

2. Analízis modell I.

63 – Lehozook

Konzulens:
Szabó András

Csapattagok

Osztrogonác Dóra	UHI5AX	osztrogonaczdora@gmail.com
Kiss Tamás	C88IEG	kisstomi03@gmail.com
Halász Adrián	CZT1LC	virtualriot09@gmail.com
Schmidt Gergő András	S2Q2D4	s.gergo.az@gmail.com
Gáspár András	I6BST3	gaspar.andras.03@gmail.com

2024.03.04.

3. Analízis modell kidolgozása

3.1 Objektum katalógus

3.1.1 Hallgató (Student)

Játékos által irányított objektum, melynek célja a logarléc megszerzése. Tárgyak felvételére, tárolására, használatára és eldobására képes. Szobákat jár be.

3.1.2 Oktató (Professor)

Szomszédos szobákat kutat hallgatók után, az ezekben található tárgyakat felemészti, kivéve a logarlécet és az összepárosított tranzisztorokat. Hallgatókkal való találkozáskor mindent megtesz, hogy elszívja a lelküket.

3.1.3 Szoba (Room)

A benne lévő objektumok (hallgatók, oktatók, tárgyak) számontartása, valamint a szomszédos szobákba való átjárhatóság biztosítása a felelőssége. A saját állapotát is nyomon követi, gázos szobák esetén megbénít minden oktatót és hallgatót (akinek nincs FFP2 maszk). Elátkozott állapot esetén az átjárhatóságot változtatja.

3.1.4 Logarléc (SlideRule)

A játék végét idézi elő a hallgatók javára. Hatása akkor érvényesül ha felveszi egy hallgató.

3.1.5 TVSZ denevér bőrre nyomtatott példánya (TVSZ)

Használatkor a szobában tartózkodó hallgatókat megvédi az oktatók támadásai elől. Háromszor használható tárgy, használatkor a szobában lévő oktatókat lebénítja.

3.1.6 Szent Sörös Pohár (StBeerCups)

Amíg tart a hatása, megvédi az oktatók ellen azt a hallgatót, aki megitta. Miután a hallgató használta a tárgyat 3 körön át megvédi őt

3.1.7 Nedves Táblatörlő Rongy (WetRag)

Használat után elhelyezi a tárgyat a szobában és a nedvessége erejéig (2 kör) lebénítja a szobában tartózkodó oktatókat.

3.1.8 Dobozolt Káposztás Camembert (Camembert)

Használatra elgázosítja a szobát.

3.1.9 FFP2 maszk (FFP2Mask)

Amíg el nem törik, megvédi azt a hallgatót a gáztól, aki viseli. Nem gázos szobákban nincs jelentősége. Három alkalommal védi meg a hallgatót a gáztól, utána eltörik.

3.1.10 Tranzisztor (Transistor)

A hallgatók hátizsákjában páronként összekapcsolódnak, mely párok két különböző szobában elhelyezve kétirányú átjárót biztosítanak egymáshoz.

3.1.11 Hátizsák (Backpack)

A hallgatók itt tudják eltárolni a tárgyaikat és elérni őket szükség esetén.

3.1.12 Ütemező (Scheduler)

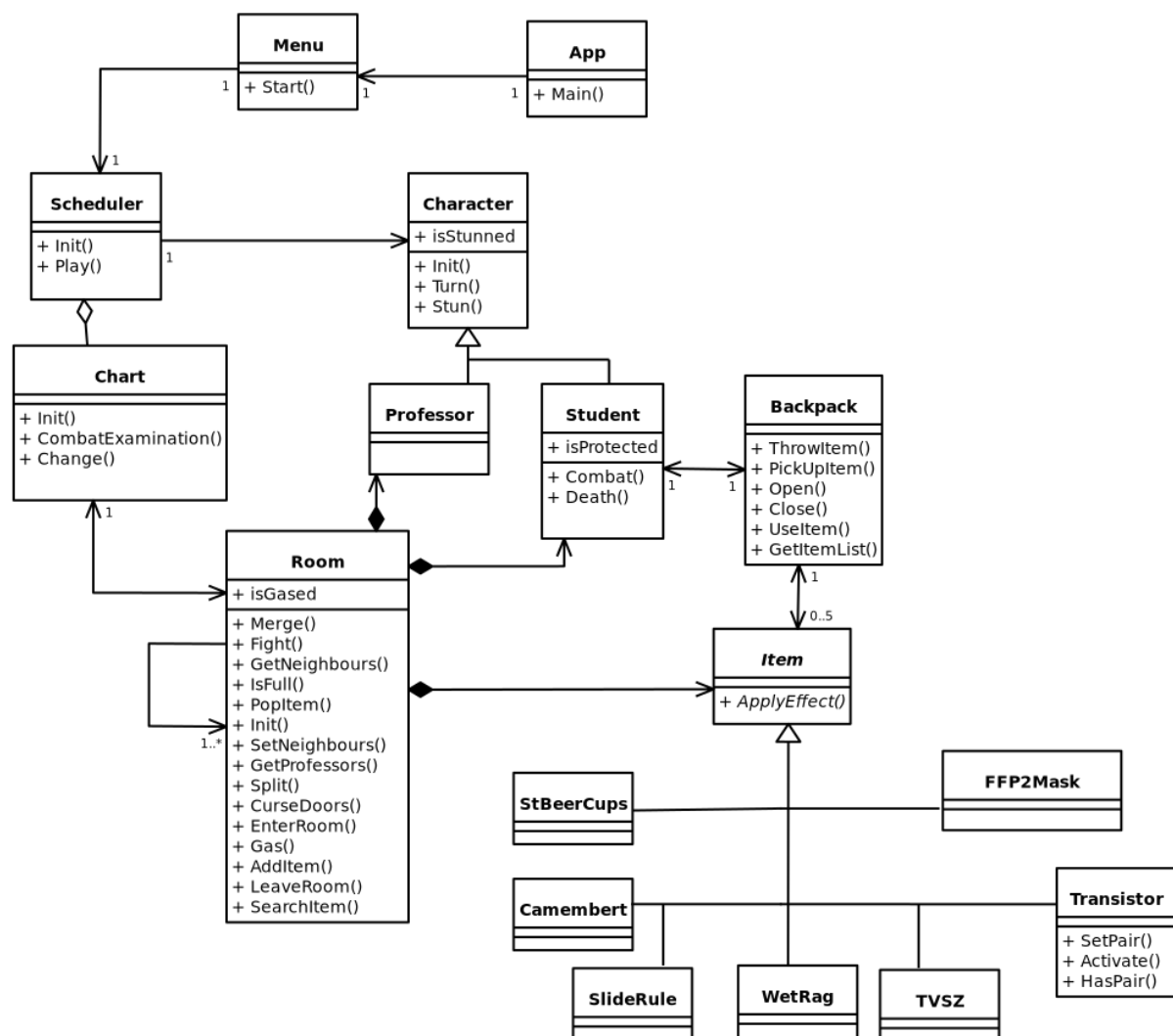
A játék során ez az objektum hívja meg a körben jövő következő objektum start metódusát.

3.1.13 Menü(Menu)

A felhasználók itt tudják beállítani a játék paramétereit(hallgatók és professzorok száma) és elindítani a játékot.

3.2 Statikus struktúra diagramok

cls Class Diagram



3.3 Osztályok leírása

3.3.1 App

- **Felelősség**

Az első osztály, ami elindul, a program valódi indítója.

- **Asszociációk**

- **1. asszociáció:** túloldali szereplő: Menu. Ő indítja el a Menu Start metódusát.

- **Metódusok**

Main(): Meghívja a Menu Start metódusát, amivel megnyílik a menü.

3.3.2 Backpack

- **Felelősség**

A Backpack osztály az Itemek tárolására való (maximum 5), minden hallgató rendelkezik egy hátizsákkal.

- **Asszociációk**

- **1.Asszociáció:** túloldali szereplő: Student. Minden hallgatónak van egy hátizsákja.
- **2.Asszociáció:** túloldali szereplő: Item. A hátizsák tartalmazza a tárgyakat.

- **Metódusok**

- **ThrowItem():** Kidobja a szobába a tárgyat, amiben a Hallgató éppen tartózkodik.
- **PickUpItem():** A szobában lévő tárgyak közül berak a hátizsákba.
- **Open():** Megnyitja a hátizsákot.
- **Close():** Bezárja a hátizsákot.
- **UseItem():** Használ egy tárgyat
- **GetItemList():** Visszaadja a hátizsák tartalmát

3.3.3 Camembert

- **Felelősség**

A szobákat el lehet gázosítani a tárgy használatával.

- **Ősosztályok**

Item

- **Metódusok**

- **ApplyEffect():** Elgázosítja a szobát amiben használva van

3.3.4 Character

- **Felelősség**

A szereplők akciói a játékban.

- **Attribútumok**
 - **1. attribútum:** boolean: isStunned: Ez jelzi, ha az adott karakter el van kábítva.
- **Metódusok**
 - **Turn():** A karakter ezzel a metódussal végzi el az akcióit, amikor rákerül a sor.
 - **Init():** Inicializáláskor az alapbeállítások beállításáért felelős
 - **Stun():** Beállítja az isStunned változót és az elkábítás időtartamát.

3.3.5 Chart

- **Felelősség**

A Chart osztály tartalmazza a pályán lévő szobákat és azoknak az elhelyezkedését. A Scheduler ezen keresztül érheti el őket.

- **Asszociációk**
 - **1.Asszociáció:** túloldali szereplő:Room. A Chart osztály tartalmaz több Room objektumot is, amelyek a Chart osztály nélkül nem létezhetnek.
 - **2.Asszociáció:** túloldali szereplő: Scheduler. Lásd: Scheduler.
- **Metódusok**
 - **Init():** A metódus létrehozza a pályát és a szükséges objektumokkal feltölti a szobákat(Item,Student,Professor)
 - **Change():** Végigmegy a szobákon, és véletlenszerűen meghívja a Split, vagy Merge metódust, amely szoba el van átkozva ott pedig a CurseDoors-t.
 - **CombatExamination():** A metódus végigmegy az összes szobán és meghívja a megfelelő helyeken a Fight() metódusokat.

3.3.6 FFP2Mask

- **Felelősség**

Megvédi a hallgatót a gázos szobáktól, ameddig a hátizsákjukban van.

- **Össztályok**

Item

- **Metódusok**
 - **ApplyEffect():** immunissá teszi a hallgatót a gázra.

3.3.7 Item

- **Felelősség**

A különböző tárgyak realizációja.

- **Asszociációk**
 - **1. asszociáció:** túloldali szereplő: Backpack. Amennyiben egy tárgyat már felvett egy Student, akkor a hátizsákjában tárolja.

- **2. asszociáció:** túloldali szereplő: Room. Ha még nem vette fel senki az adott tárgyat, akkor a szobában helyezkedik el.
- **Metódusok**
 - **ApplyEffect():** Az adott tárgy kifejti a hatását.

3.3.8 Menu

- **Felelősség**

Ez valósítja meg a játékot indító menüt, ahol ki lehet választani a kezdési paramétereket(játékos szám, tanárok száma, körök száma).
- **Asszociációk**
 - **1. asszociáció:** túloldali szereplő: Scheduler. Példányosítva van az osztályban.
- **Metódusok**
 - **Start():** Megnyitja a menüt, amiben a felhasználó kiválaszthatja a Hallgatók(Student) számát és a Tanárok(Professor) számát és a játék hosszát. Ezekkel a paraméterekkel létrehozza a Scheduler-t és meghívja a Start metódusát.

3.3.9 Professor

- **Felelősség**

Az oktató mozgása a szobák között, a tárgyak bekebelezése és a hallgatók lelkének elszívása a felelőssége.
- **Ősosztályok**

Character
- **Asszociációk**
 - **1. asszociáció:** túloldali szereplő: Room. Egyrészt a Professor tárolja hogy éppen melyik szobában helyezkedik el, másrészt, a szoba is tárolja, hogy éppen mely oktatók tartózkodnak benne.
- **Metódusok**
 - **Turn():** A Tanár véletlenszerűen átlép egy szomszédos szobába és a szobában lévő tárgyakat megeszi, ha vannak.

3.3.10 Room

- **Felelősség**

A Room osztály felelős a szobák állapotáért. Ellenőrzi, hogy a szoba átkozott/gázos, illetve azt, hogy egy kör végén alakult-e ki olyan szituáció, hogy oktató(k)(Professor) és diák(ok)(Student) együtt maradtak az adott szobában (ebben az esetben ugyanis harcra kerül sor). A szobák összeolvadásáért (merge) és szétválásáért(split) is ő felel.

- **Asszociációk**
 - **1. asszociáció:** túloldali szereplő: Professor. A szoba egy kör végén megnézi, hogy vannak-e benne professzorok és hallgatók egyszerre. A jelen lévő professzorokat tárolja.
 - **2. asszociáció:** túloldali szereplő: Student. A szoba egy kör végén megnézi, hogy vannak-e benne professzorok és hallgatók egyszerre. A jelen lévő hallgatókat
 - **3. asszociáció:** túloldali szereplő: Item. A szoba nyilvántartja, hogy milyen tárgyak vehetőek fel benne.
 - **4. asszociáció:** túloldali szereplő: Room. A szoba nyilvántartja, hogy milyen más szobákkal szomszédos.
 - **5. asszociáció:** túloldali szereplő: Chart. Lásd: Chart.
- **Attribútumok**
 - **1. attribútum:** boolean: isGased. Azt jelzi, hogy egy szoba gázos-e vagy nem. Ezt például a Dobozolt Káposztás Camembert kinyitásával lehet előidézni.
- **Metódusok**
 - **Fight():** rendezzi a harcot a szobában lévő résztvevőkkel, ha ilyen helyzet áll fenn.
 - **GetNeighbors():** Visszaadja a szobának a szomszédait.
 - **SetNeighbors():** Beállítja a szoba szomszédait.
 - **GetProfessors():** Visszaadja a szobában tartózkodó professzorok listáját.
 - **IsFull():** Visszaadja, hogy a szoba tele van-e.
 - **SearchItem():** visszaadja a szobában felvehető itemeket.
 - **PopItem():** A szobában lévő Itemek közül visszaadja a kiválasztottat és a saját listájából törli azt.
 - **AddItem():** A szoba Item listájához ad egy Itemet.
 - **Split():** szétválasztja a szobát és visszaadja az új két szobát.
 - **Merge():** összeolvasztja a szobát az egyik szomszédjával és az új szobát visszaadja.
 - **EnterRoom():** Belerakja a listájába a karaktert, aki a szobába lép.
 - **LeaveRoom():** Kiveszi a szobából az adott karaktert, amikor kilép a szobából.
 - **Gas():** Elkábítja a szobában lévő Karaktereket.
 - **CurseDoors():** Eltüntetheti az elátkozott szobák ajtajait.
 - **Init():** Inicializálja a szoba alapbeállításait.

3.3.11 Scheduler

- **Felelősség**

Ez az osztály valósítja meg a körök lebonyolítását és számon tartja a játékból hátralévő körök számát. A Scheduler kéri fel a külső szereplőket (Chart, Student, Professor), hogy hajtsák végre a lépésüket az adott körben.

- **Asszociációk**

- **1. asszociáció:** túloldali szereplő: Professor. Tartalmaz egy listát, ami a Tanárokat tartja számon.
- **2. asszociáció:** túloldali szereplő: Student. Tartalmaz egy listát, ami a Diákokat tartja számon.
- **3. asszociáció:** túloldali szereplő: Chart. Tartalmaz egy pálya objektumot.

- **Metódusok**

- **Play():** Meghatározott sorrendben meghívja az adott körben résztvevő objektumokat(Char→ Professor→ Student).
- **Init():** létrehozza a Chartot és a két Character listát

3.3.12 SlideRule

- **Felelősség**

A játék megnyeréséhez ezt a tárgyat kell birtokolni.

- **Ősosztályok**

Item

- **Metódusok**

- **ApplyEffect():** amikor a hallgató felveszi, véget ér a játék. Nyertek a hallgatók.

3.3.13 StBeerCups

- **Felelősség**

A Hallgatónak, aki ezt megissza adott körön át immunitást biztosít a támadások ellen.

- **Ősosztályok**

Item.

- **Metódusok**

- **ApplyEffect():** A hallgató isProtected értékét igazra állítja.

3.3.14 Student

- **Felelősség**

A játékos(ok) figuráinak kezelése. A figura mozgása, a hátizsák módosítása, és a harc kivitelezése is az osztály felelőssége

- **Ősosztályok**

Character

- **Asszociációk**

- **1. asszociáció:** túloldali szereplő: Room. Egyrészt a Student tárolja hogy éppen melyik szobában helyezkedik el, másrészt, a szoba is tárolja, hogy éppen mely tanulók tartózkodnak benne.

- **Attribútumok**

- **boolean: isProtected** - Azt mutatja meg, hogy a Tanárok támadásai ellen immunis-e.

- **Metódusok**

- **Combat()** - Ha egy szobába kerül egy Student egy oktatóval, akkor harcolni kezdenek, ezt végzi ez a metódus.
- **Death()**: Kidobja a szobába az összes tárgyat és törli magát belőle.

3.3.15 Transistor

- **Felelősség**

A Tranzisztorokat össze lehet kapcsolni páronként, amiket azután átjáróként lehet használni a szabályok szerint.

- **Ősosztályok**

Item

- **Metódusok**

- **ApplyEffect()**: A tárgyat bekapcsolja és a földre rakja a Hallgatóval azonos szobában.
- **Activate()**: A hallgatót a tranzisztor párjához teleportálja
- **SetPair()**: Beállítja a tranzisztor párját
- **HasPair()**: Lekérdezi, hogy van-e párja a tranzisztornak

3.3.16 TVSZ

- **Felelősség**

A Hallgató háromszor tudja ezt a tárgyat használni arra, hogy a szobában lévő összes Tanárt elkábítsa.

- **Ősosztályok**

Item

- **Metódusok**

- **ApplyEffect()**: A hallgatót védetté teszi 3 támadás erejéig

3.3.17 WetRag

- **Felelősség**

Oktatók elkábítása

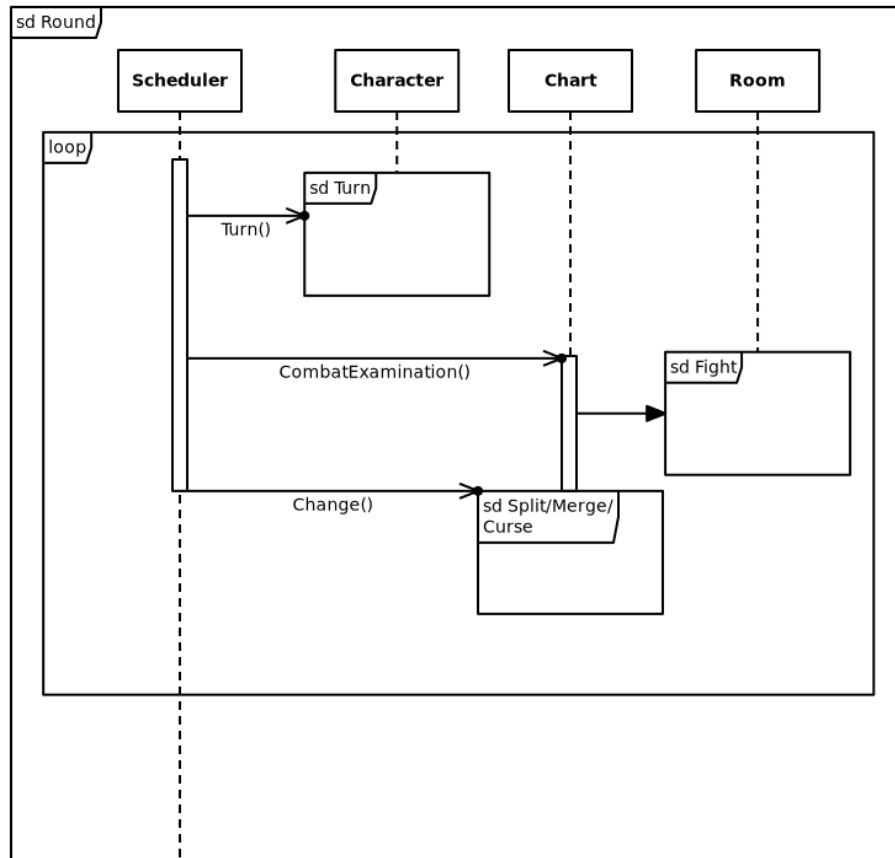
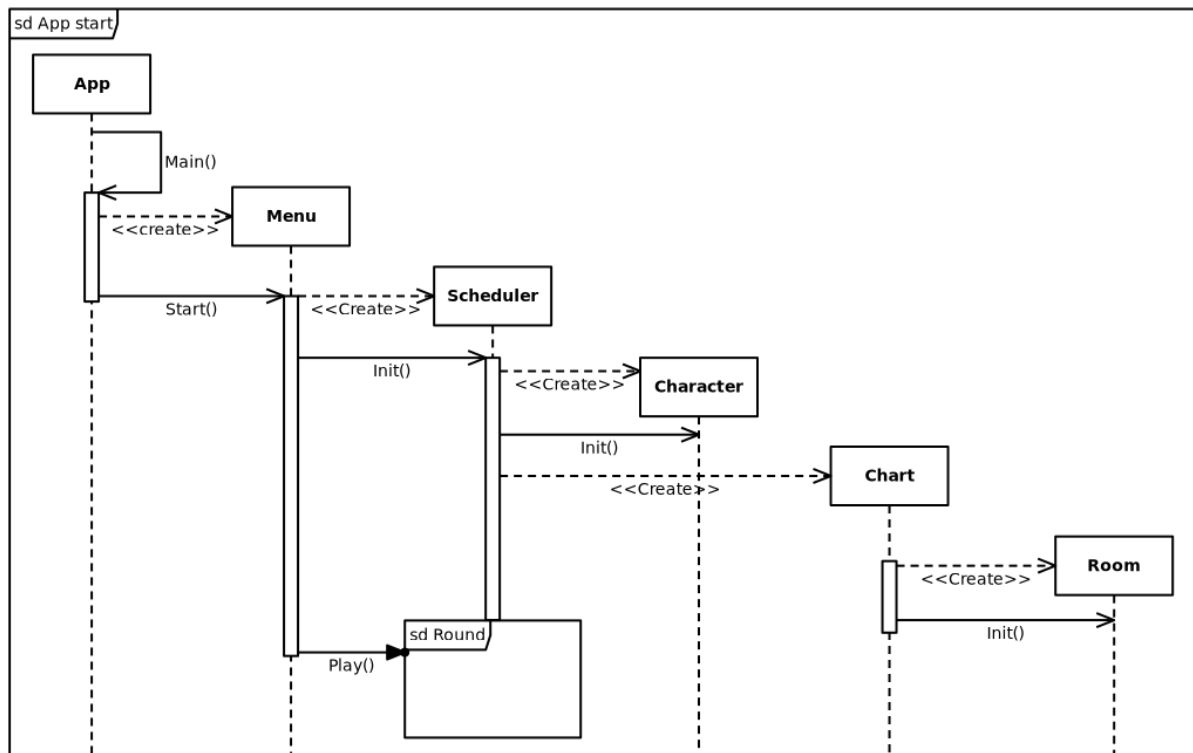
- **Ősosztályok**

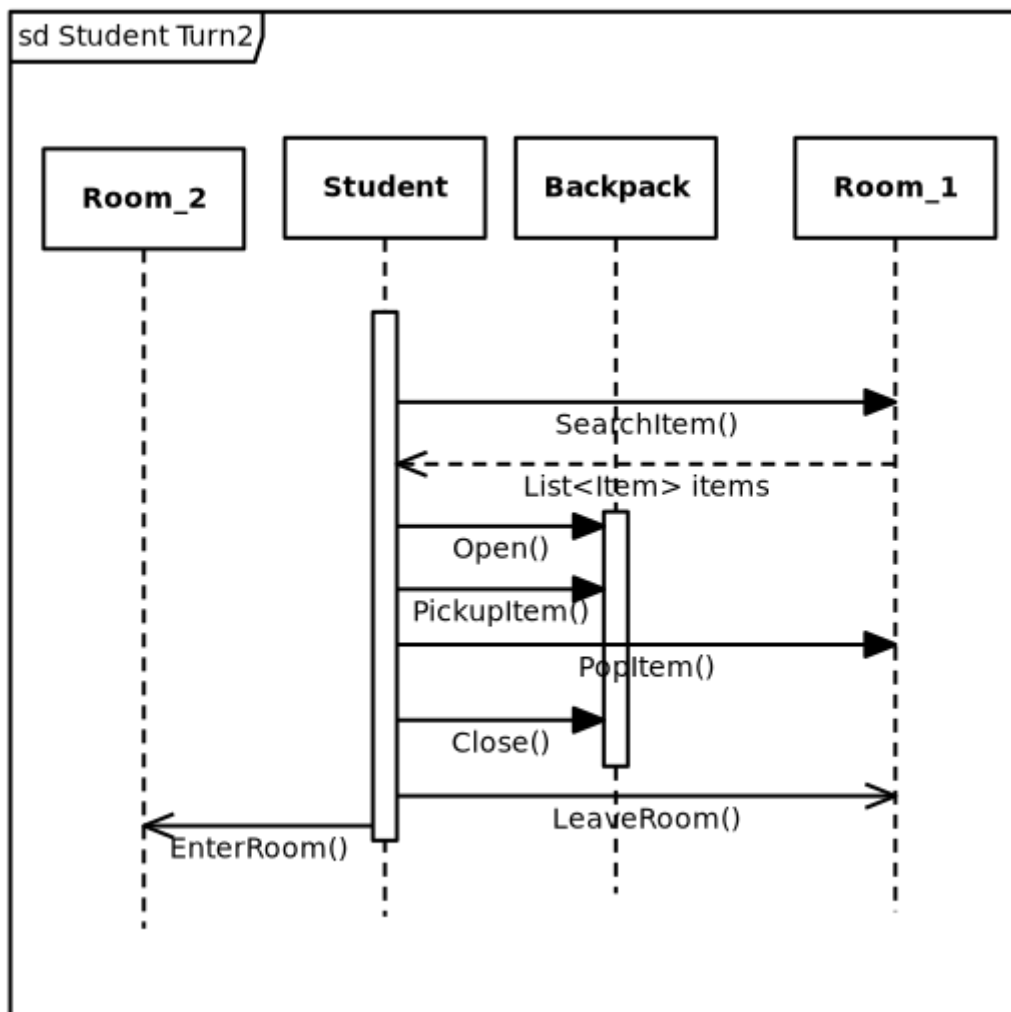
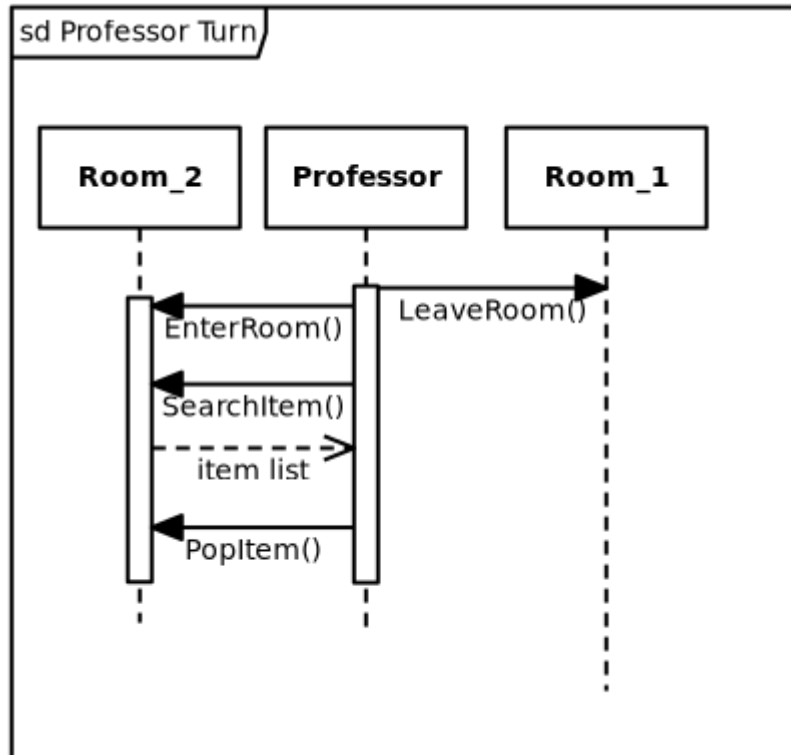
Item

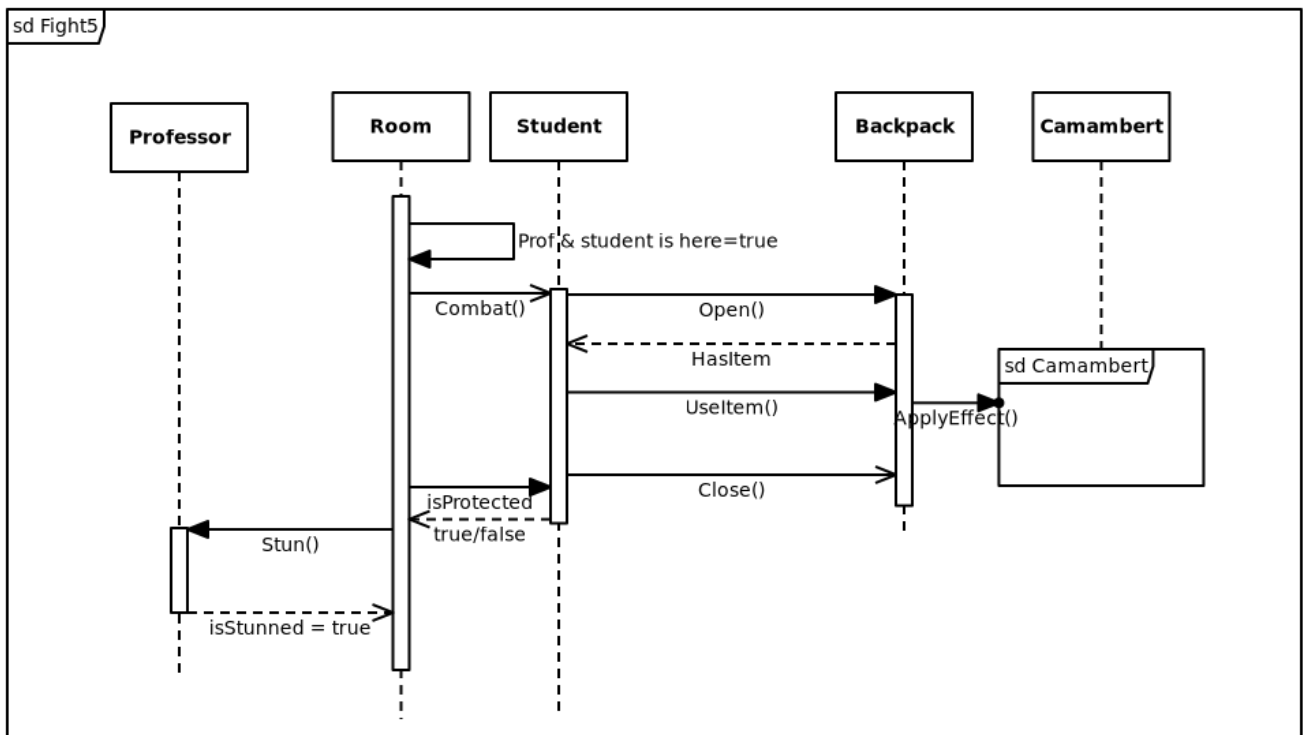
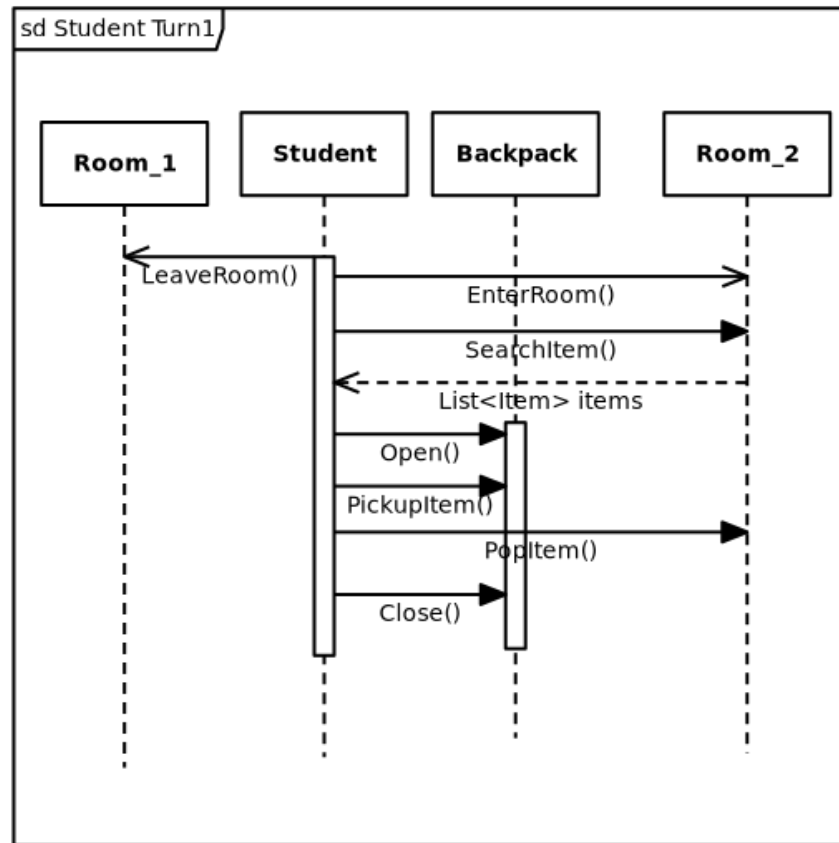
- **Metódusok**

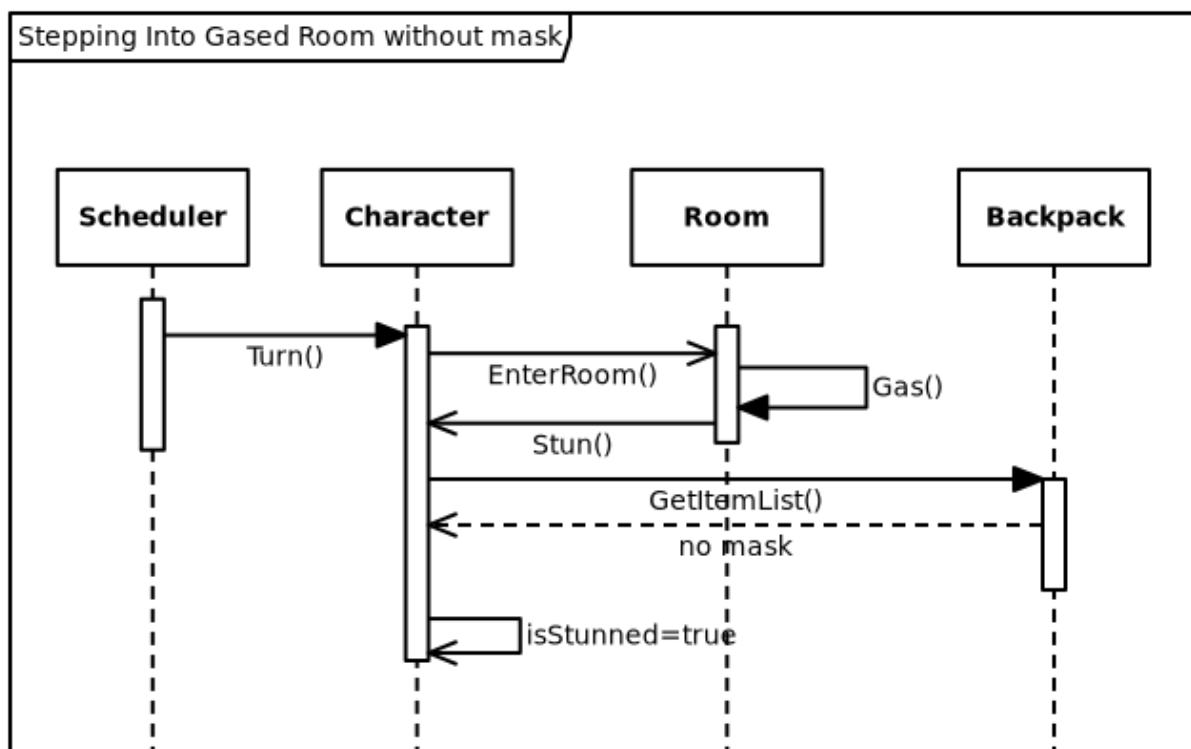
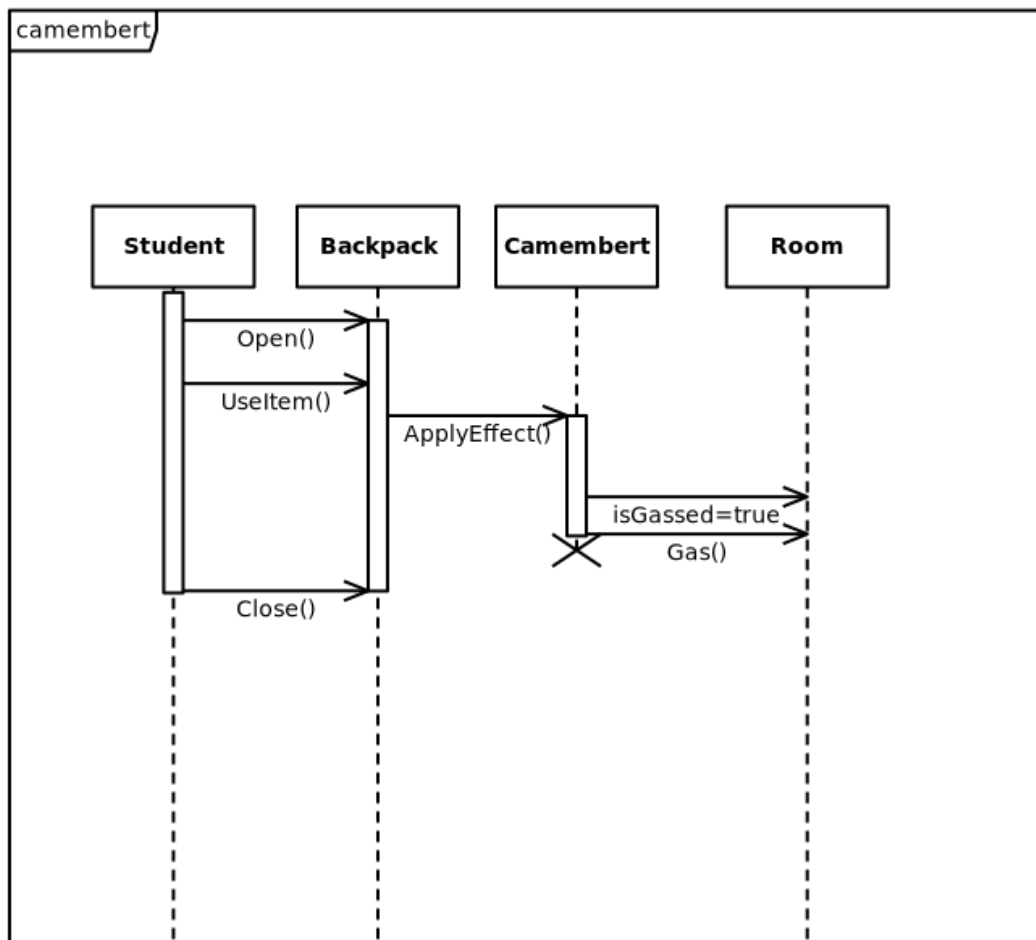
- **ApplyEffect()**: Elkábítja a szobában tartózkodó oktatókat a nedvességével arányos ideig

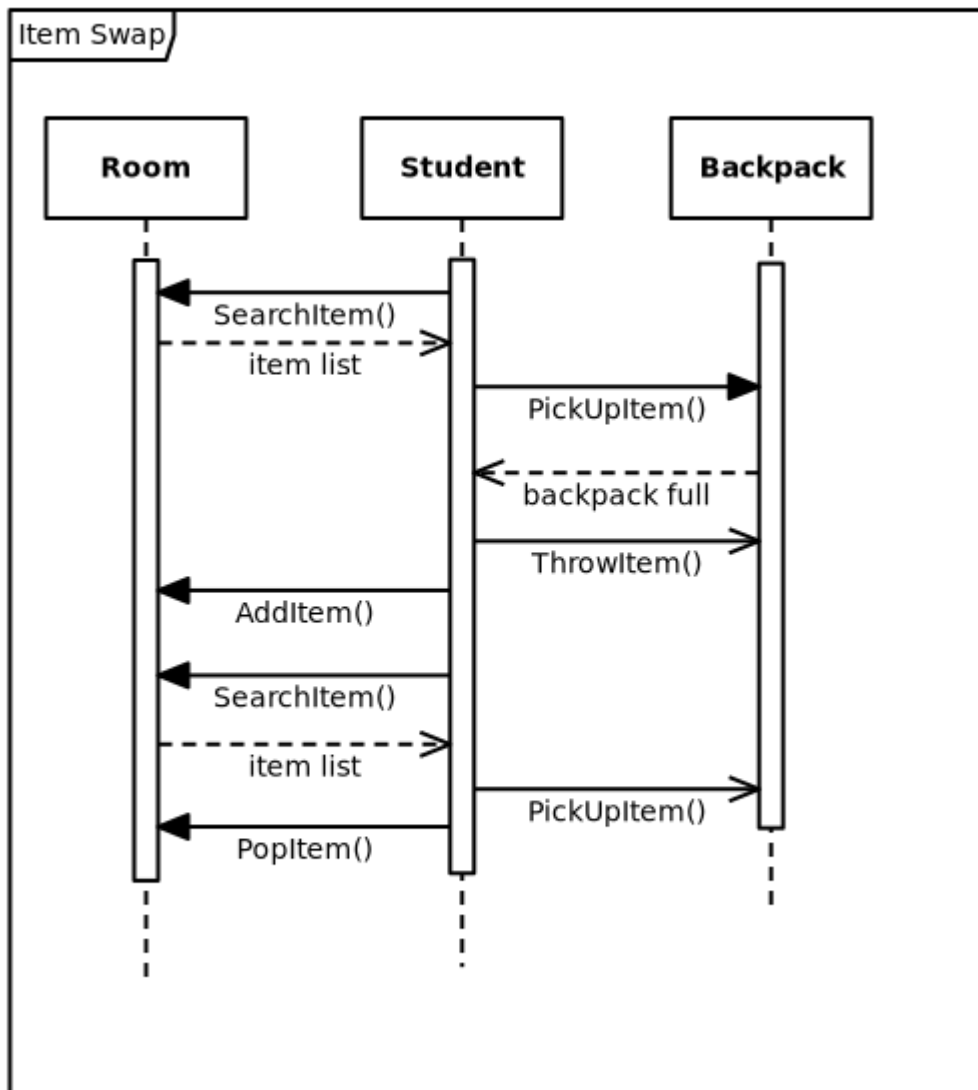
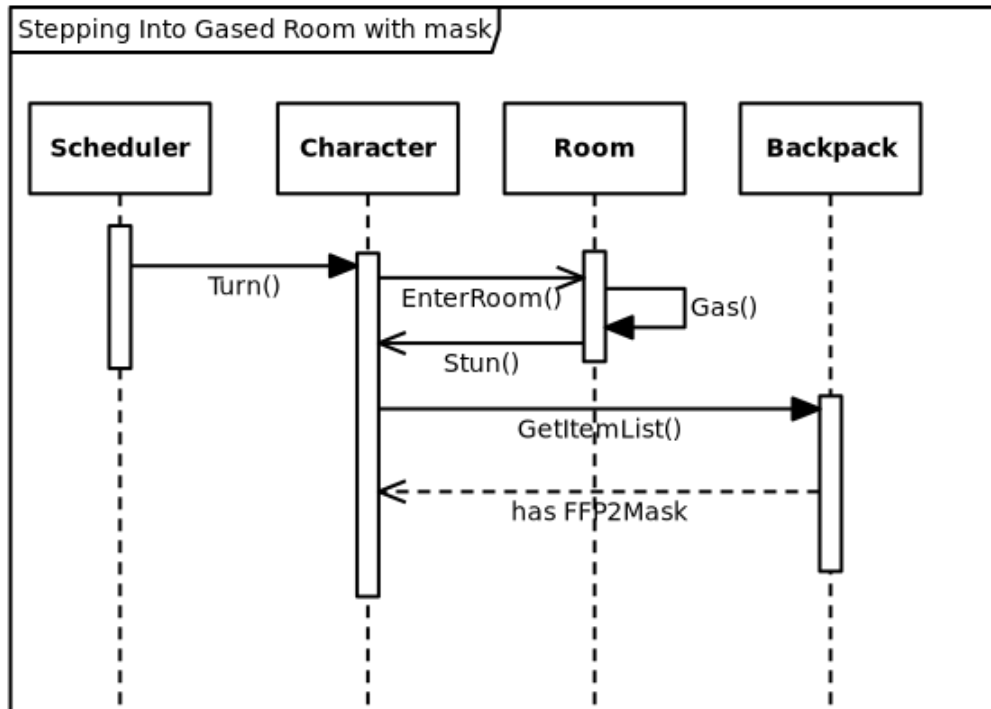
3.4 Szekvencia diagramok

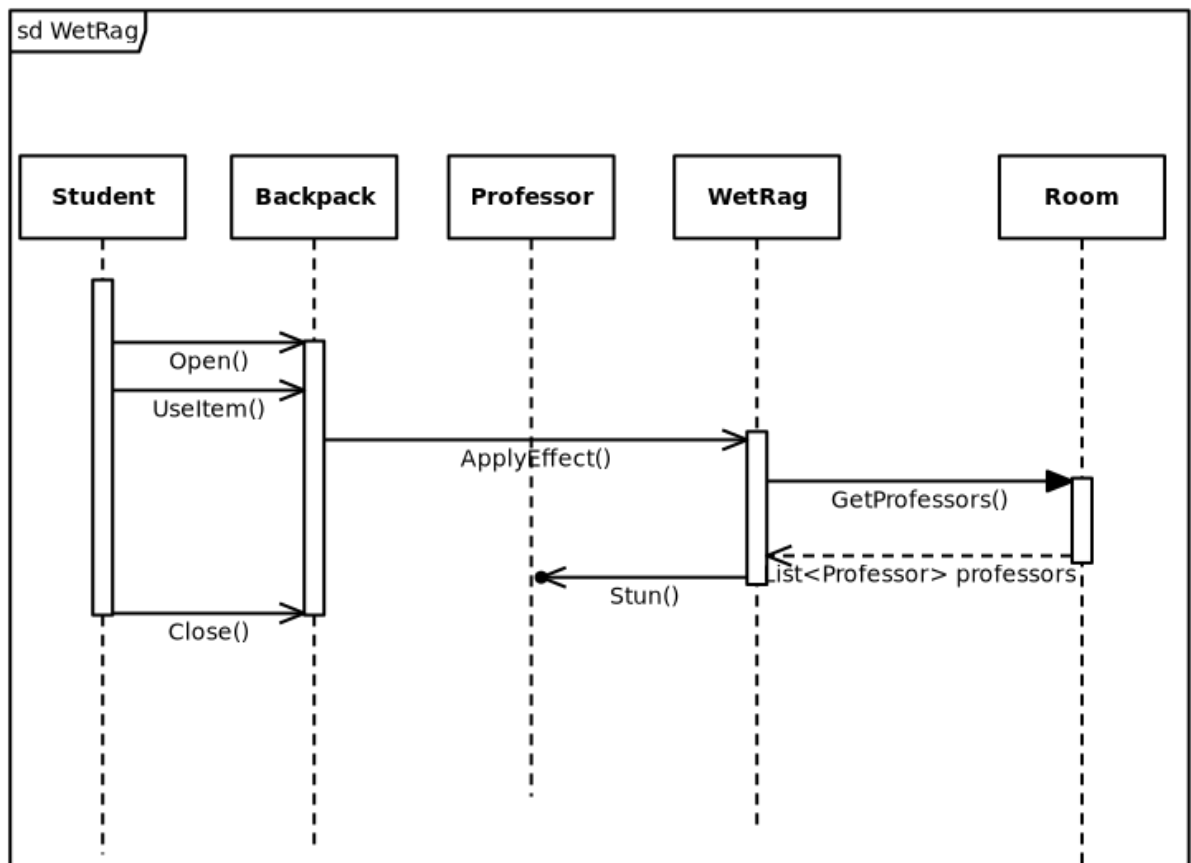
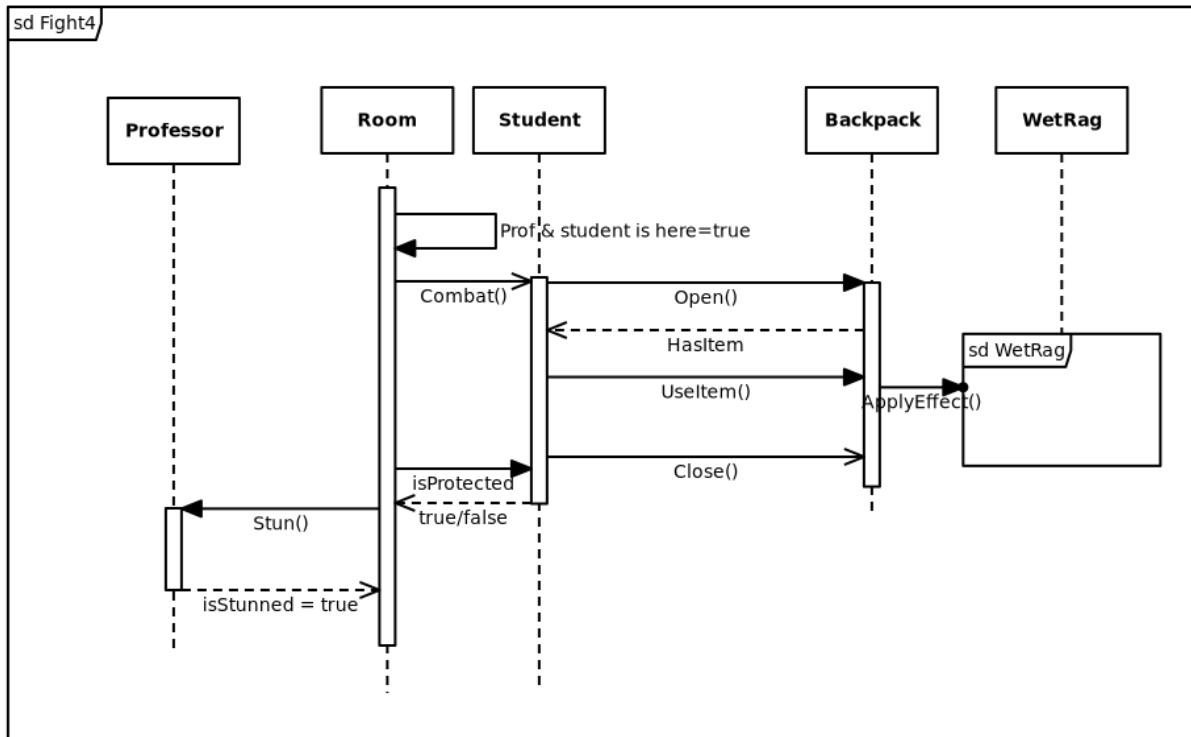


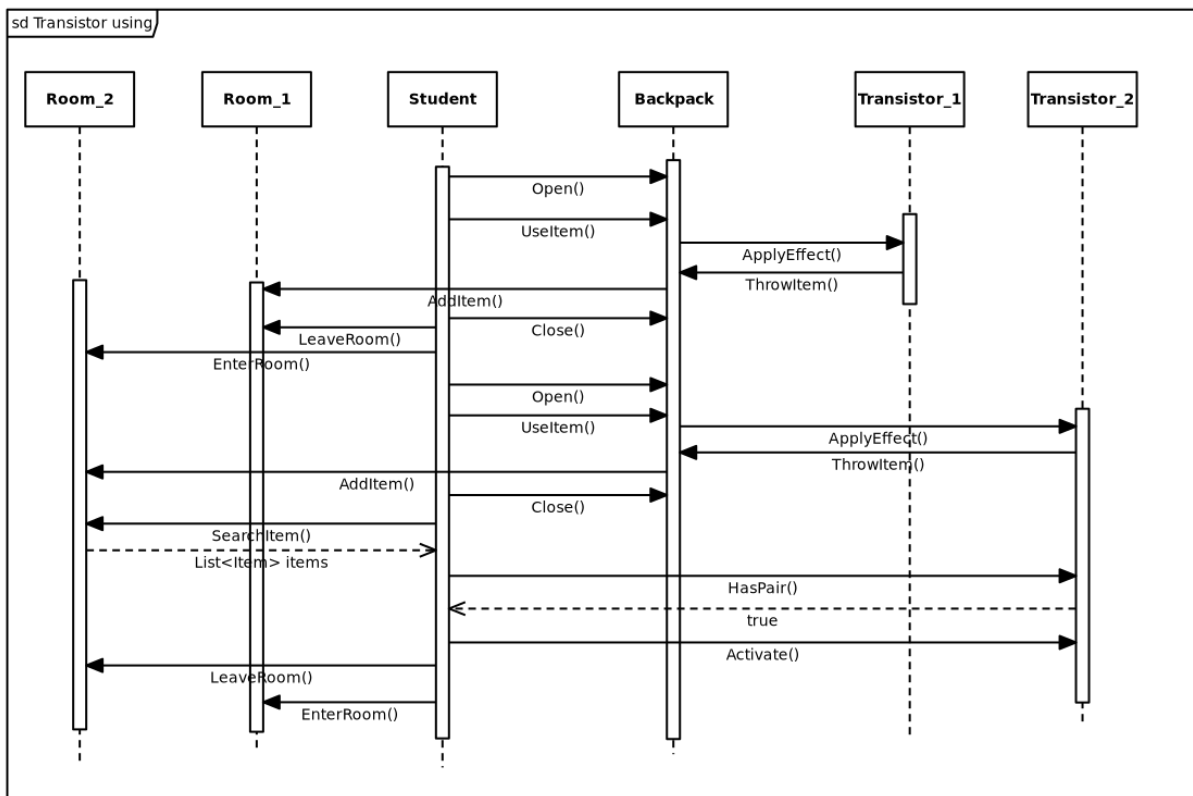
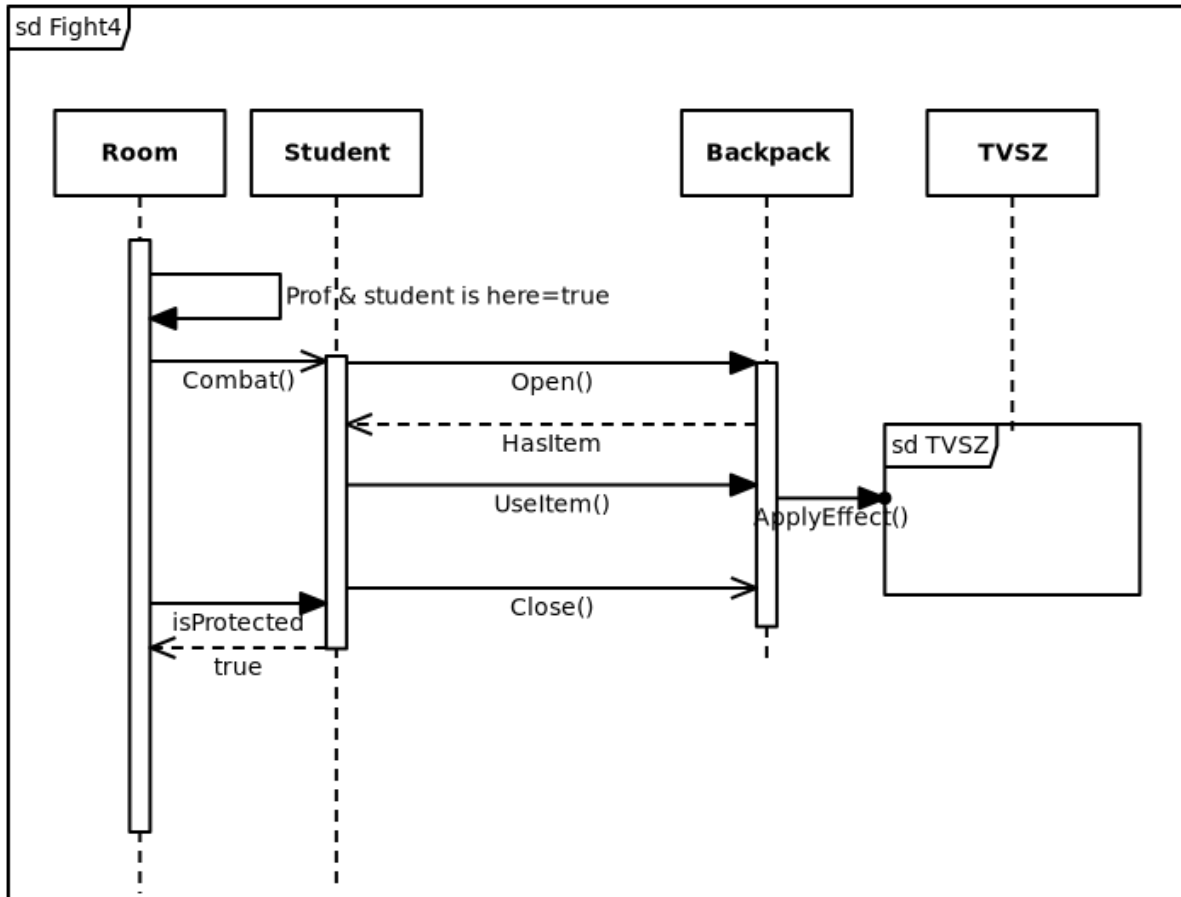


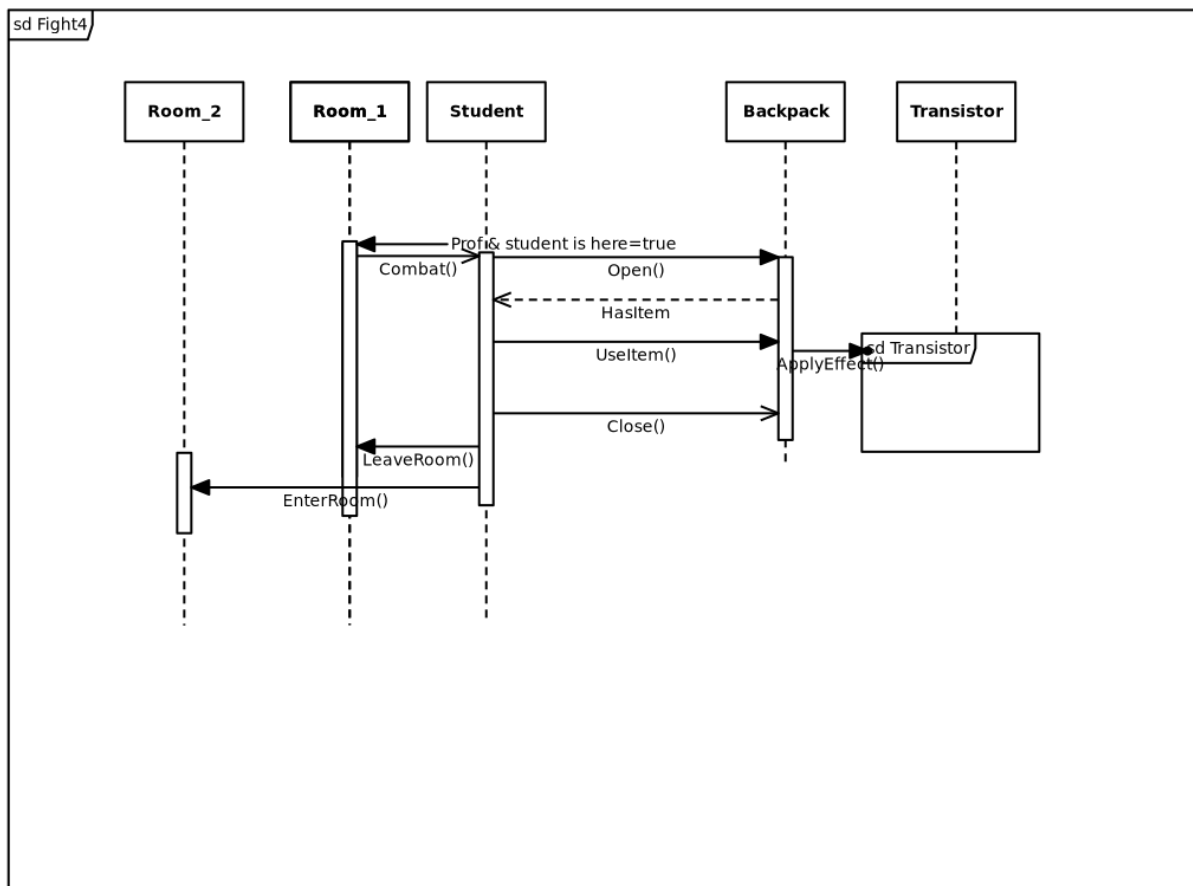
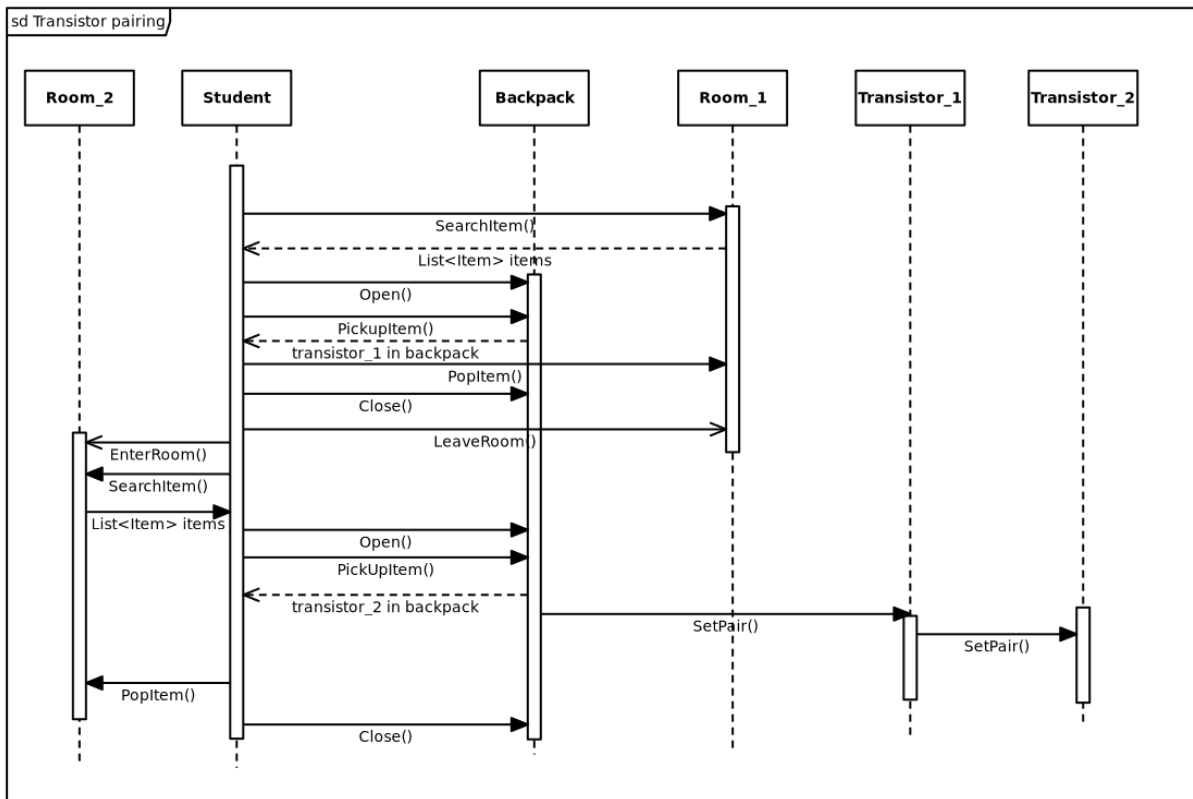


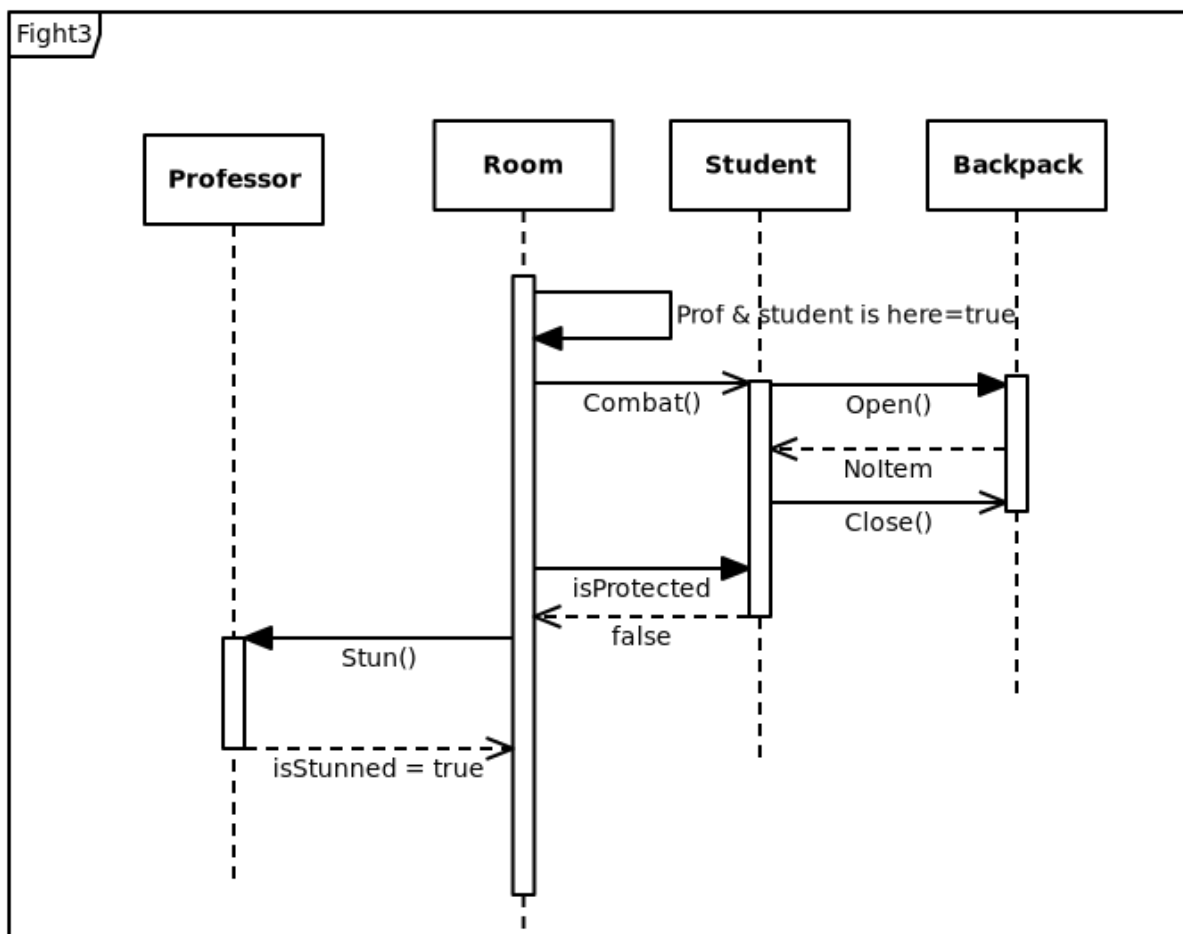
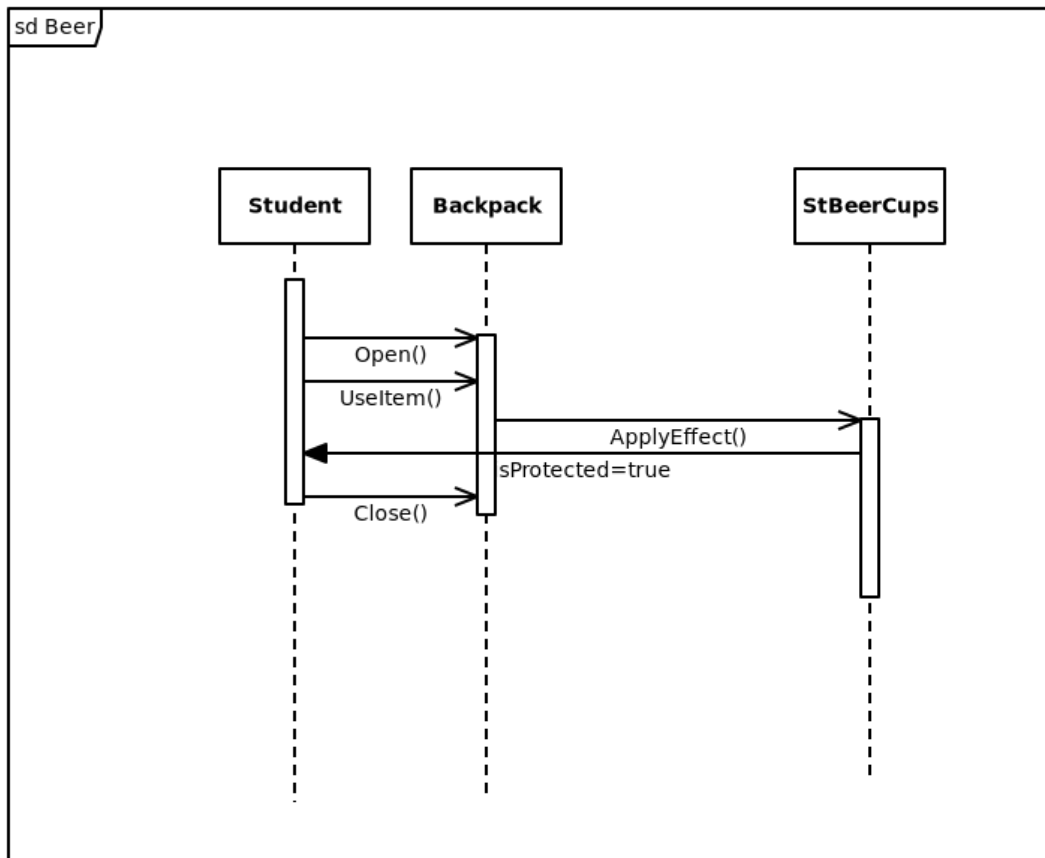


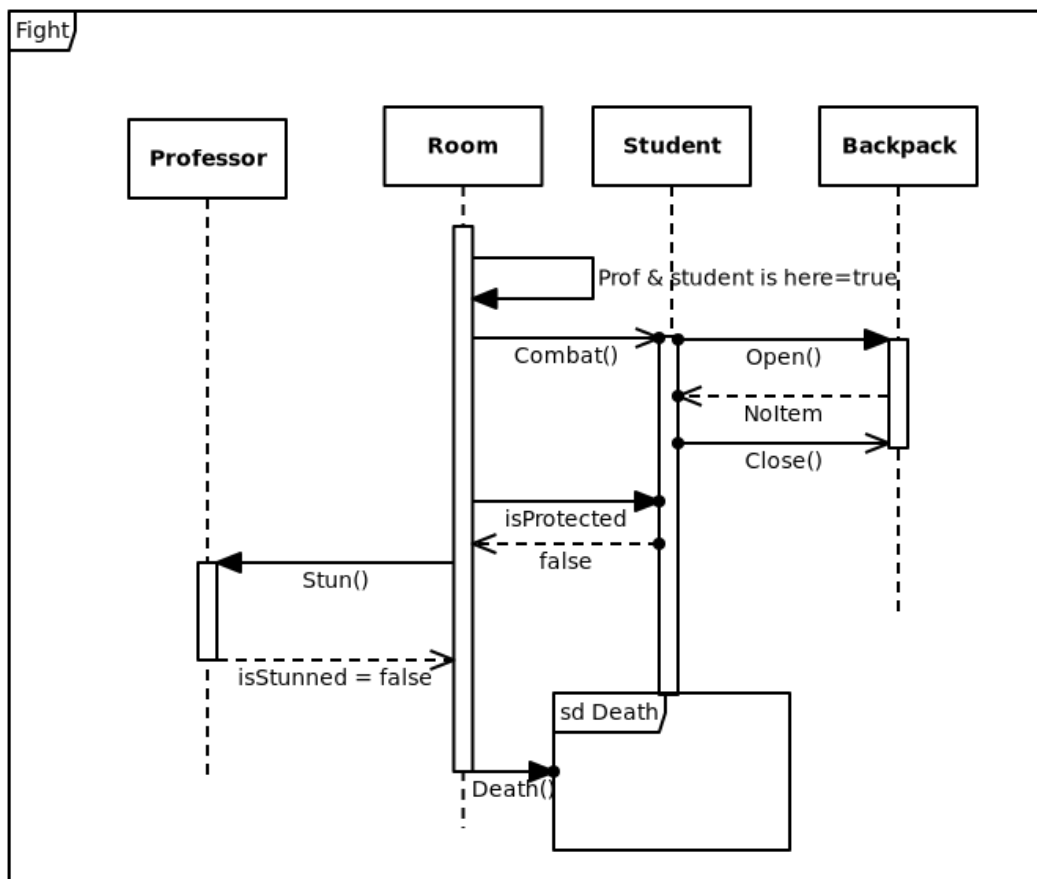
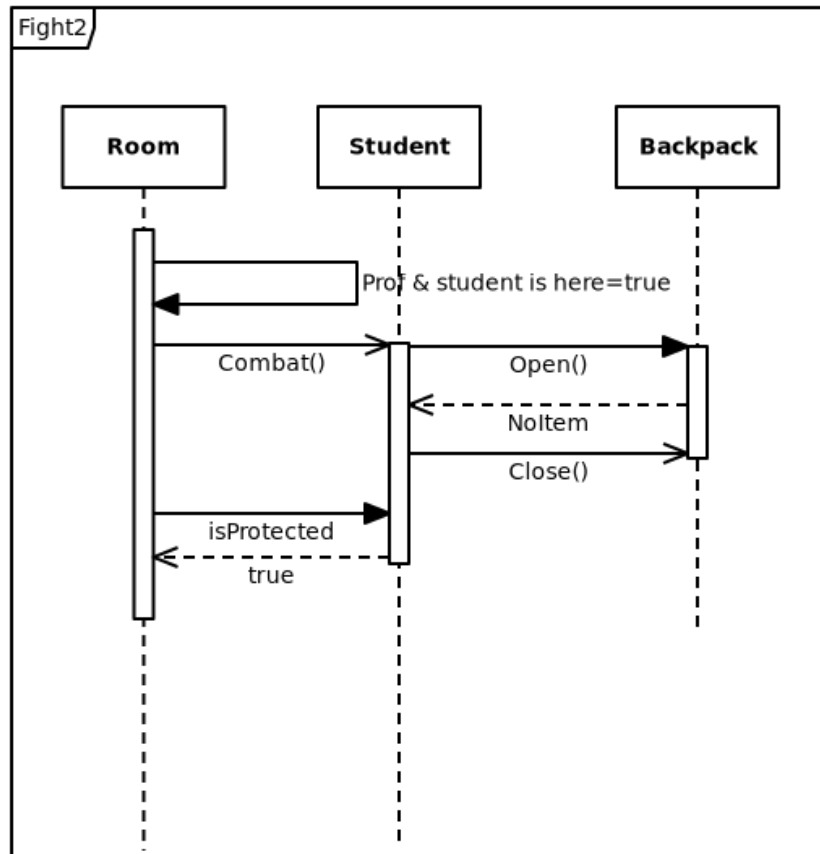


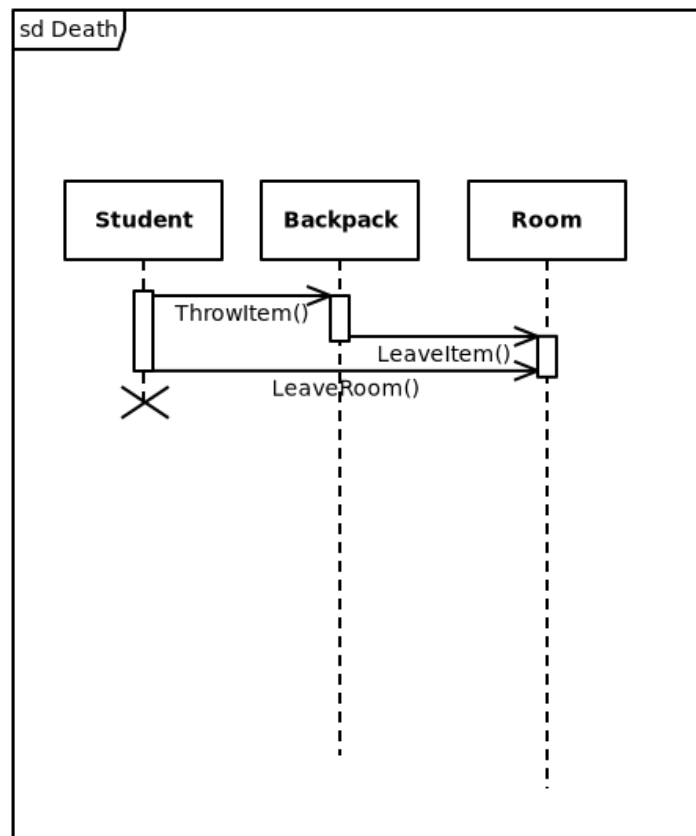
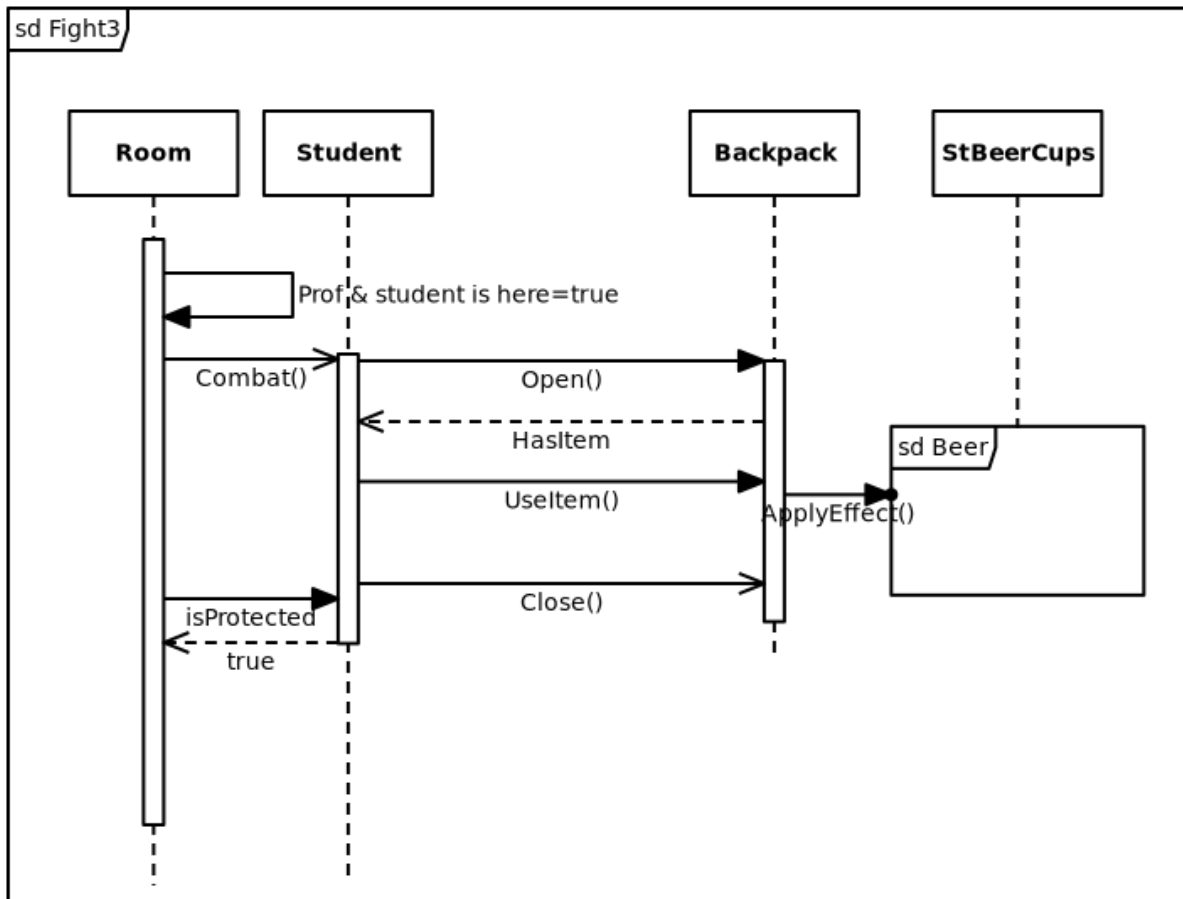


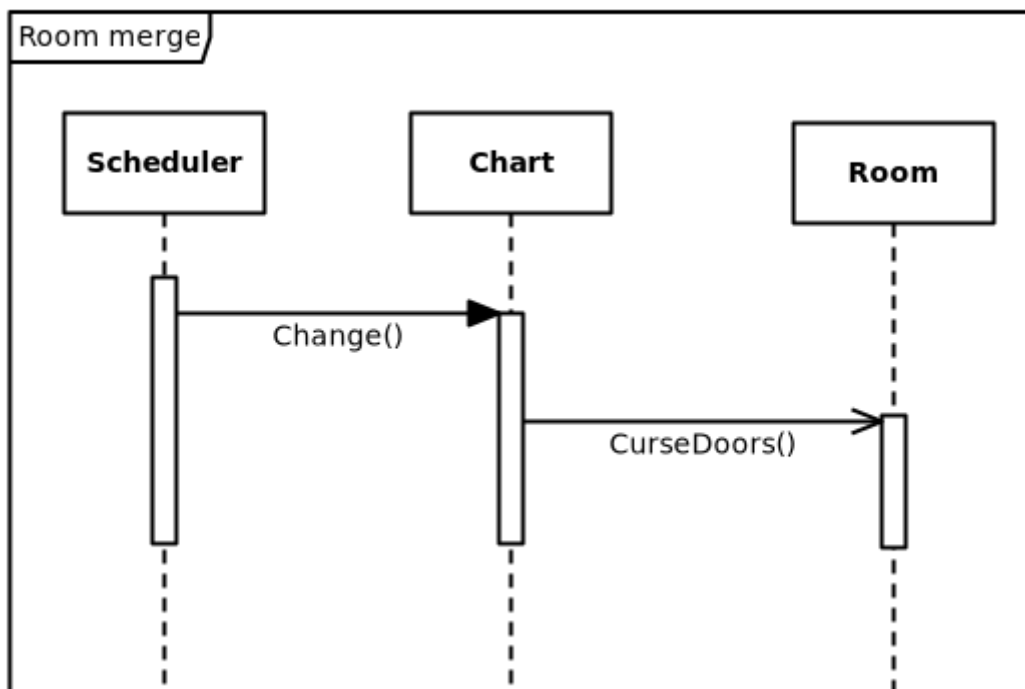
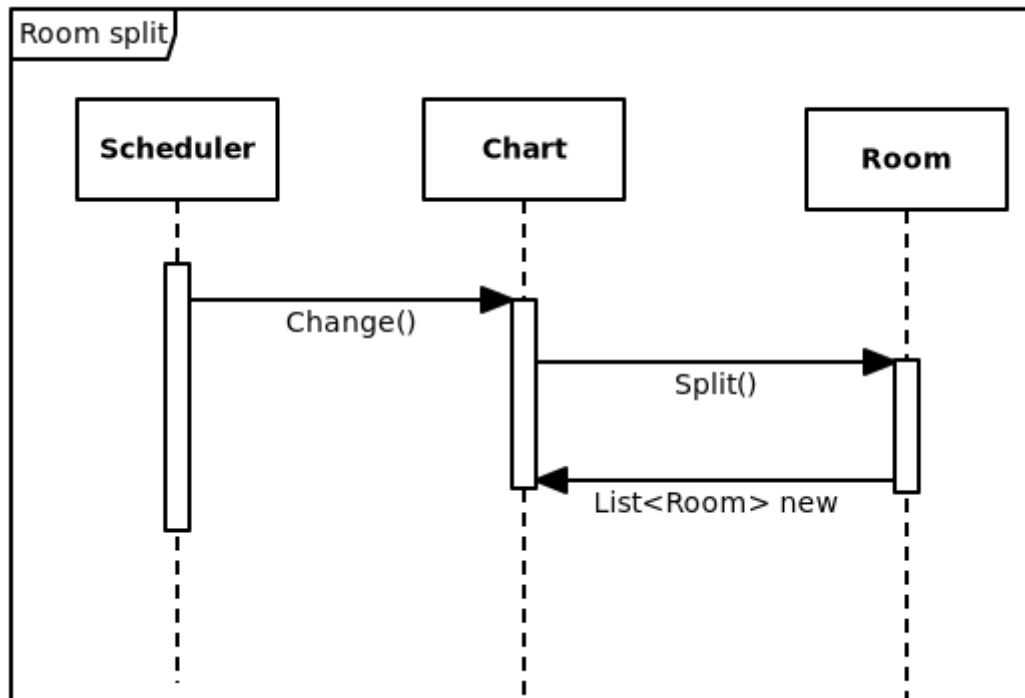


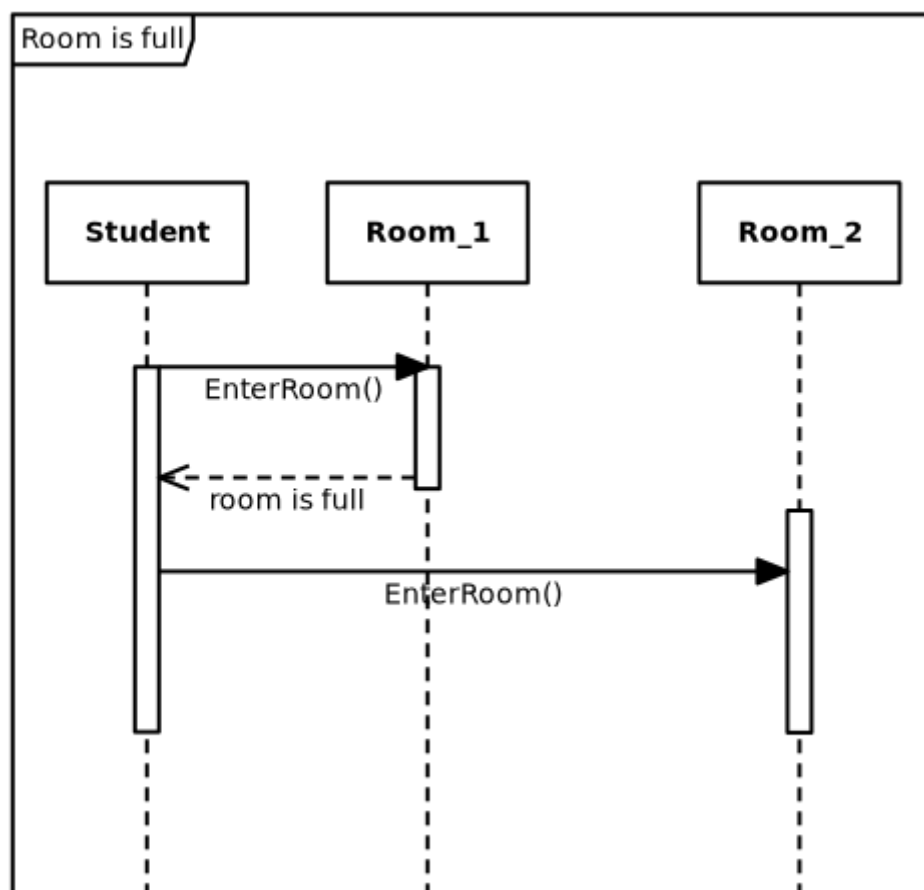
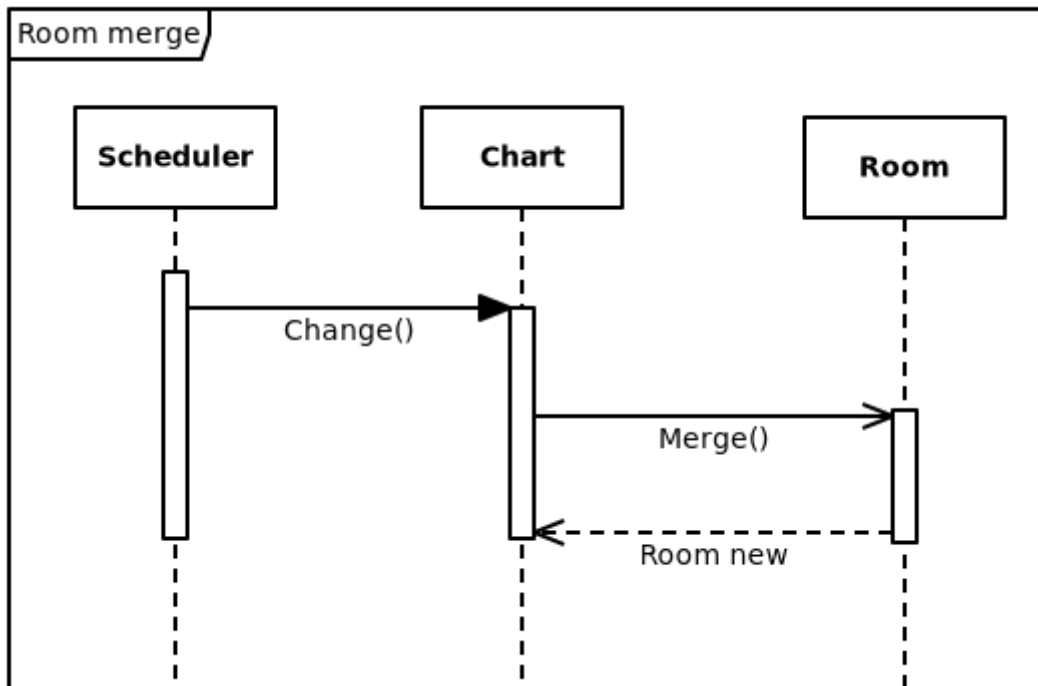












Napló

Kezdet	Időtartam	Résztevők	Leírás
2024.02.28.	1,5 óra	Osztrogonác Kiss	Dokumentum elkezdése. Objektum katalógus elkészítése.
2024.02.29. 15:00	1,5 óra	Osztrogonác Kiss Gáspár Schmidt Halász	Megbeszélés, feladat tisztázása.
2024.02.29. 19:10	2 óra	Schmidt Gáspár	Osztálydiagram befejezése, Osztály leírás elkezdése.
2024.03.01 11:45	2 óra	Gáspár	Osztály leírás
2024.03.01 13:45	1,25 óra	Gáspár Schmidt	Osztály leírás folytatás és konzultálás.
2024.03.01 15:00	1,25 óra	Schmidt	Osztály leírás folytatása.
2024.03.01 18:20	2 óra	Halász	Szekvencia diagramok
2024.03.01 21:00	2,75 óra	Kiss	Osztálydiagram és GameFlow pontosítás
2024.03.02 10:30	3 óra	Schmidt Halász	Szekvencia diagramok
2024.03.02.13:30	2,75 óra	Schmidt	Szekvencia diagramok
2024.03.03 9:30	1 óra	Schmidt	Szekvencia diagramok befejezése és dokumentáció elkezdése
2024.03.03. 23:30	2,25 óra	Osztrogonác Kiss	Szekvencia diagramok kiegészítése
2024.03.04. 2:10	1,5 óra	Osztrogonác Kiss	Osztályok leírása kiegészítés. Diagramok beillesztése. Dokumentum szerkesztés