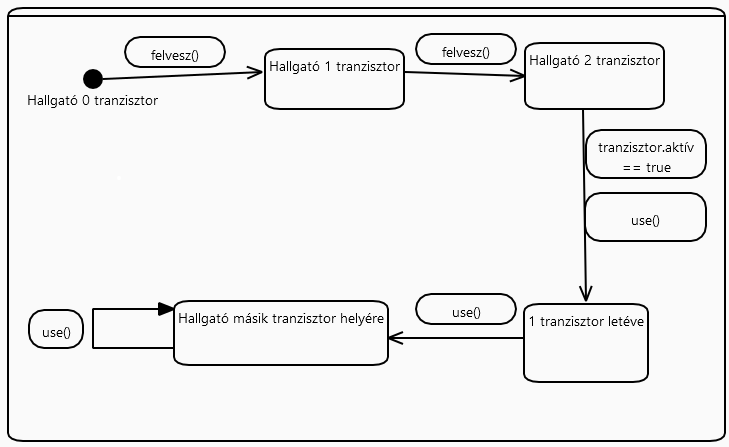


**3.4.13 Pálya generálása**



## **3.5** **State-chartok**

# **3.5.1 Tranzisztor használata**



## **3.6** **Napló**

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2024.02.28. 20:15 | 2 óra | Csapat | Értekezlet.Döntés: Osztálydiagram elkezdése. |
| 2024.02.29. 20:15 | 2 óra | Csapat | Értekezlet. Döntés:  Osztálydiagram befejezése, feladatok kiosztása. |
| 2024.03.01.9:00 | 2 óra | Czotter | Szekvencia diagramok, állapotgép készítése |
| 2024.03.02.14:00 | 2 óra | Schulcz | Szekvencia diagramok |
| 2024.03.03. 18:00 | 2 óra | Németh | Szekvencia diagramok |
| 2024.03.03. 20:00 | 2 óra | Csapat | Értekezlet. Döntések: Heti feladatok megbeszélése és lezárása/javítása. |
| 2024.03.03. 19:00 | 1,5 óra | Tarsoly | Objektum-katalógus és osztályok leírása |
| 2024.03.04. 11:00 | 2 óra | Hermann, Tarsoly | Objektum-katalógus és osztályok leírása, dokumentum megformázása, végső ellenőrzés. |

# 

# 