

Szoftvertchnológia és -technikák

3. Előadás

Szekvenciadiagram



Tartalom

Osztálydiagram (ismétlés)

Szekvenciadiagram



Modellezés – UML - Osztálydiagram

Ismétlés

Szoftverfejlesztés nagyban

- Szoftver
 - > Kis feladat → “Csak meg kell írni”
 - > Komplex feladat → Érdemes megtervezni
- Tervezés
 - > Ha a kódméret/feladat komplex
 - > Ha egyeztetni kell a megrendelővel
 - > Ha meg kell becsülni a költségeket
 - > Ha több ember fejleszti, új belépők is lehetnek
 - > Ha segédmunkások is vannak
- Megoldás
 - > Modellezés



Szoftvermodellezés

- Igény a szoftverek tervezésére
 - > Magas absztrakciós szint, vizuális ábrázolás
 - > '90-es évektől több modellező nyelv: kaotikus helyzet
 - > 1997. – Unified Modeling Language (UML)
- Unified Modeling Language
 - > Cél: szoftverfejlesztés és dokumentáció modellezése
 - > Általános modellező nyelv
 - > A szabvány modellezés terén



Osztályok és interfészek

- Osztályok és interfészek

- > Fejléc

- > Tagváltozók

- [Láthatóság] [Név]: [Típus][Multiplicitás] [= kezdőérték]
 - Statikus: aláhúzás

- > Metódusok/Műveletek

- [Láthatóság] [Név]([paraméterek])([: visszatérési érték])

- > Láthatóság

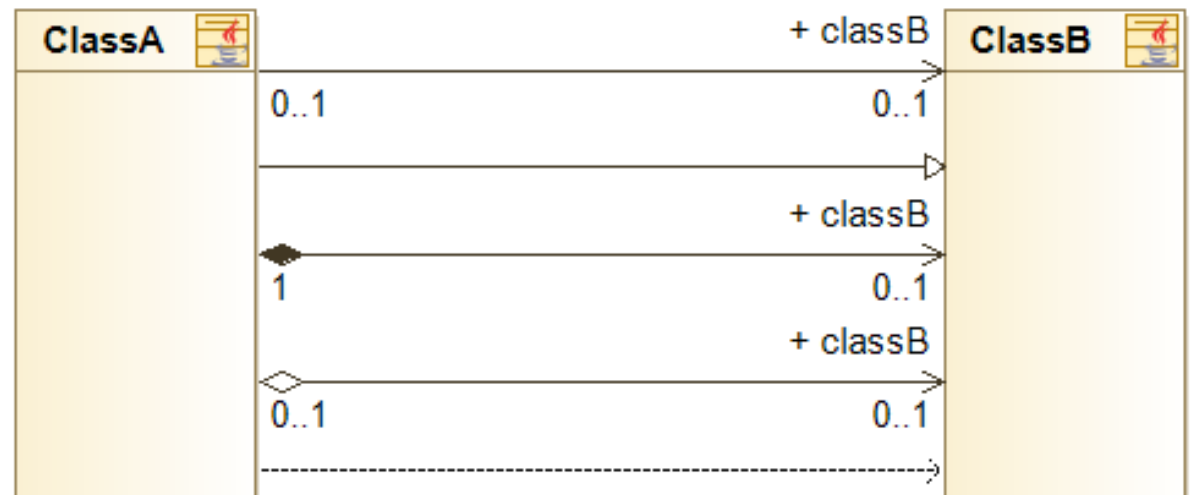
- Public: +
 - Private: -
 - Protected: #
 - Package: ~



Kapcsolatok

- Kapcsolat típusok

- > Asszociáció
- > Általánosítás/Öröklés
- > Kompozíció
- > Aggregáció
- > Függőség
- > Interfész megvalósítás



- Multiplicitás

- Navigálás

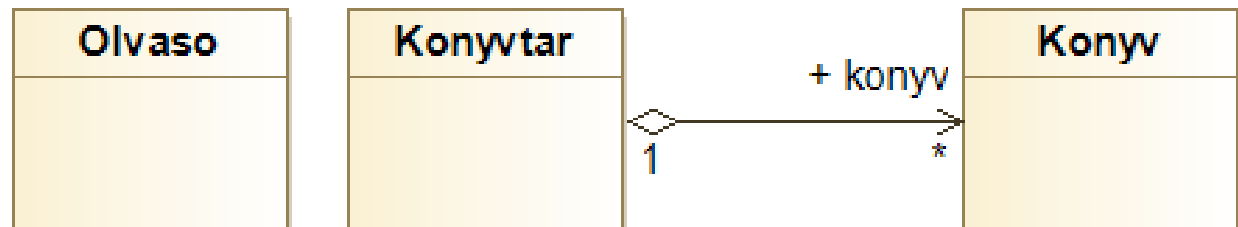
- > Szerepnév
- > Navigálhatóság



Osztálydiagram: könyvtár

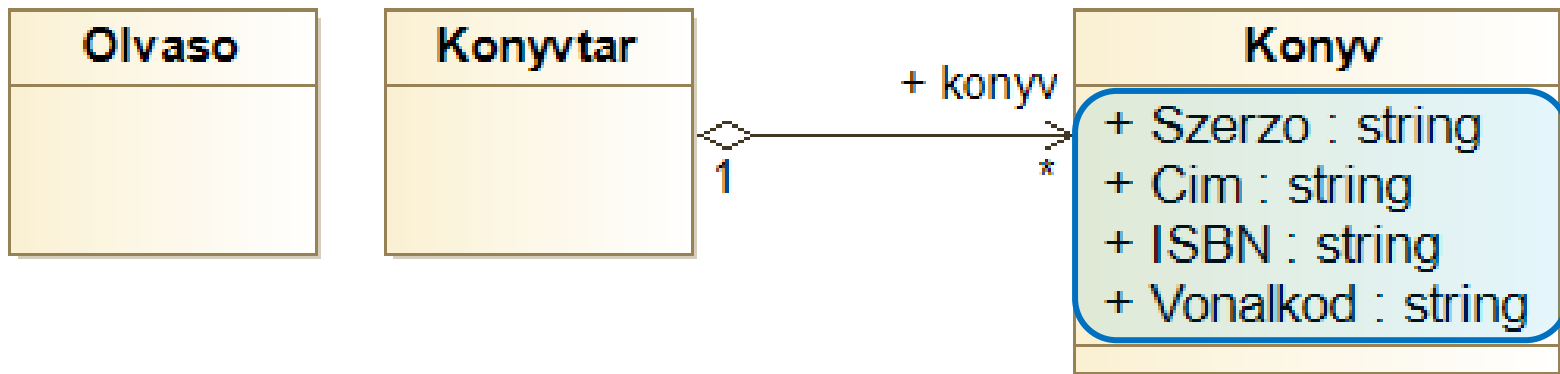
Könyvtár

- Hozzuk létre egy nyilvántartó rendszert könyvtárak számára!
 - > Legyenek benne olvasók és könyvek!
 - > ... természetesen a könyvtárat is modellezni kell
 - > ... a könyvtár tartalmaz könyveket



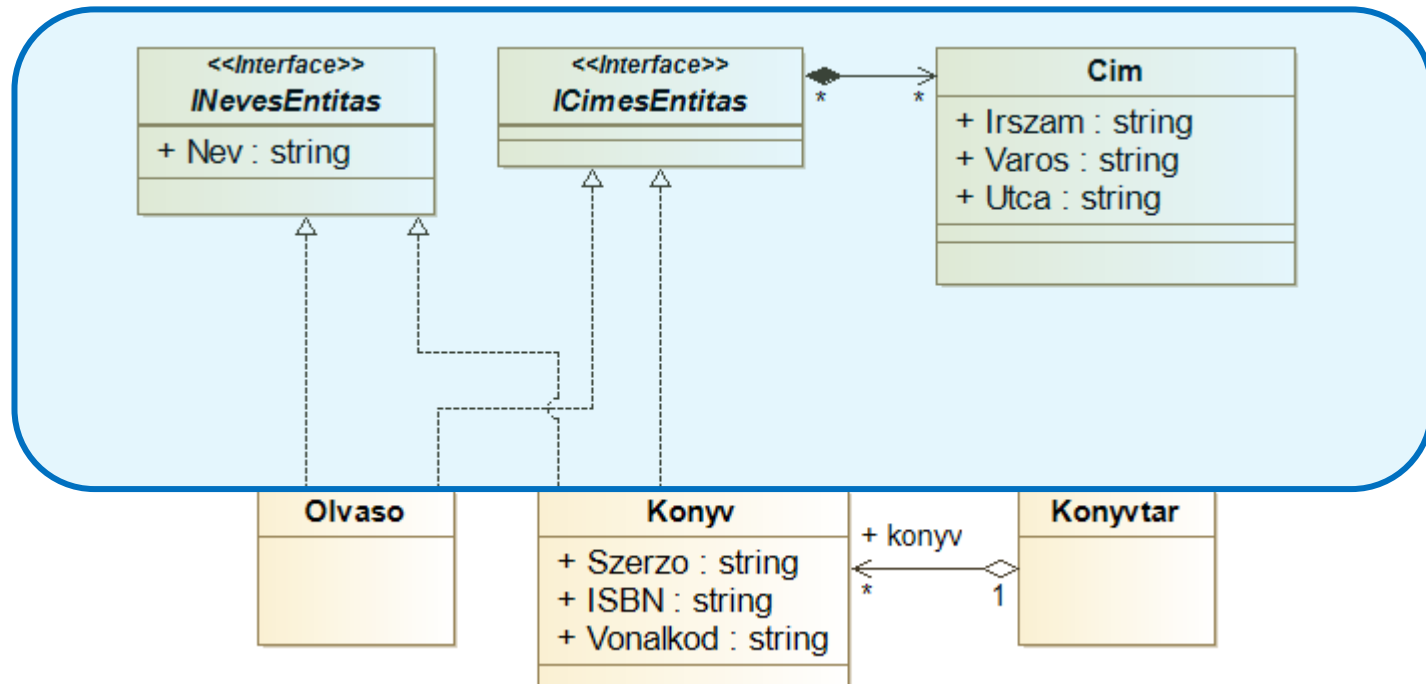
Könyvtár - adatok

- A könyveknek van szerzője, címe és ISBN száma
 - > Mi a könyv példány? Mi a könyv osztály?
 - > ISBN szám vs. egyedi azonosító
 - ISBN: Statikus?



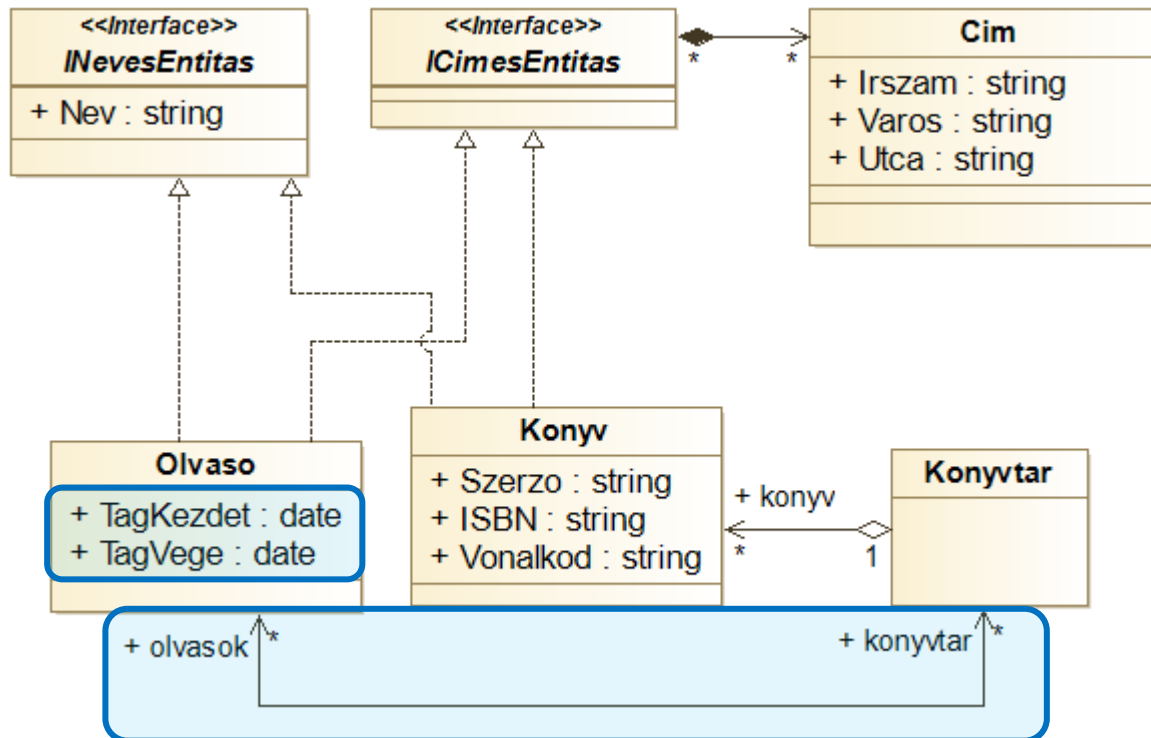
Könyvtár - adatok

- A könyvtárnak és az olvasónak van neve és címe
 - > Közös ős?
 - > Vagy... közös interfész? Egy interfész, vagy kettő?
 - > Vagy... csak egy saját, összetett típusú attribútum?
 - > Vagy... összetett attribútum + interfész?



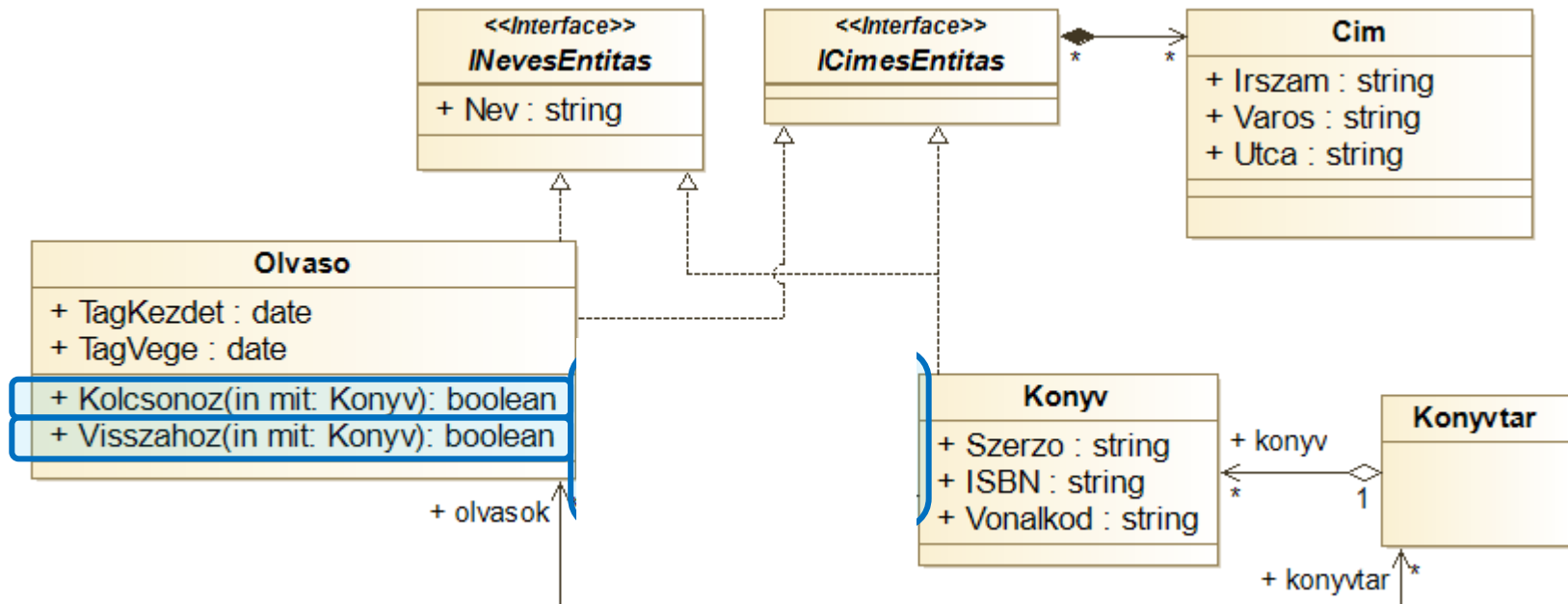
Könyvtár - adatok

- Az olvasó tagságának van kezdete és vége
 - > A könyvtár-olvasó kapcsolatot is érdemes lenne számontartani!



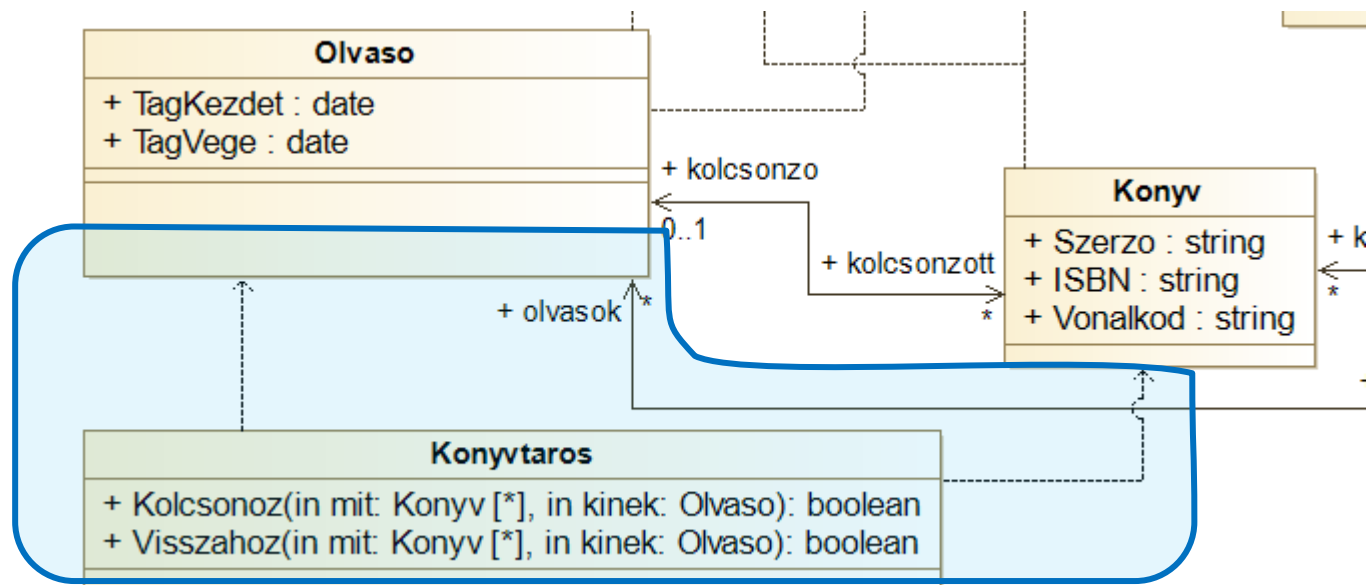
Könyvtár - kölcsönzés

- Az olvasó kölcsönözhet könyveket.
 - Meg kell jegyezni, kinél van az adott könyv
 - Tartalmazás vagy asszociáció?
 - Vissza is kell tudni hozni



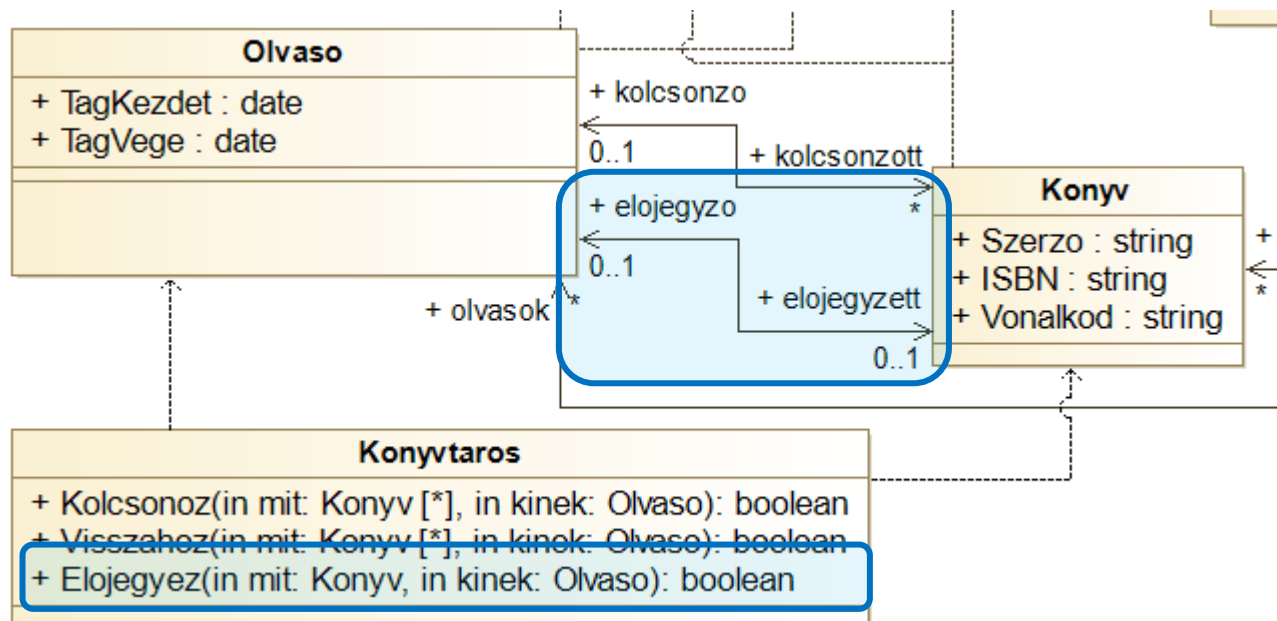
Könyvtár – könyvtáros

- Az olvasó kölcsönözhet könyveket.
 - > Az olvasó kölcsönzi?
 - > ... vagy a könyvtár?
 - > ... vagy a könyvtáros?



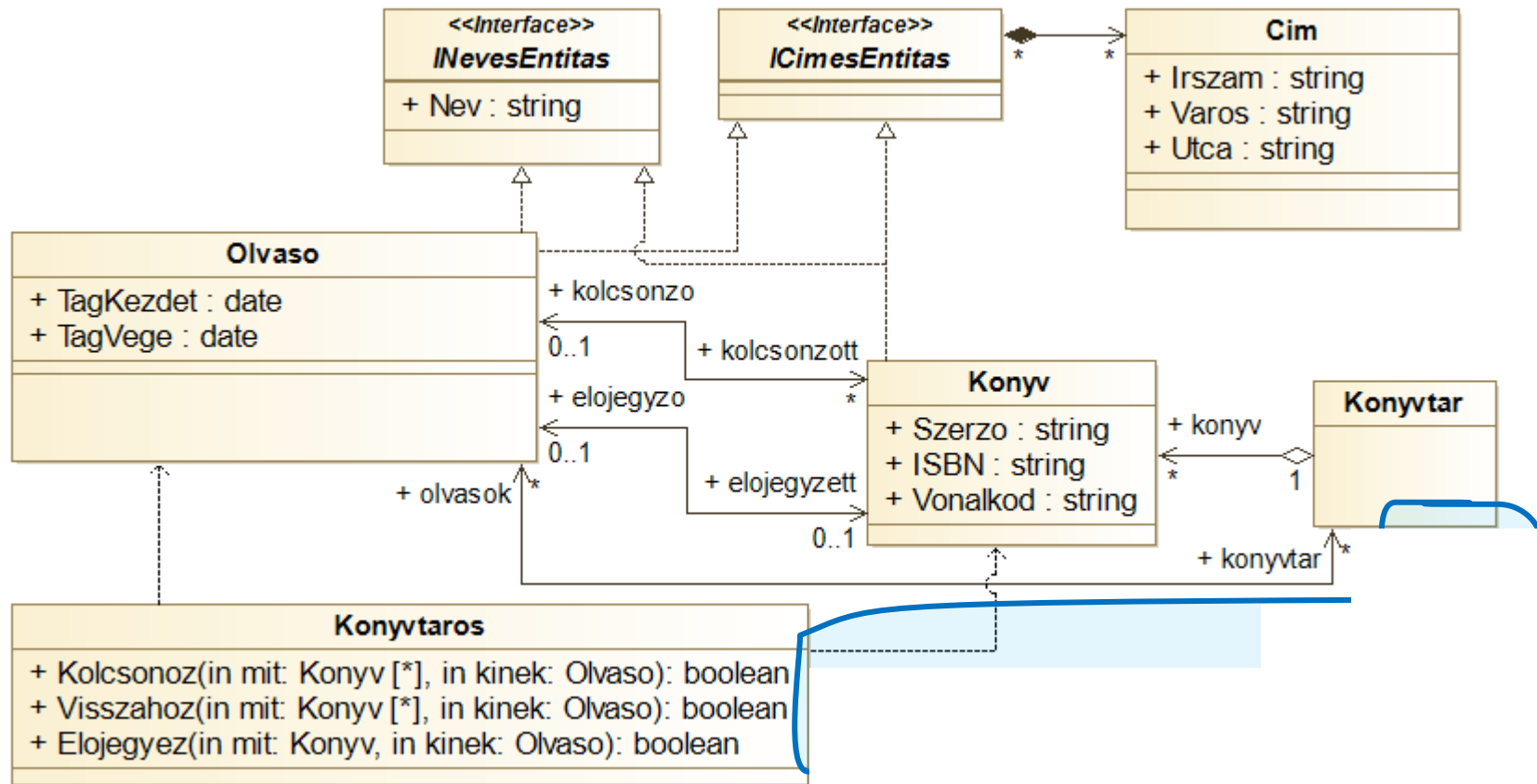
Könyvtár - előjegyzés

- Az olvasó elő is jegyezhet könyvet, de csak egyet!
 - > Az előjegyzés műveletet is támogatni kell
 - > Kiemeljük a könyv-olvasó adatpárost külön típusba?



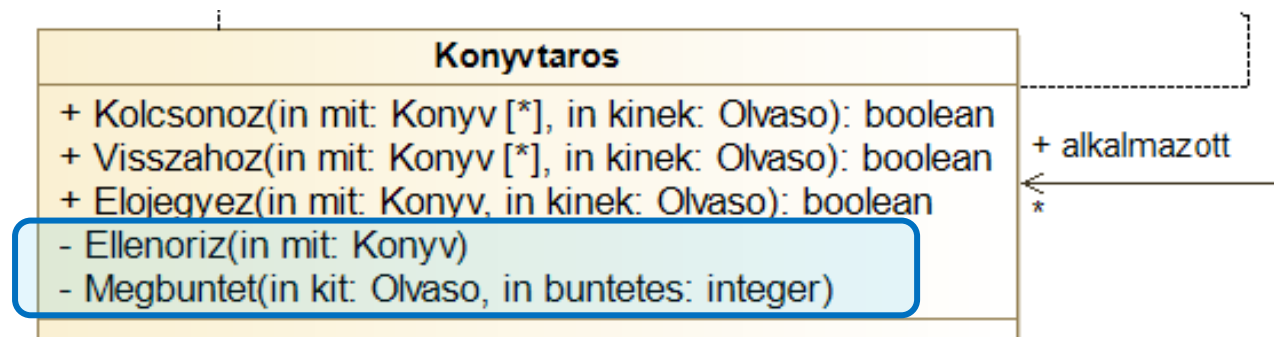
Könyvtár – könyvtáros

- A könyvtáros munkahelye a könyvtár



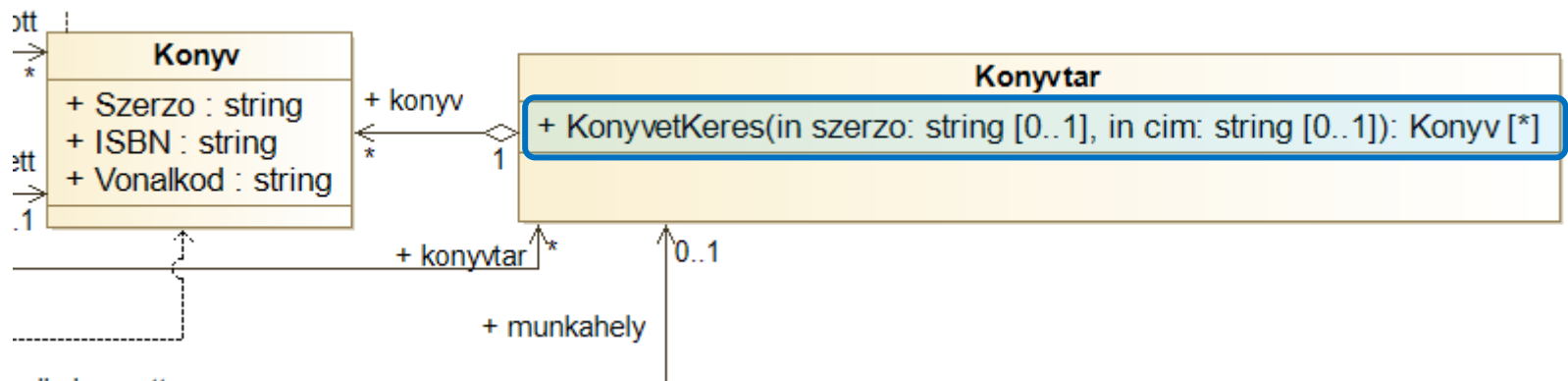
Könyvtár – könyvtáros

- A könyvtáros tudja a könyvet ellenőrizni és büntetést kiszabni, ha szükséges.
 - > Publikus művelet?



Könyvtár - keresés

- Lehesse könyvet keresni a könyvtárban szerző és cím szerint (mindkét parameter elhagyható)!
 - > Mi legyen a visszatérési érték?
 - > Az olvasó, a könyvtáros, vagy a könyvtár keres?



Szekvenciadiagram

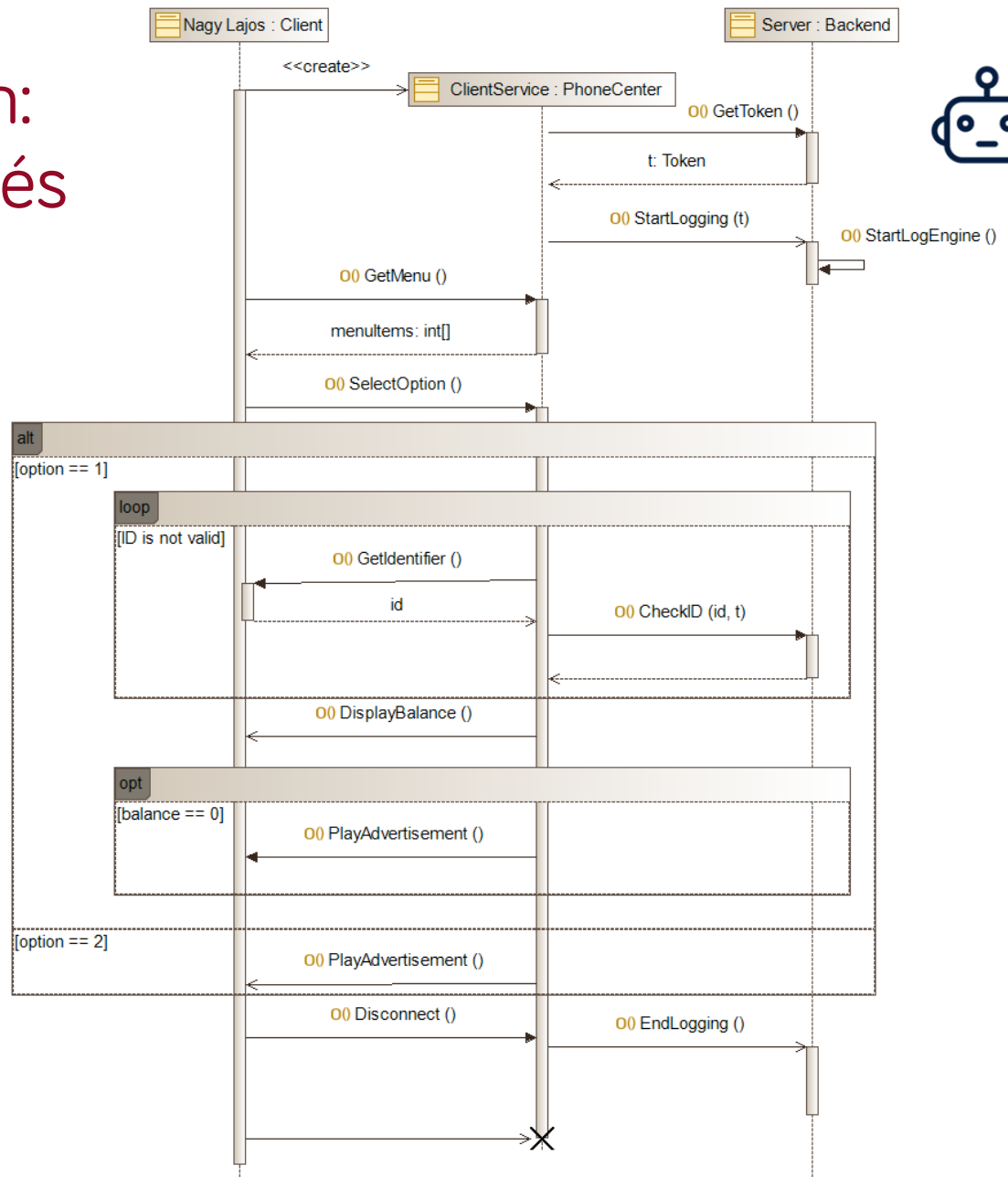
Struktúra vs. viselkedés

- Osztálydiagram
 - > Struktúra, adatszerkezet
 - > Metódusok
 - > Statikus
- Hogyan írjuk le a dinamikus működést?
 - > Szekvenciadiagram
 - > Állapotgép
 - > Aktivitásdiagram
 - > Használati eset diagram

Szekvenciadiagram

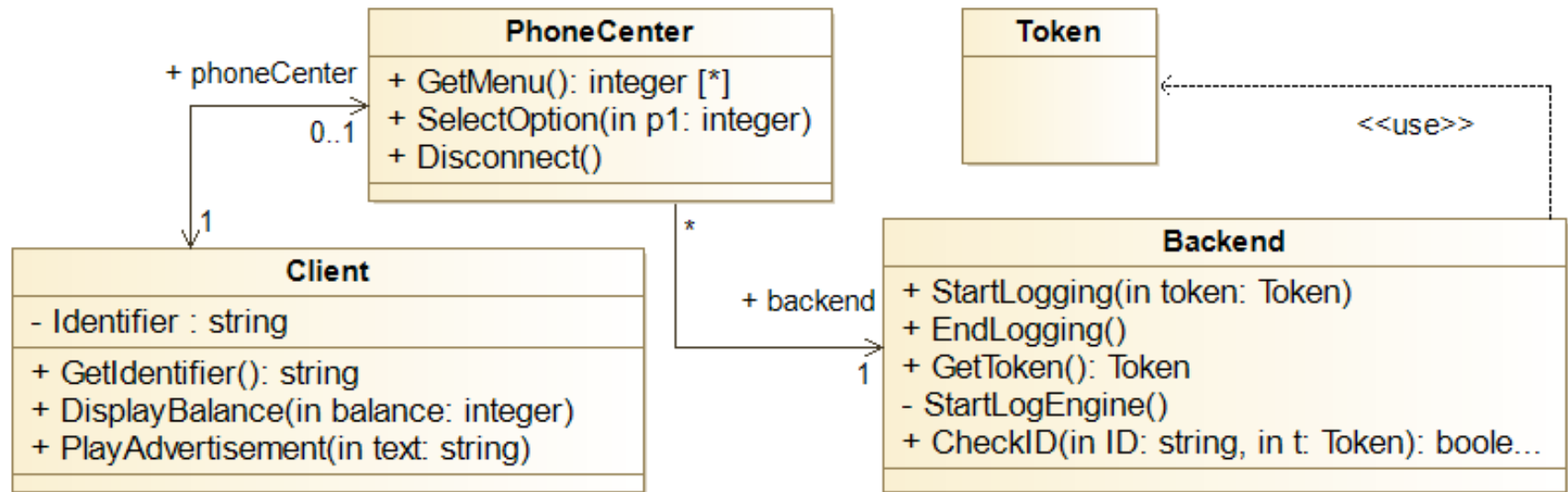
- A folyamatok, műveletek sorrendjére fókuszál
 - > Üzenetek, válaszüzenetek
 - > Interakciók
 - > Információcsere
- Rendszer-, és metódusszinten is alkalmazható
- Egy adott lefutást ábrázol
 - > Ugyanazon résztvevők közt több lefutás is lehetséges (több szekvenciadiagram ábrázolható)
- Résztvevők: osztályok példányai és *aktorok*
 - > Nem “független”

Szekvenciadiagram: Telefonos ügyintézés



Osztálydiagram

- Értelmezzük az alábbi struktúrát!



Életvonalak és üzenetek

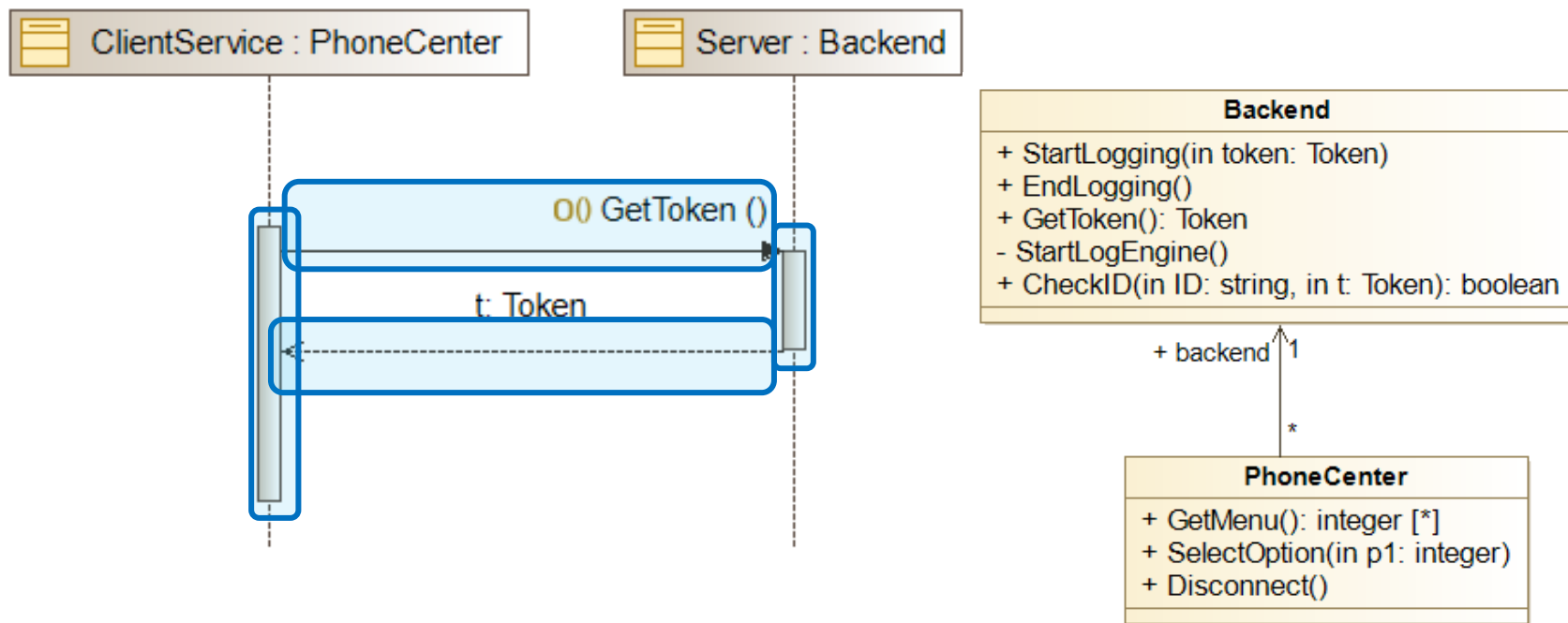
Életvonal

- Vegyünk egy konkrét ügyfelet, aki egy konkrét telefonos szervízzel kommunikál majd a server jóváhagyásával!
 - > Életvonal (Life line)
 - > Nem osztályok szerepelnek, hanem példányok!
 - > A diagram épít az osztálydiagramra!
 - > Idő fentről lefelé telik



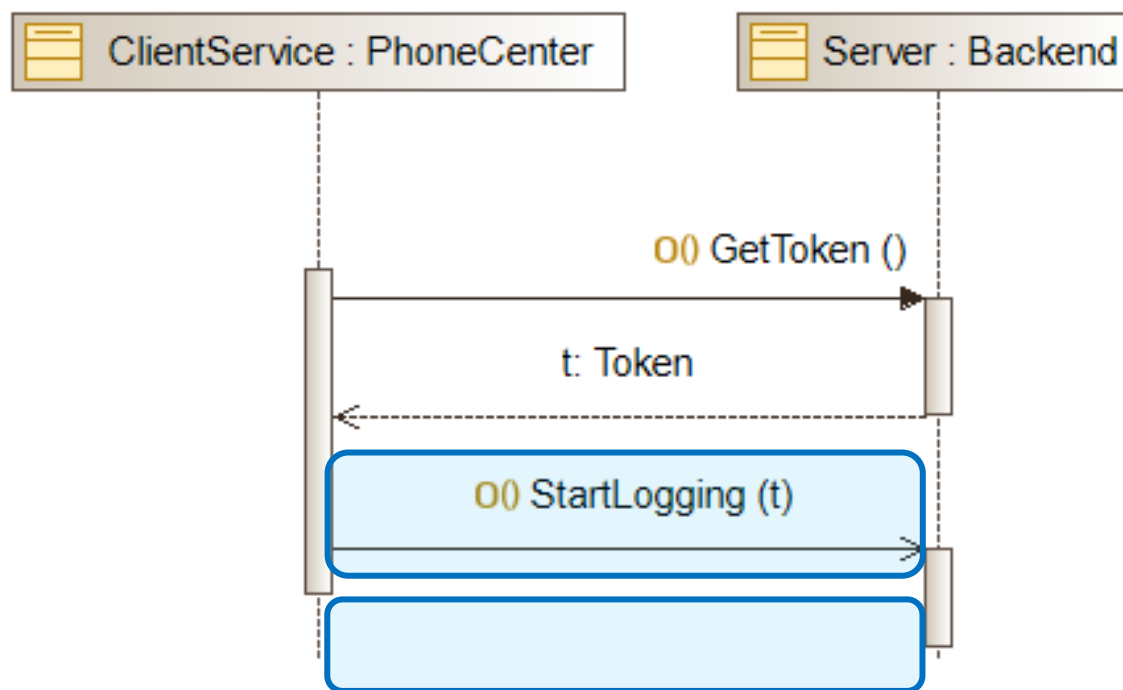
Üzenetek

- A szervíz kérjen el egy token a backendtől!
 - > Szinkron függvényhívás, visszatérési értékkel
 - > Futási specifikáció (execution specification)
 - > Üzenet (message)
 - > Válaszüzenet (return message)



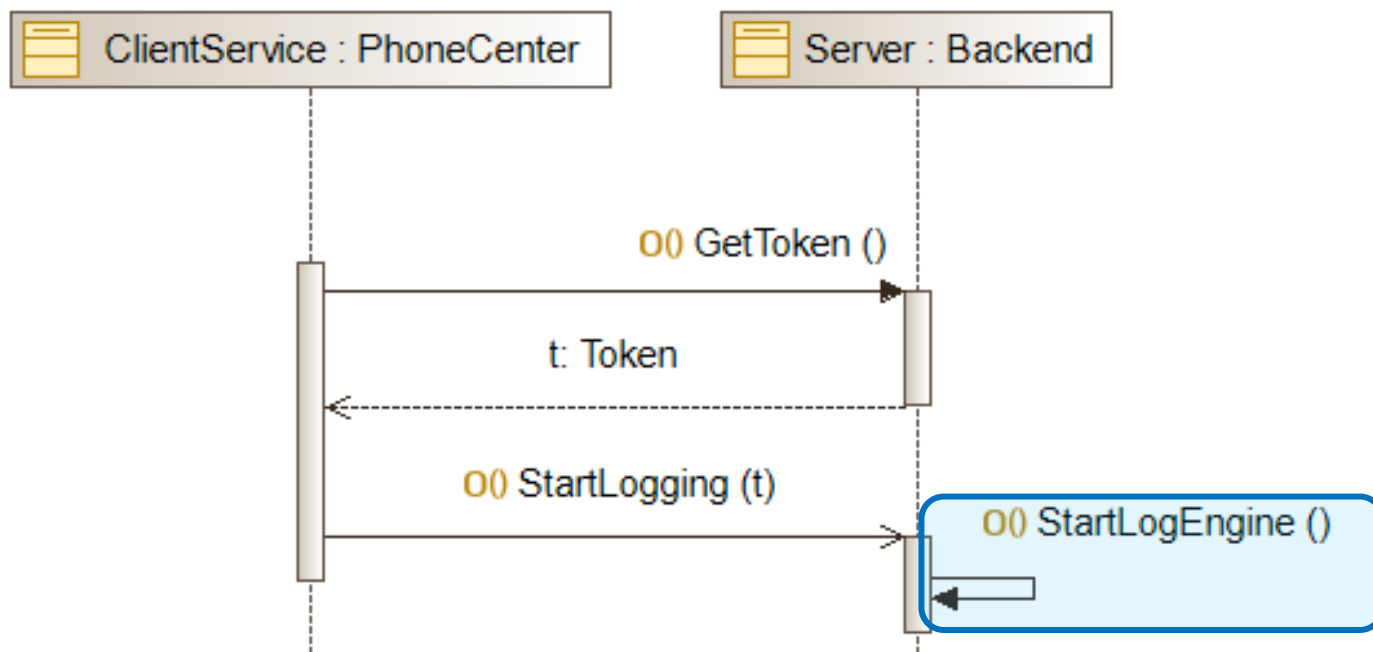
Üzenetek

- A szervíz indítsa el a loggolást a szerveren a token segítségével!
 - > Aszinkron függvényhívás, paraméterrel
 - > Nincs válaszüzenet!



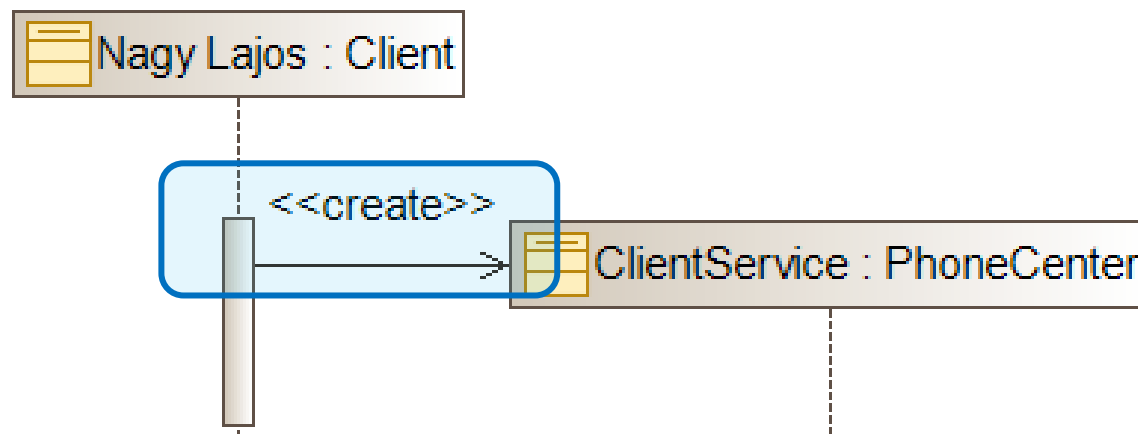
Üzenetek

- A szerver belső folyamatként indítsa el a loggoló metódusát, amikor felkérés jön a szervízről!



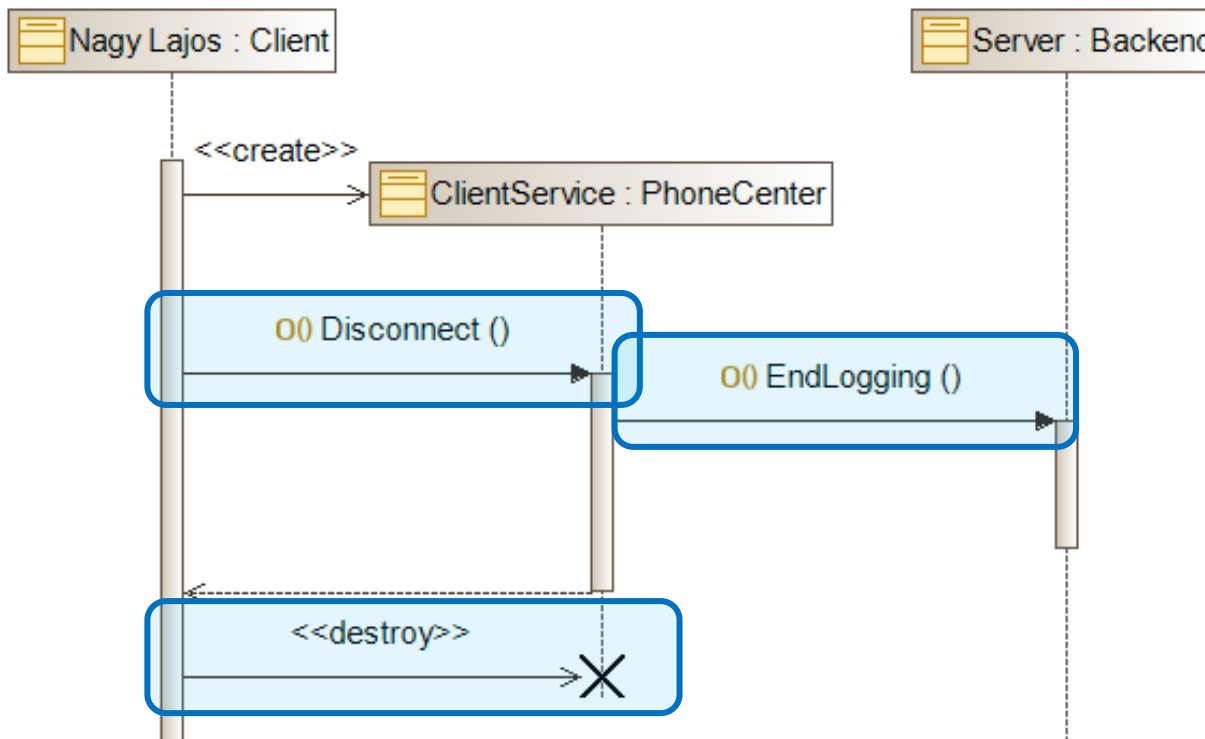
Üzenetek

- A szervíz csak a hívás elején jön létre!
 - > Létrehozási üzenet, konstruktor
 - > “Create” sztereotípia
 - > Később létrejövő életvonal



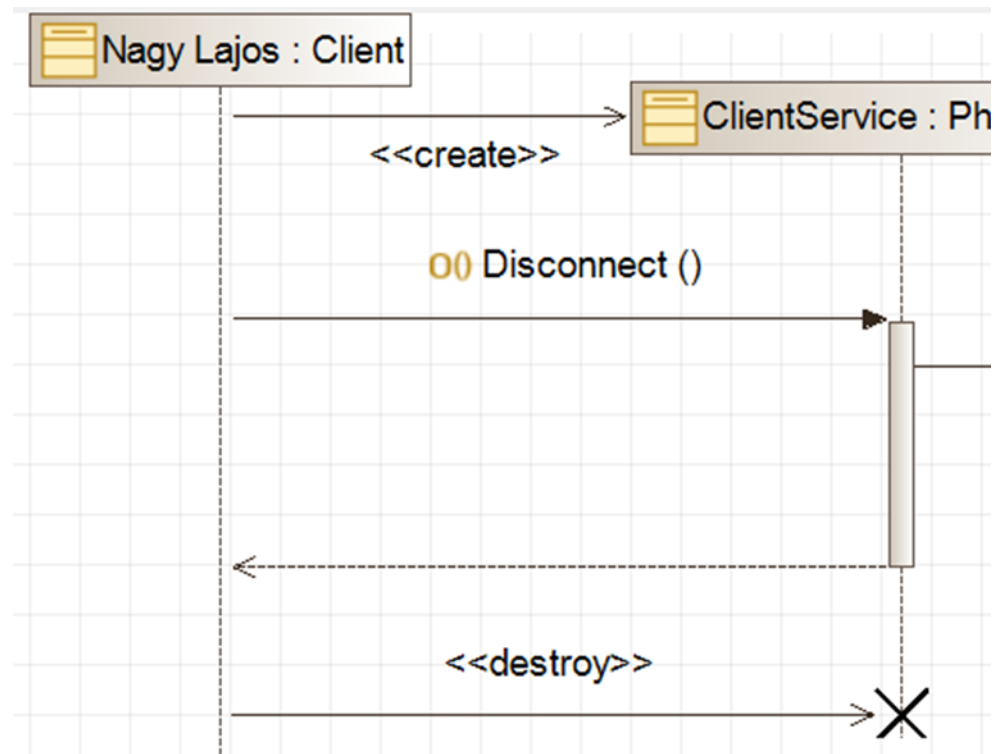
Üzenetek

- Modellezük le a hívás végét
 - > Hívó értesíti a szervízt, ami leállítja a loggolást
 - > Majd megszünteti a szervízt (destruktor)



Életvonalak és üzenetek

- Életvonal
 - > [Név]: [típus]
- Üzenet
 - > Név([params])
 - > Szinkron/aszinkron
 - > Létrehozás/
megszűntetés
- Futási/végrehajtási
specifikáció (execution specification)

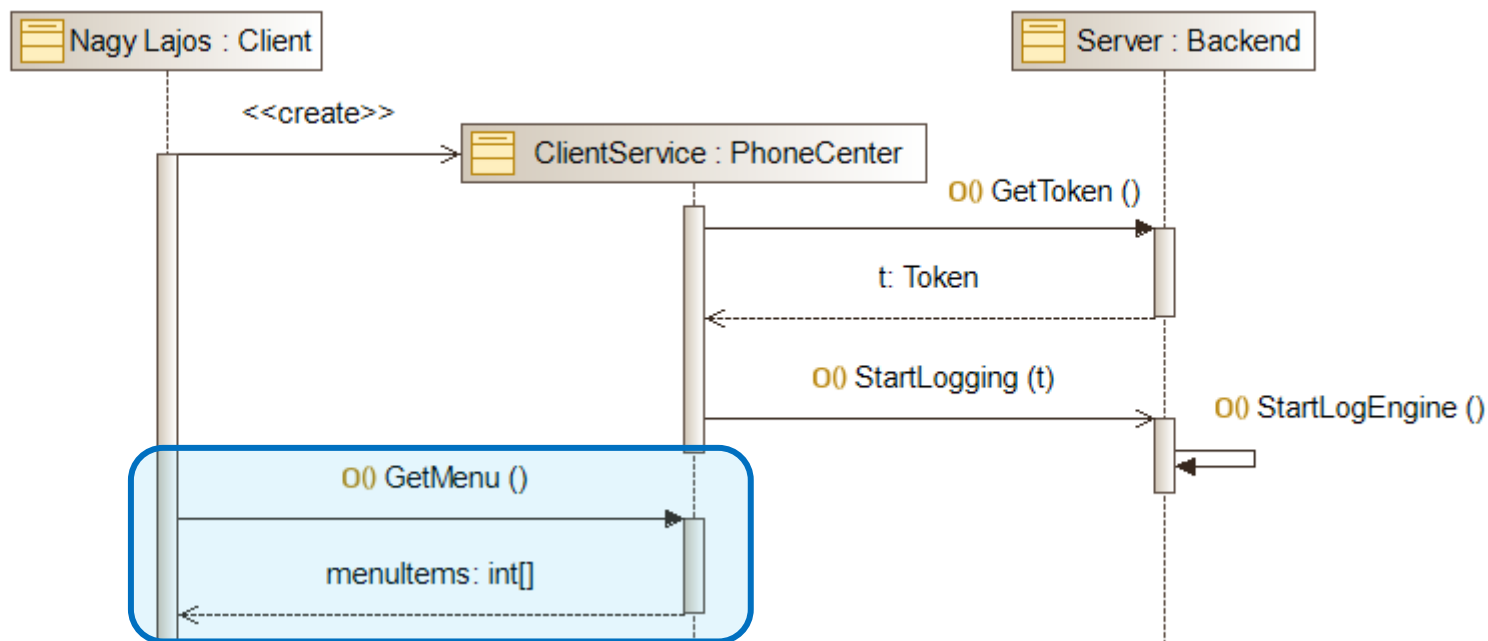


Blokkok

Combined fragments

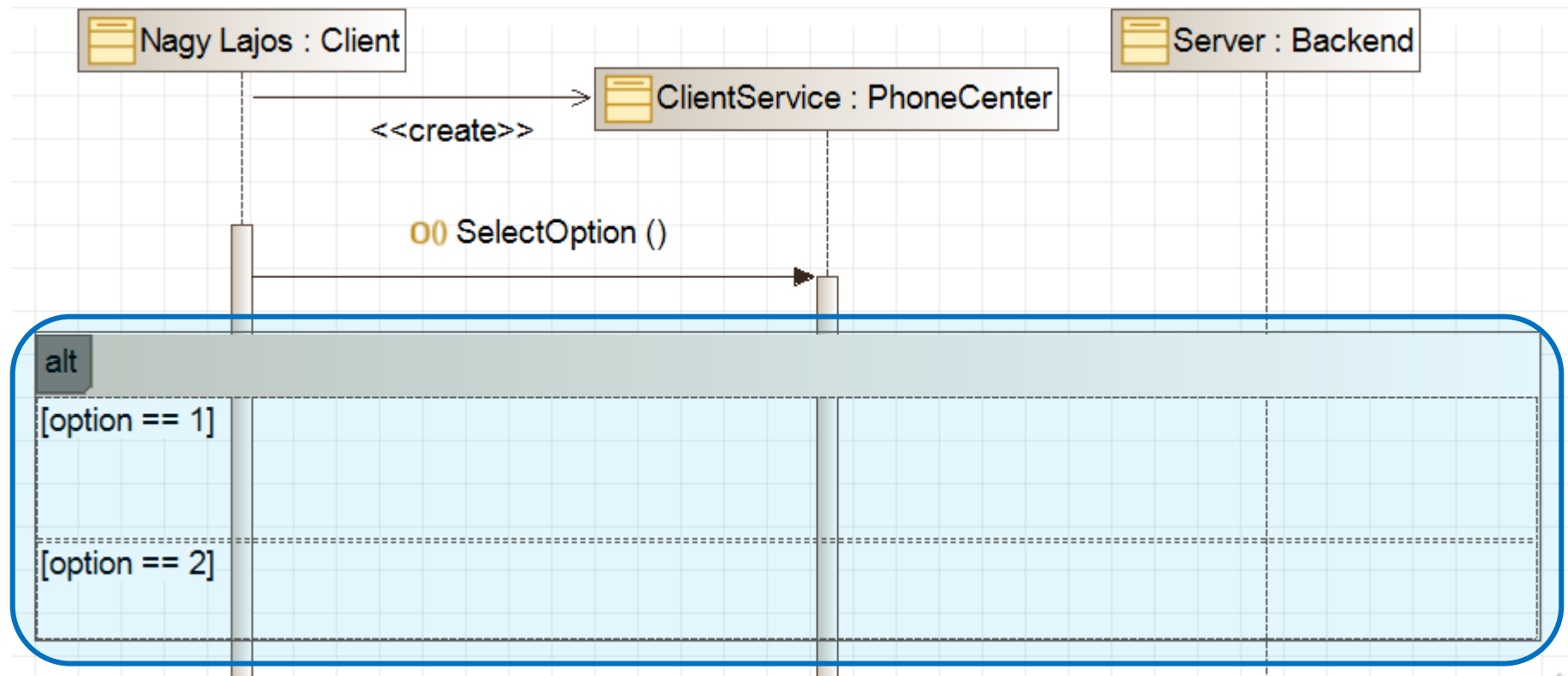
Üzenetek

- A hívó kérhesse le a menüpontokat a szervíztől!



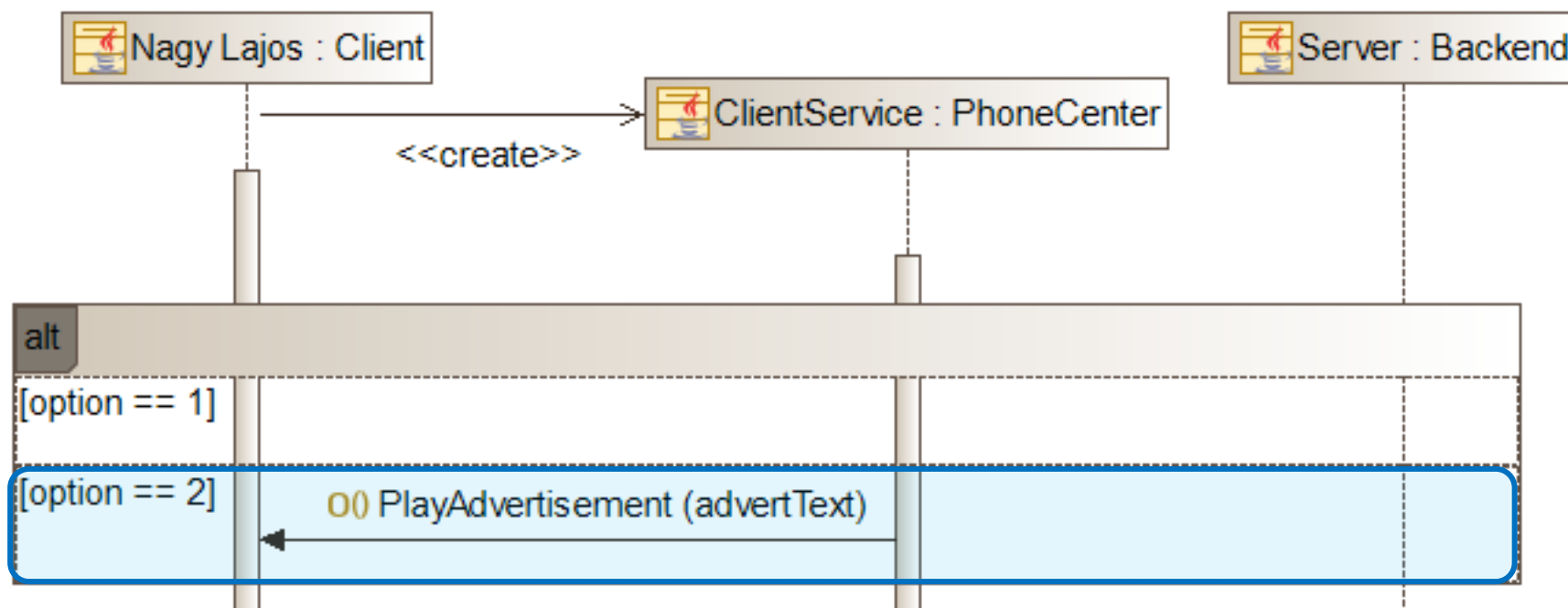
Üzenetek

- Válasszuk szét a műveleteket a hívó választásától függően!
 - > Összetett részlet (combined fragment)
 - > Alternatív blokk (alt) (if – then – else jellegű kompozíció)



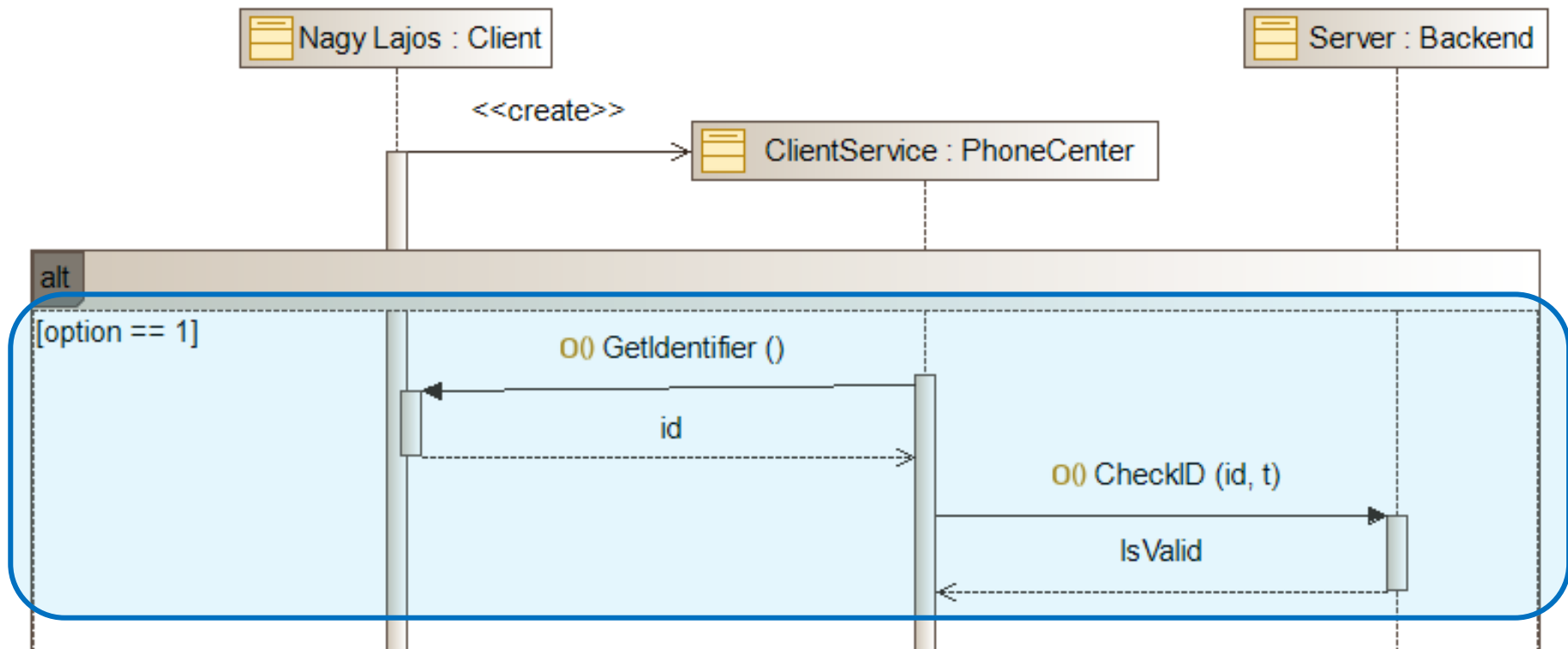
Üzenetek

- A “2-es” menüpont játszon be reklámokat!



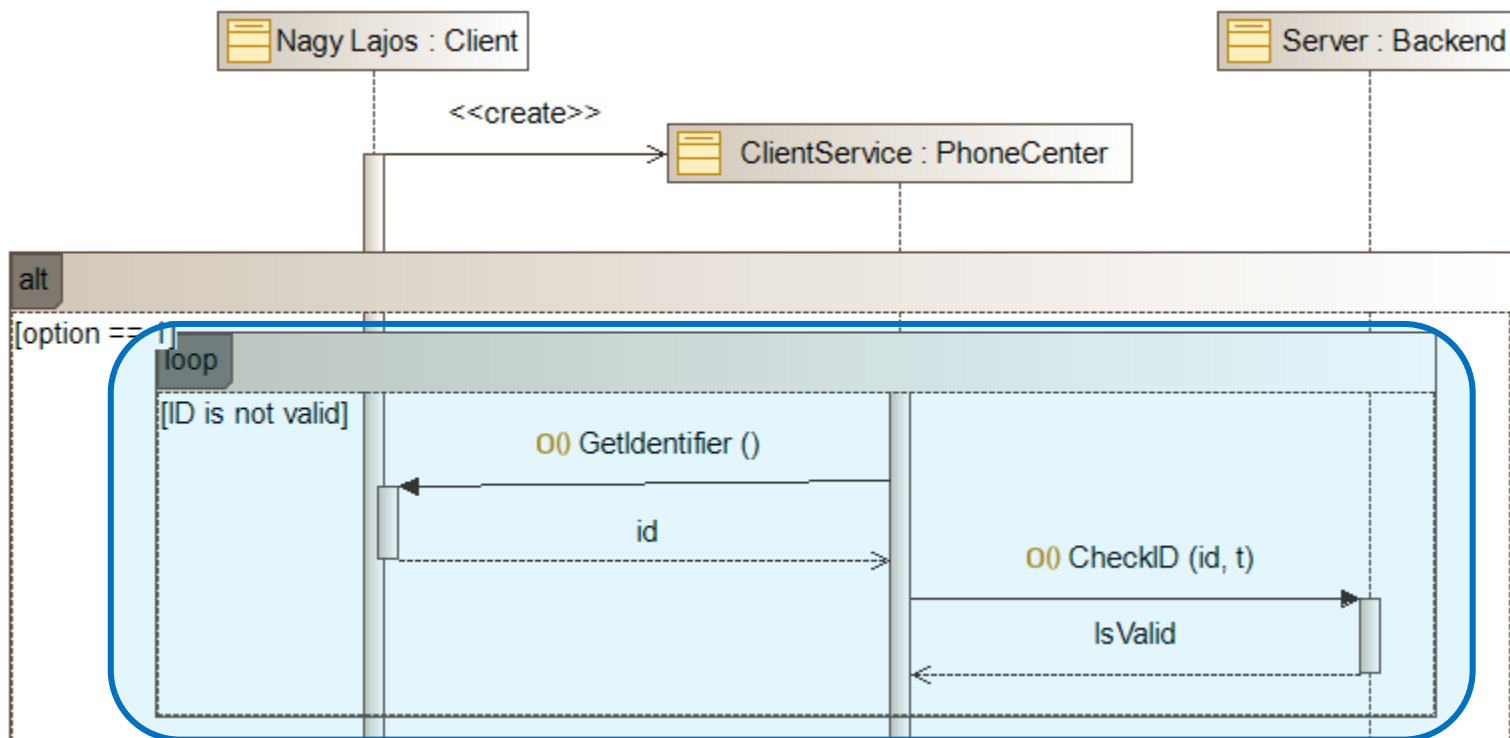
Üzenetek

- Az “1-es” menüpontban kérjük be a hívó azonosítóját és ellenőrizzük is le!



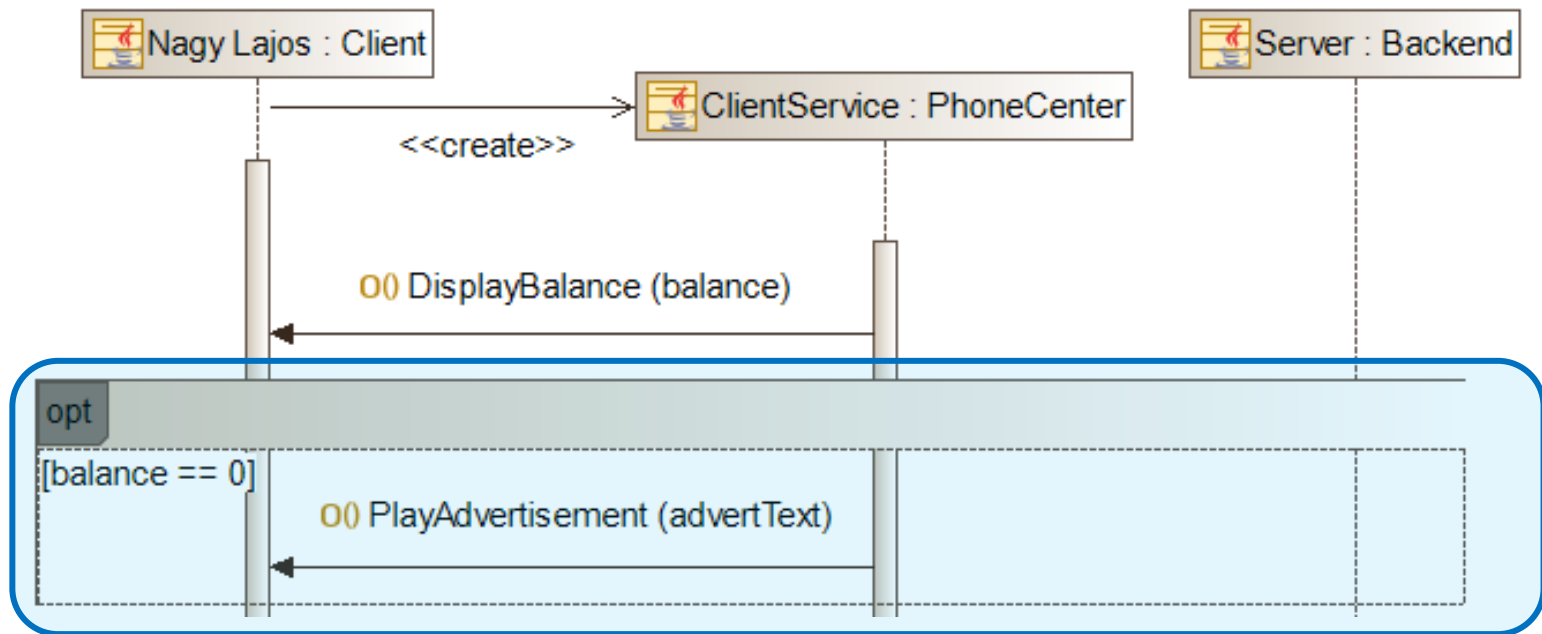
Üzenetek

- Az azonosítót mindaddig újra bekérjük, amíg az nem helyes!
 - > Ciklus blokk (loop) (feltétel szerinti ismételés)



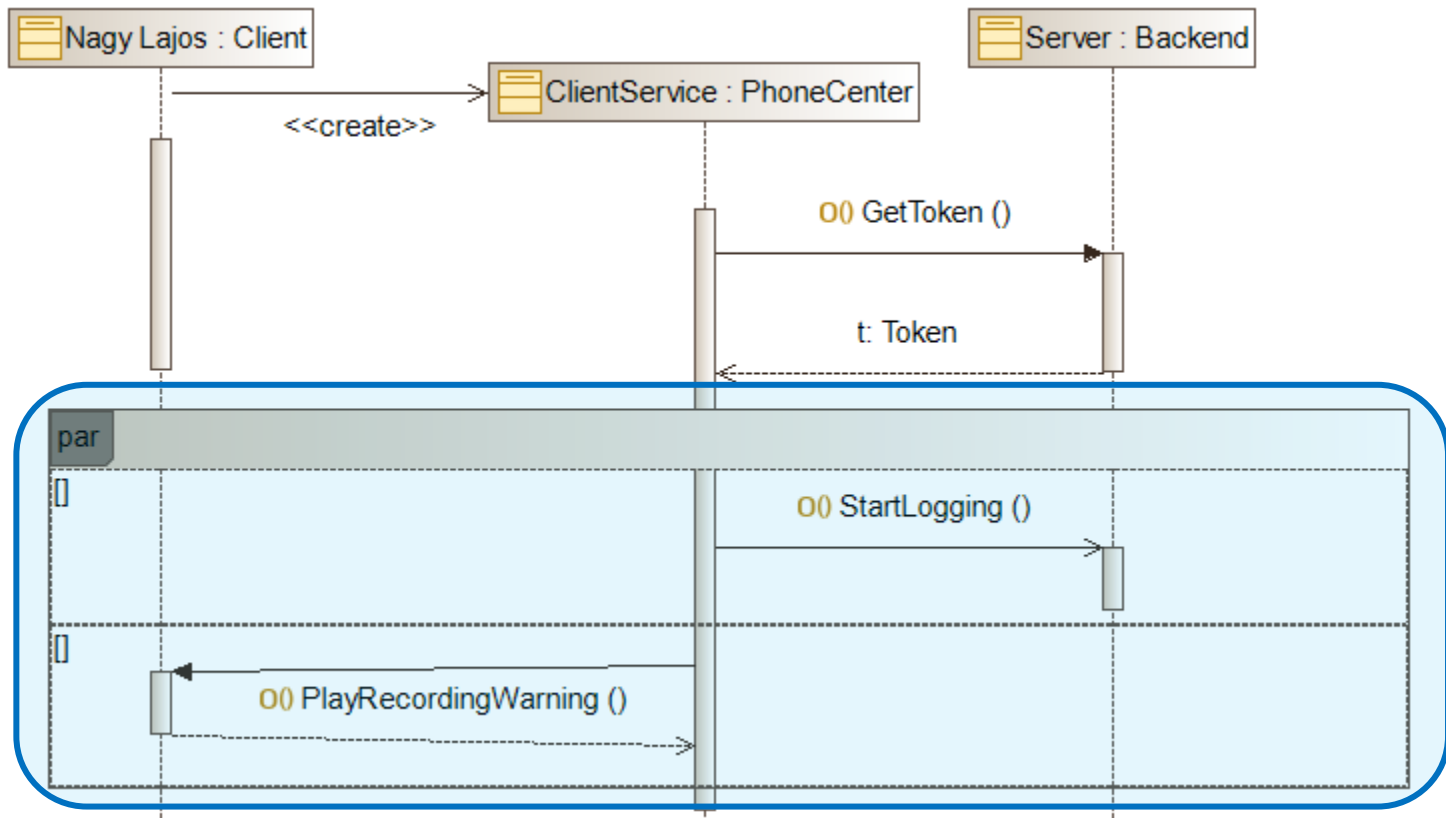
Üzenetek

- Mutassuk meg a hívónak az egyenlegét és ha üres, automatikusan játszunk le neki reklámot!
 - > Opcionális blokk (opt) (if else nélkül)



Párhuzamosság

- A beszélgetés kezdetén a loggolás elindításával egyidőben játszunk le egy figyelmeztetést a hívónak, hogy rögzítjük majd a beszélgetést!
 - > Párhuzamos blokk (par)



Blokkok

- Kombinált részletek (Combined fragments)
 - > Alt: Feltételes elágazás (if – else if – else szerkezet)
 - > Loop: Feltétel szerinti előltesztelő ciklus
 - > Opt: Opcionális utasítások (else ág nélküli if)
 - > Par: Párhuzamosan végrehajtandó utasítások
 - > Seq, Critical, Assert, ... (nem tanuljuk)

Összefoglalás

- Modellezés
 - > Statikus nézet, szerkezet, interfészek -> Osztálydiagram
 - > Metódusok egymás utáni lefutása -> Szekvenciadiagram
- Szekvenciadiagram
 - > Életvonalak
 - > Üzenetek
 - > Blokkok

Köszönöm a figyelmet!