

Használati eset diagram elemei

- ❑ **Használati esetek:** azon tevékenységek (funkciók, szolgáltatások), amelyek a rendszert kívülről szemlélő (azaz a felhasználó vagy más külső szereplő) szempontjából azonosíthatóak, és kezdeményezhetőek.
- ❑ **Kapcsolatok** a használati esetek között, amelyek a használati esetek
 - részekre bontását, illetve plusz funkciókkal történő kiegészítését ábrázolják
 - sorrendjét határozzák meg
 - általános használati esetből történő származtatására mutatnak rá
- ❑ **Aktorok:** a funkciókat használó felhasználók vagy más külső szereplők, akik vagy amik a tervezett rendszer működését vezérlik. Jelölhető az, hogy egy aktor az aktorok mely fajtájához tartozik (specializálás), illetve az, hogy egyfajta aktorból hány darab lehet (multiplicitás).

Használati esetek kapcsolatai

□ Használati eset **részei**

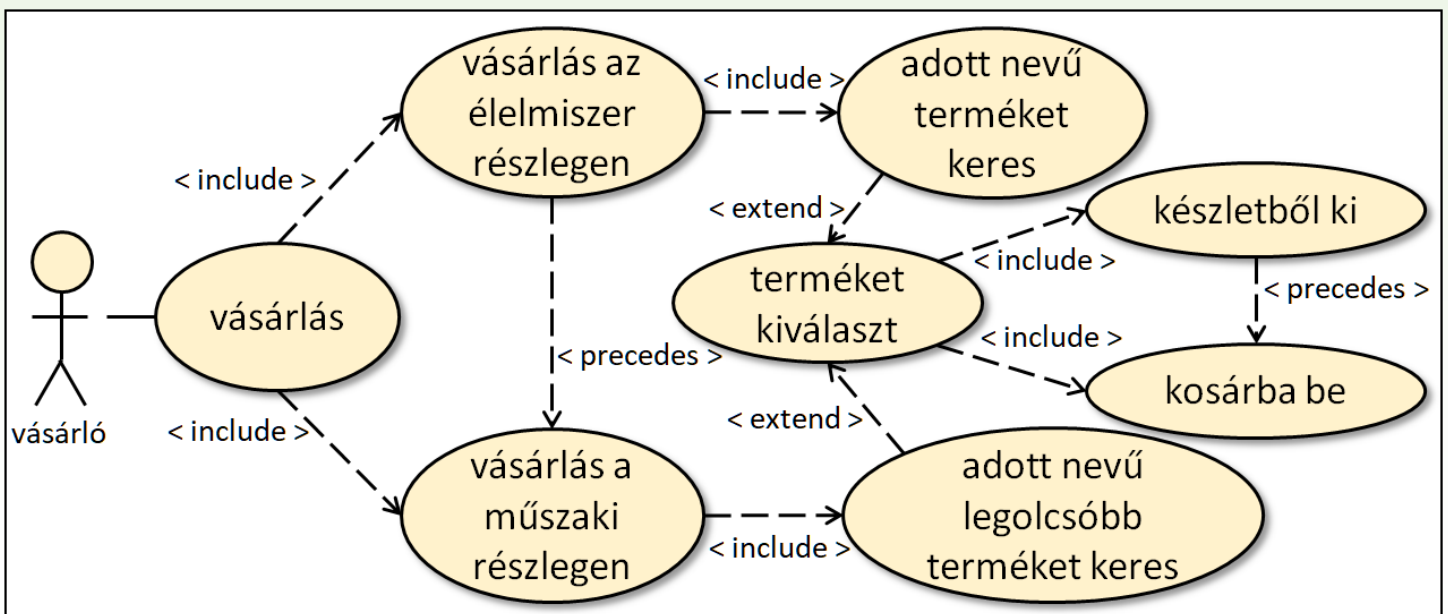
- **include**: rámutat egy használati esetnek egy olyan jól elkülöníthető, önállóan is kezdeményezhető részére, amely nélkül a tartalmazó tevékenység már nem lenne teljes (hanem absztrakt)
- **extend**: egy önmagában is teljes (tehát nem absztrakt) használati esetet opcionálisan kiegészítő másik esetre mutat

□ Használati esetek **rákövetkezési sorrendje**

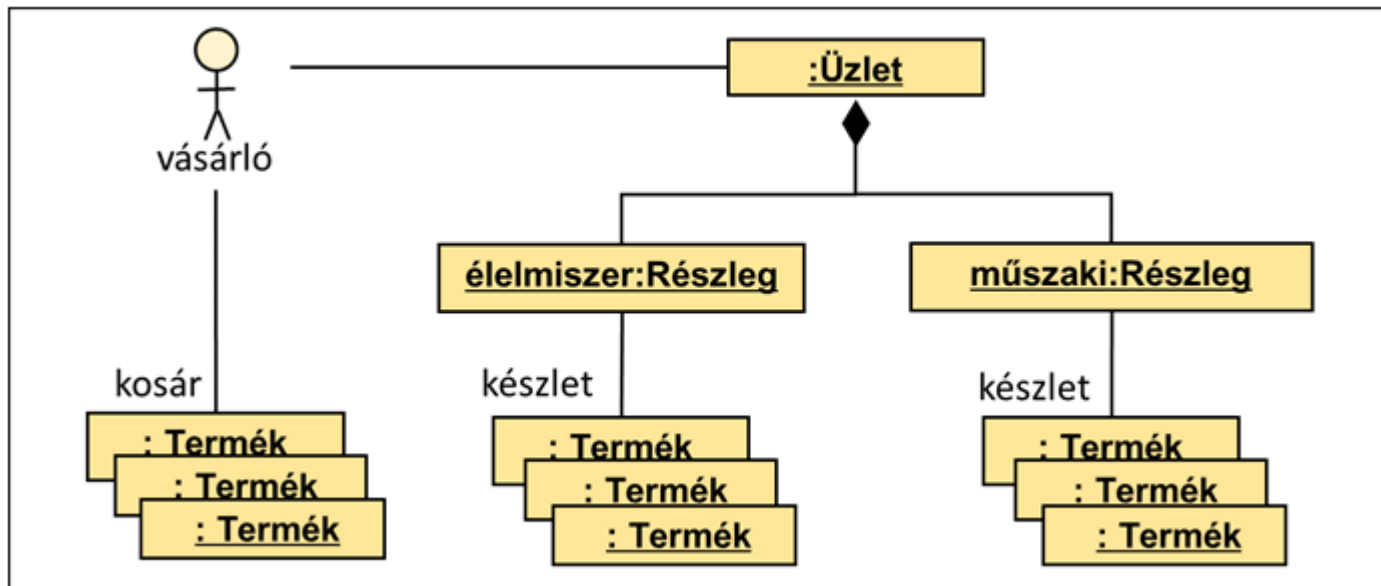
- **precedes**: aktor által közvetlenül kezdeményezhető tevékenységek közötti sorrendet jelöli ki.
- **invokes**: rámutat arra, hogy egy tevékenység végrehajtását mely másik tevékenységnek kell követnie az aktor akaratától függetlenül

Vásárlás

Egy kisvárosi üzlet élelmiszer részlegből és műszaki részlegből áll, ahová a vásárlók egy bevásárlólistával jönnek, amely azon termékek neveit tartalmazza, amit megvennének. Az üzletben a listájukon szereplő **termékeket keresik**: **először az élelmiszer részlegen** nézik végig a teljes bevásárlólistát, és a megtalált termékeket **magukhoz veszik** (beteszik a kosarukba), **majd a műszaki részlegen ezt megismétlik**, de megfontoltabban: ha egy (a bevásárlólistán szereplő) termékből több is van a részlegen, akkor a **legolcsóbbat választják**.

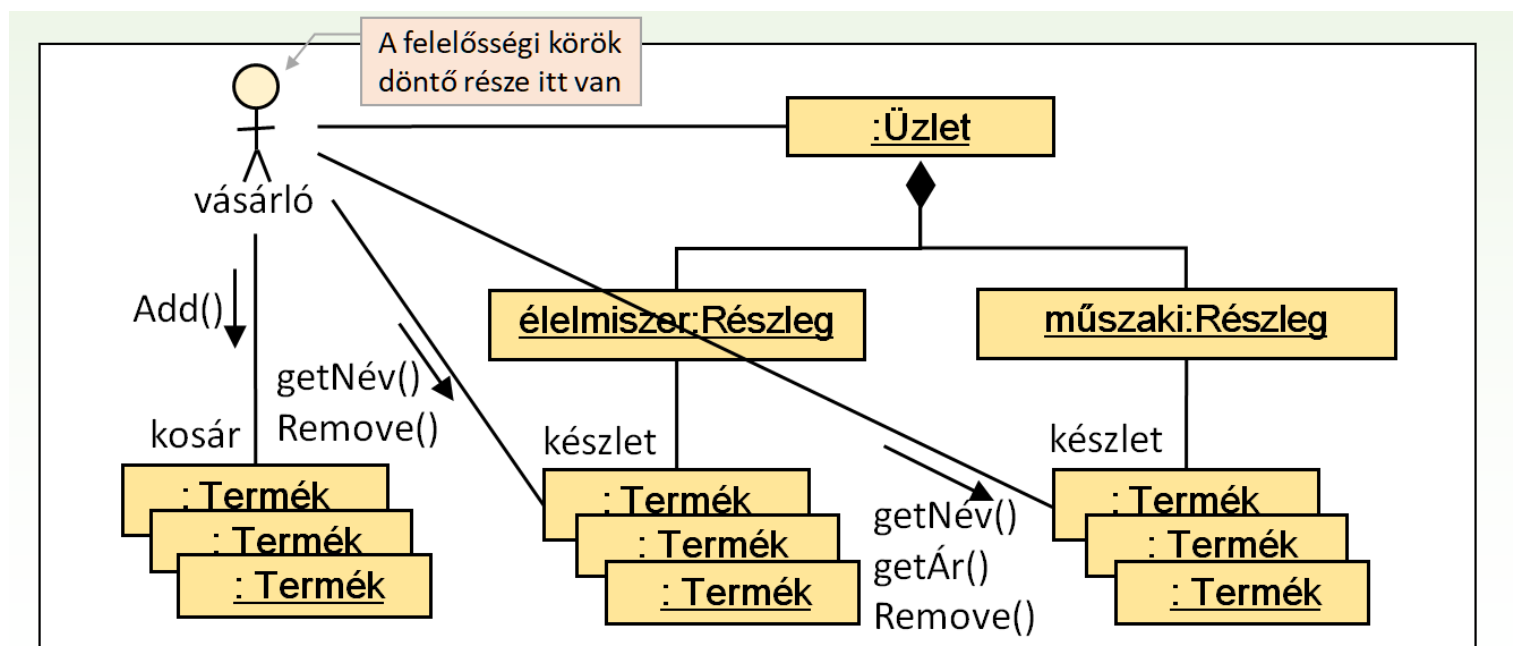


Az objektum diagram nemcsak az objektumokat, hanem azok kapcsolatrendszerét is ábrázolja. A feladat szövegéből az objektumok adattagjaira is következtethetünk.



Azt, hogy egy-egy tevékenység melyik objektum felelősségi körébe tartozik, azaz melyik objektum metódusa lesz.

Kirajzolódnak a felelősségi körök: milyen metódusokkal rendelkezzenek az objektumok

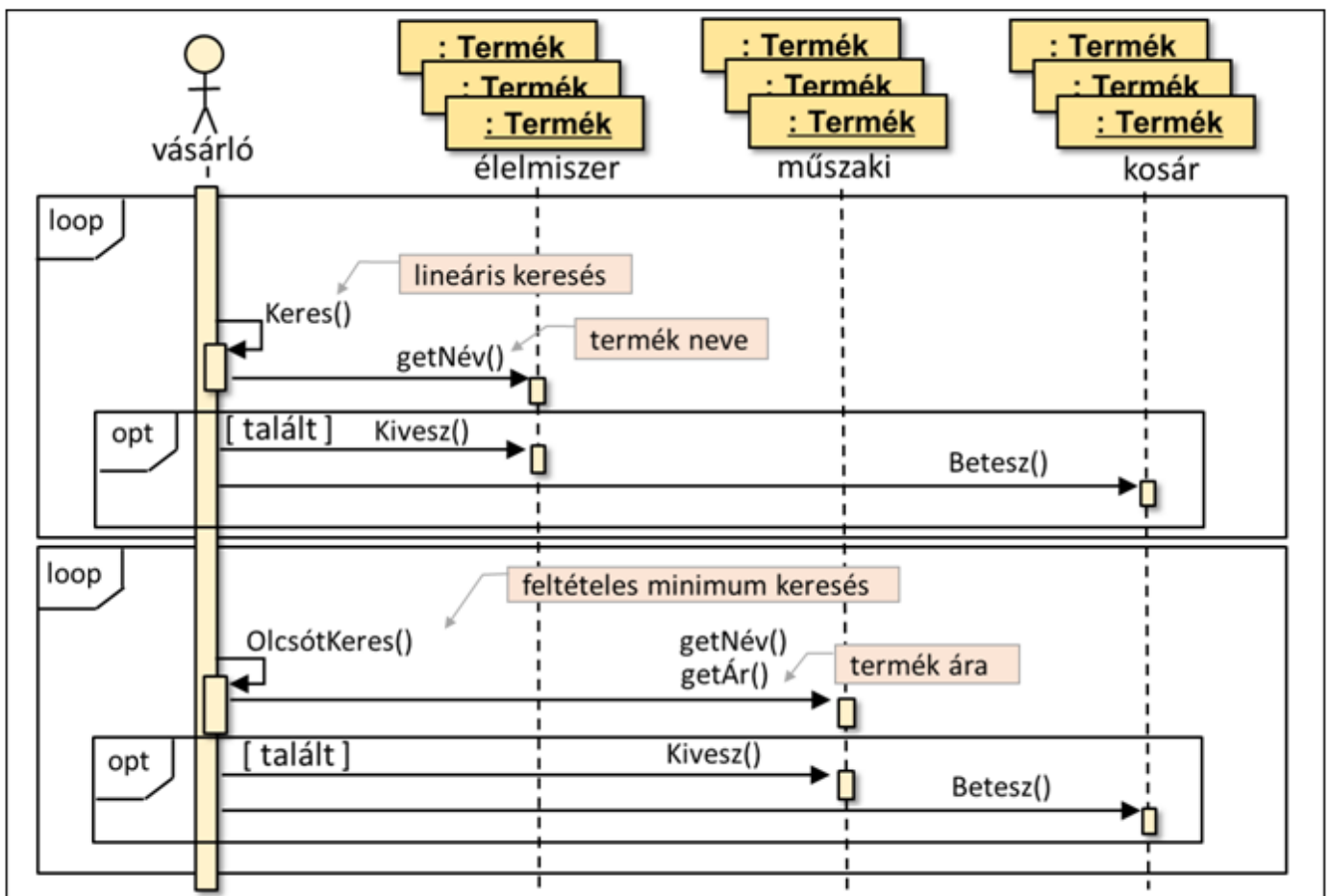


A vásárlás a vásárló metódusa.

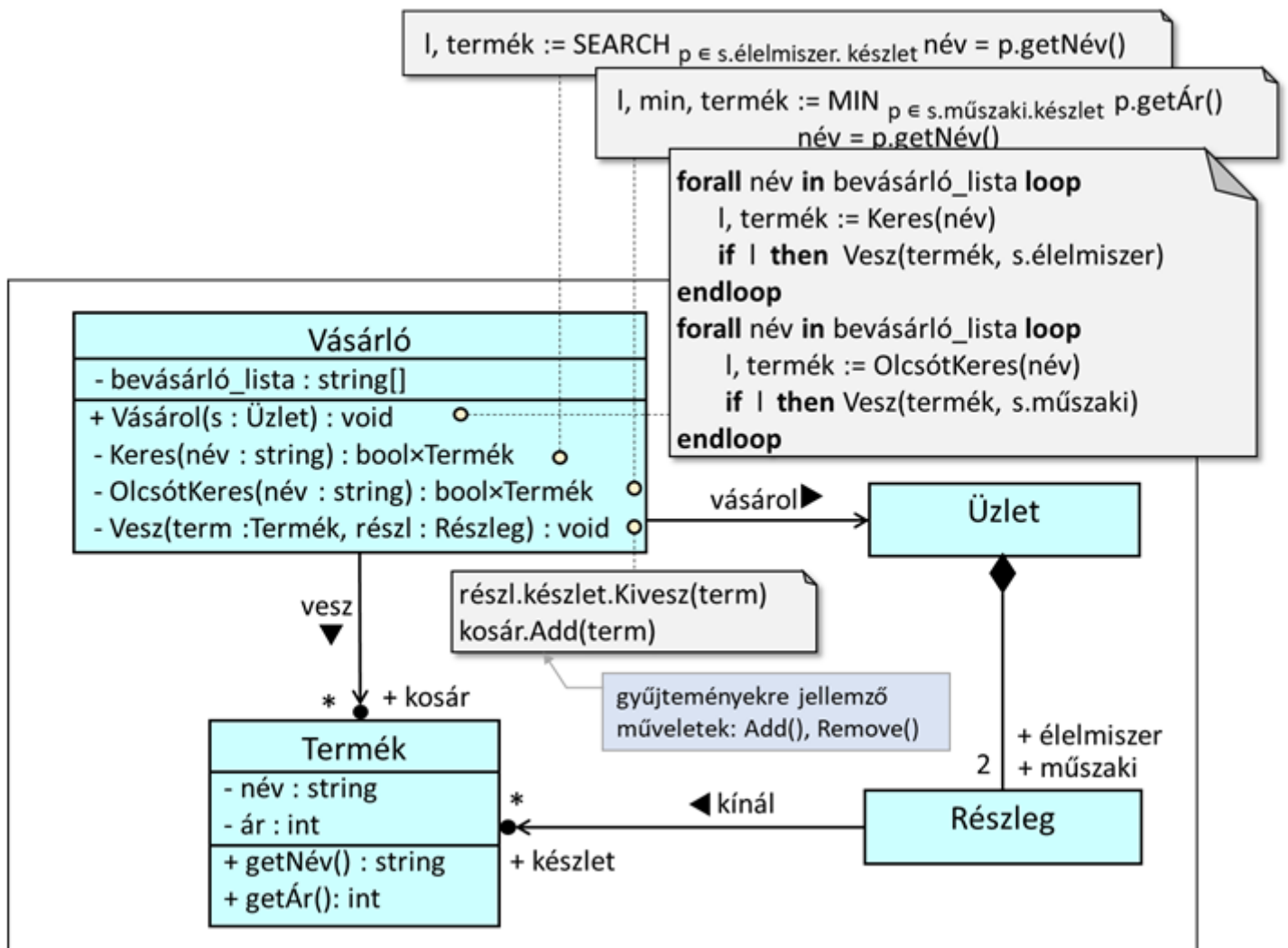
Ez két lépésből áll: az élelmiszer részlegen történő vásárlásból, és a műszaki részlegen történő vásárlásból.

Mindkét részlegen a bevásárló listán szereplő összes terméket meg kell keresni: az élelmiszer részlegen az első találatig (lineáris keresés), a műszaki részlegen a legjobb árral rendelkezőt (feltételes maximum keresés).

A vásárlás folyamatát a szekvencia diagram mutatja meg.



A részlegek árukészletei, és a vásárló kosara is gyűjtemények. Egy gyűjteménybe történő beillesztést most a Betesz(), egy elem eltávolítását a Kivesz() metódussal jelöljük.



Kialakultak az objektumok osztályai az adattagokkal és a metódusokkal.

Az objektumok kapcsolatainak osztály diagram szintű ábrázolásában * multiplicitással jelöljük a gyűjteményeket.

Például a Részleg-Termék kapcsolatban a * multiplicitás szerepneve a készlet.

Ez azt jelenti, hogy minden részleg rendelkezik egy ilyen készlet adattaggal, amely a részlegen árusított termékek gyűjteményére hivatkozik.

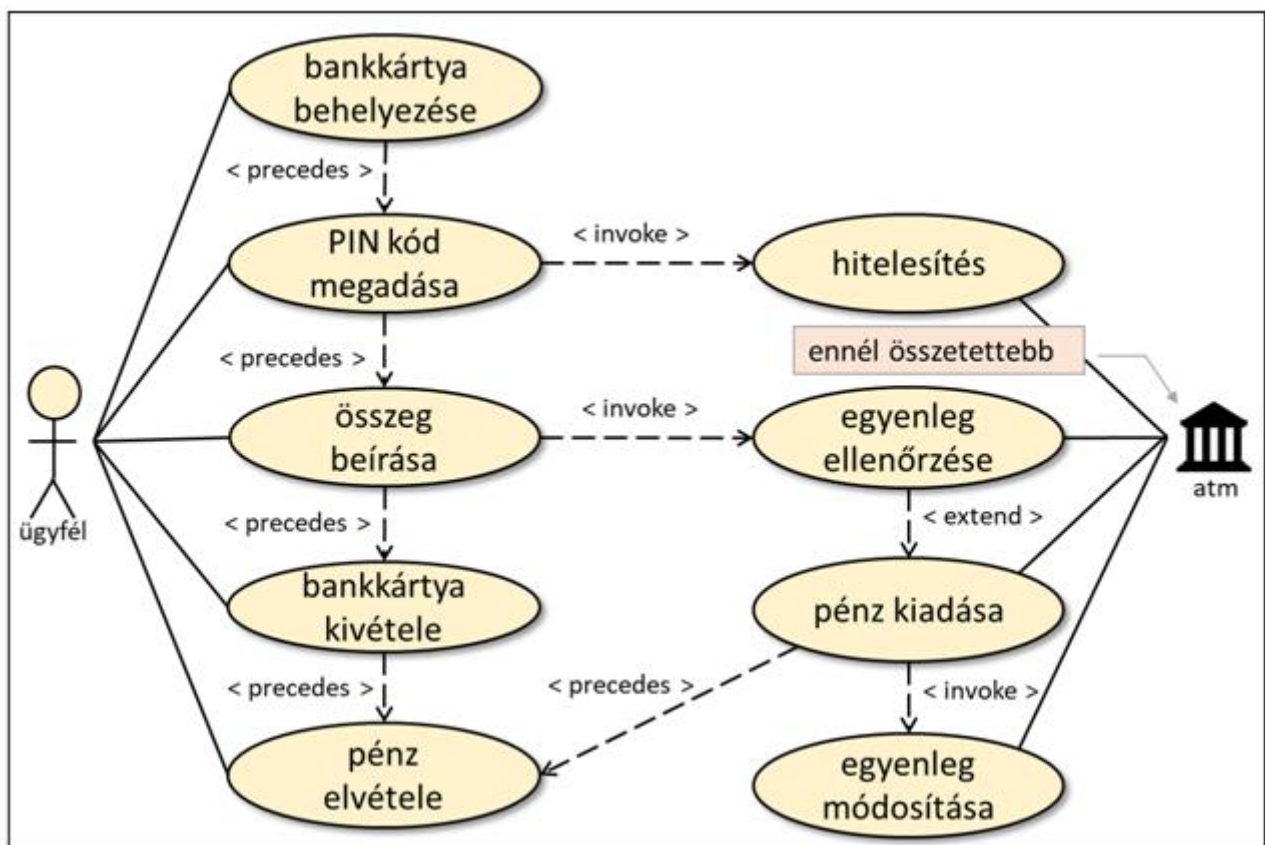
Erre alkalmazhatóak a gyűjtemény műveletei.

Hasonló mondható el a Vásárló-Termék kapcsolatban a * multiplicitású kosár szerepéről.

2.

- Egy ATM automatánál az ügyfelek sorban állnak, hogy pénzt vehessenek fel.
- Az ügyfelek rendelkeznek bankkártyákkal. Egy bankkártya egy bankszámlához tartozik, és van egy PIN kódja.
- Egy ügyfél odaadja a bankkártyáját és a PIN kódját az ATM-nek, és az ellenőrzi ennek hitelességét.
- Ezután az ügyfél megadja a felvenni kívánt összeget. Ha az összeget levonva az ügyfél számlájának egyenlegéből az továbbra is pozitív marad, akkor az ATM kiadja az összeget.
- Ehhez a folyamathoz az ATM egy központon keresztül lekéri az ügyfél számlaegyenlegét a kártyájának adatai alapján, illetve elküld a egy jelentést a lebonyolított tranzakcióról az ügyfél bankjának, amely ezalapján leveszi az összeget az ügyfél számlájáról.

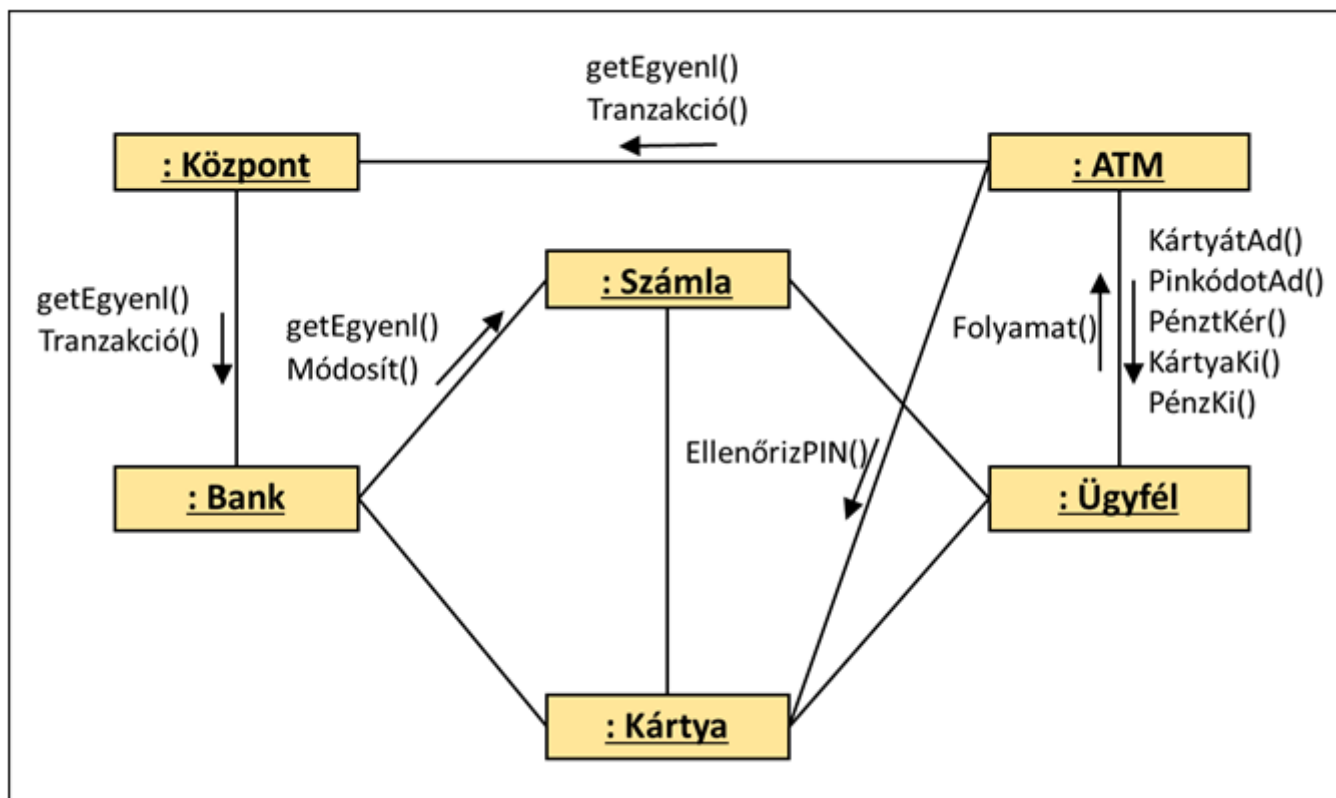
A használati esetek felsorolásánál még nincs jelentősége annak, hogy az atm mögött egy központ van, amely a bankokkal áll kapcsolatban, ahol az ügyfélszámlákat vezetik.



A pénzfelvételt az atm.Folyamat() metódus hívása váltja ki, amelyet a soron következő ügyfél Pénzfelvétel() metódusa hív meg.

A pénzfelvétel egyes lépései különböző objektumok felelősségi köréhez tartozó metódusok végzik.

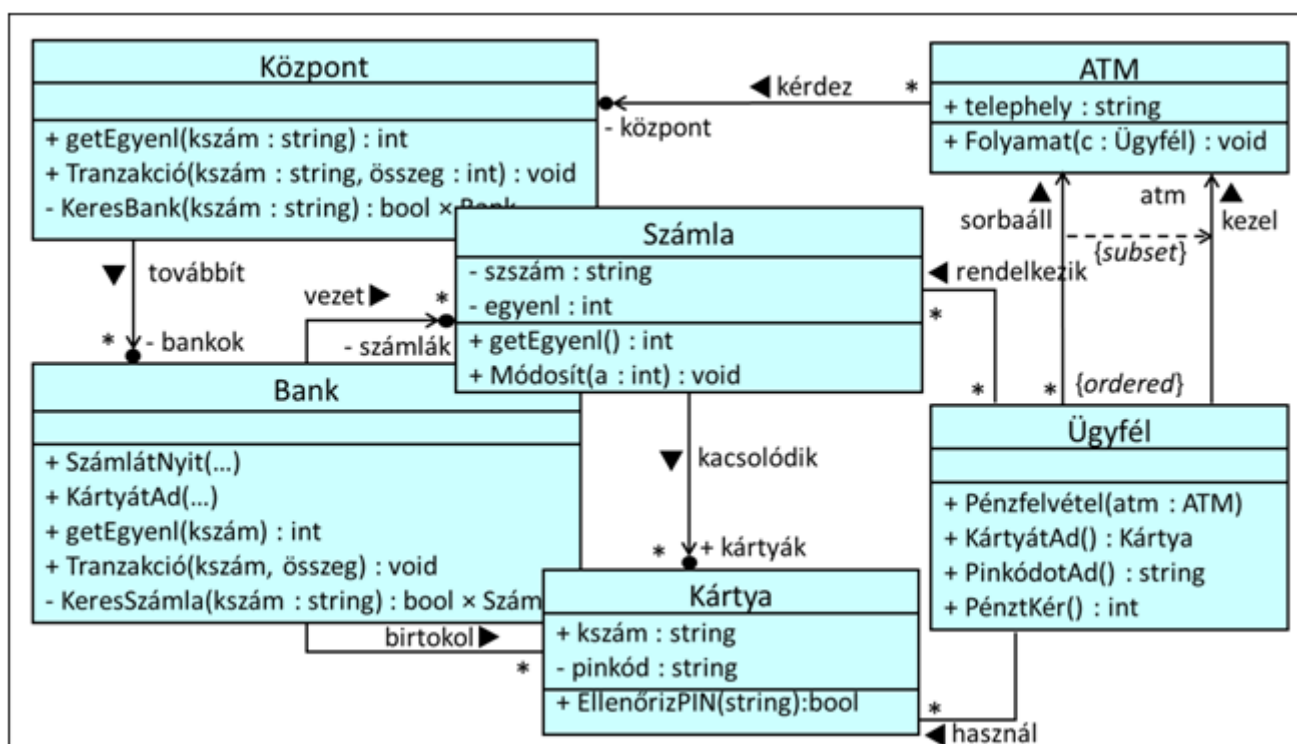
Ezek az objektumok: ügyfél, atm, központ, bank, számla, bankkártya.



Érdekes része a megoldásnak az, ahogyan az atm hozzájut az ügyfél számlaegyenlegéhez, illetve ahogyan végrehajtja a pénzfelvételt kiegyenlítő banki tranzakciót.

Mindkét tevékenység több objektumon keresztül küldött üzenet-lánc formájában, a kérdez, továbbít, vezet, és kapcsolódik asszociációk általi kirajzolt navigációs útvonalon valósul meg,

ezért a navigációt itt érdemes szerepnevekkel és azok tulajdonosának megjelölésével segíteni.

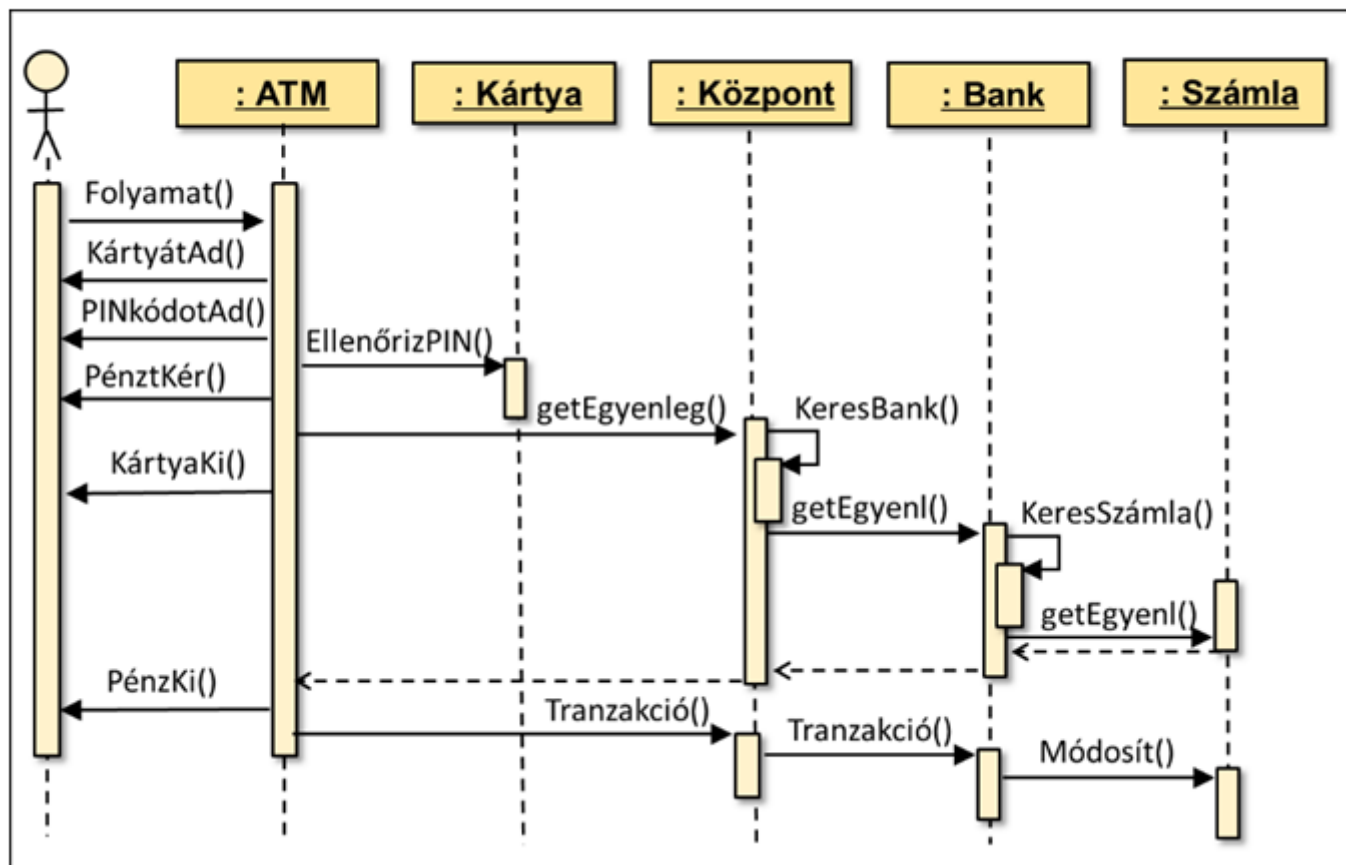


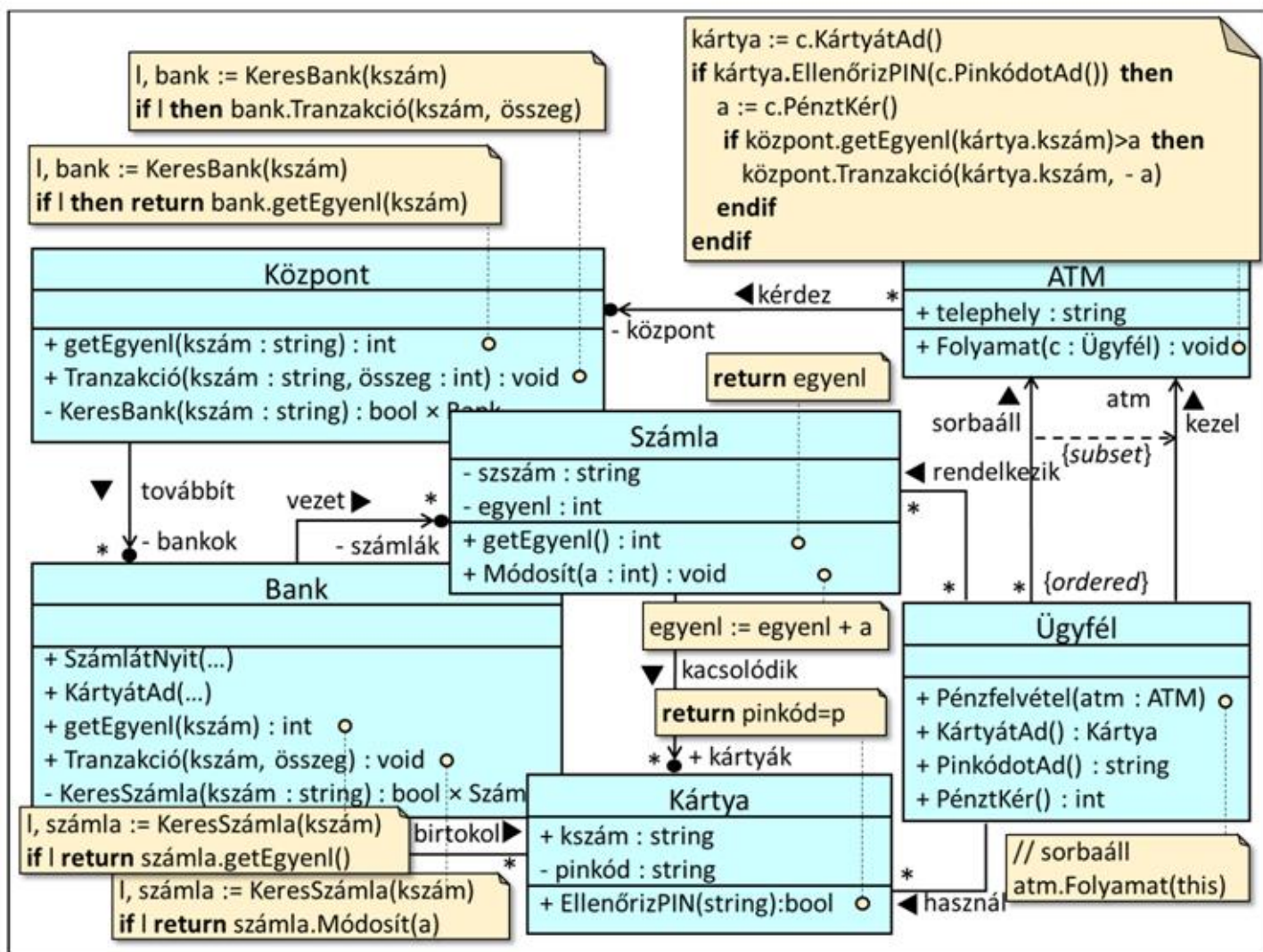
Különféle gettereket és egyéb segéd metódusokat is bevezethetünk, amelyek a folyamat átláthatóságát szolgálják.

Például a központnak a `KeresBank()` metódusa a központnál nyilvántartott bankok közül választja ki az ügyfél bankját a bankkártya számából kinyert bankkód alapján.

A banknak a `KeresSzámla()` metódusa a banknál vezetett számlák között keresi meg azt, amelyhez az adott bankkártya tartozik.

A pénzfelvétel lépéseinek sorrendjét az alábbi szekvencia diagram írja le:





3.

Egy csomag kiszállító futár egy telephelyről hordja szét a megrendelt csomagokat különböző benzinkutakhoz telepített PickPack pontokra.

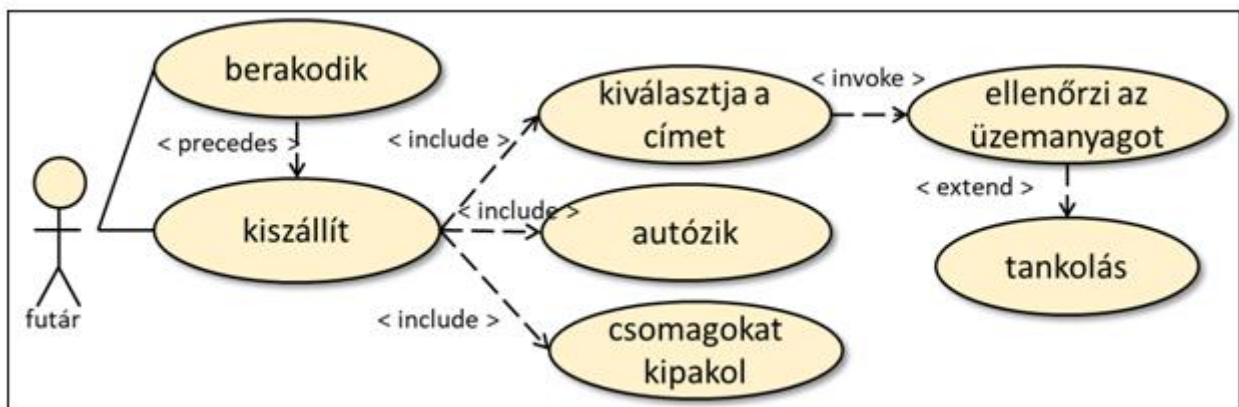
Tehát minden kiszállítási címen tankolni is lehet.

A csomagokra ráírták a kiszállítás címét, és ebből kiszámolható, hogy mekkora távolságot kell autóznia a futárnak aktuális tartózkodási helyétől a kiválasztott címig (km-ben).

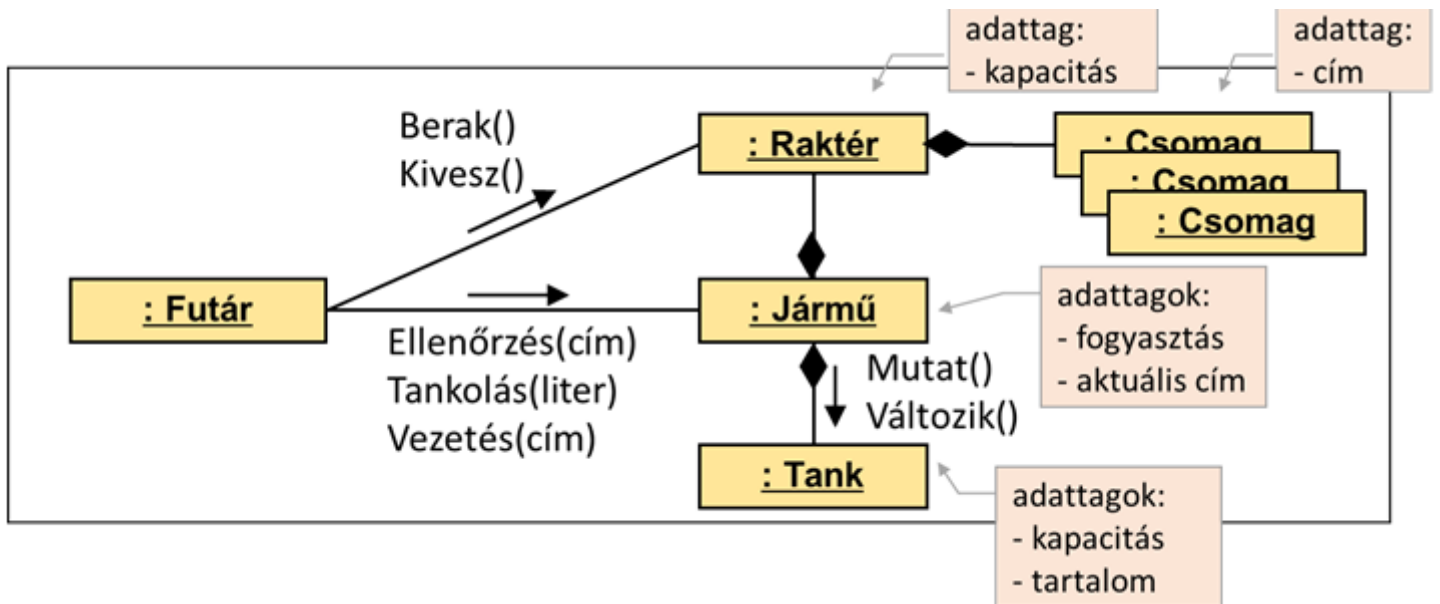
A futár bepakol annyi csomagot a járműve rakterébe, amennyit csak tud, majd a következők szerint jár el: kiválasztja a címét az egyik csomagjának (ha üres a rakodótér, akkor a telephelyének); ellenőrzi a benzinszintet, hogy elegendő-e a kiszállításhoz (ha nem, akkor tankol); elautózik a kiválasztott címre (ennek következményeképp csökken a benzinszint); majd kipakolja az adott címre küldött csomagokat.

A járműnek van rakodótere és egy benzintartálya. A rakodótér megadott számú csomagot képes tárolni. A tartályba a maximális benzinszint figyelembe vételével tankolhatunk. Ismert a jármű fogyasztása (liter/km mértékegységben).

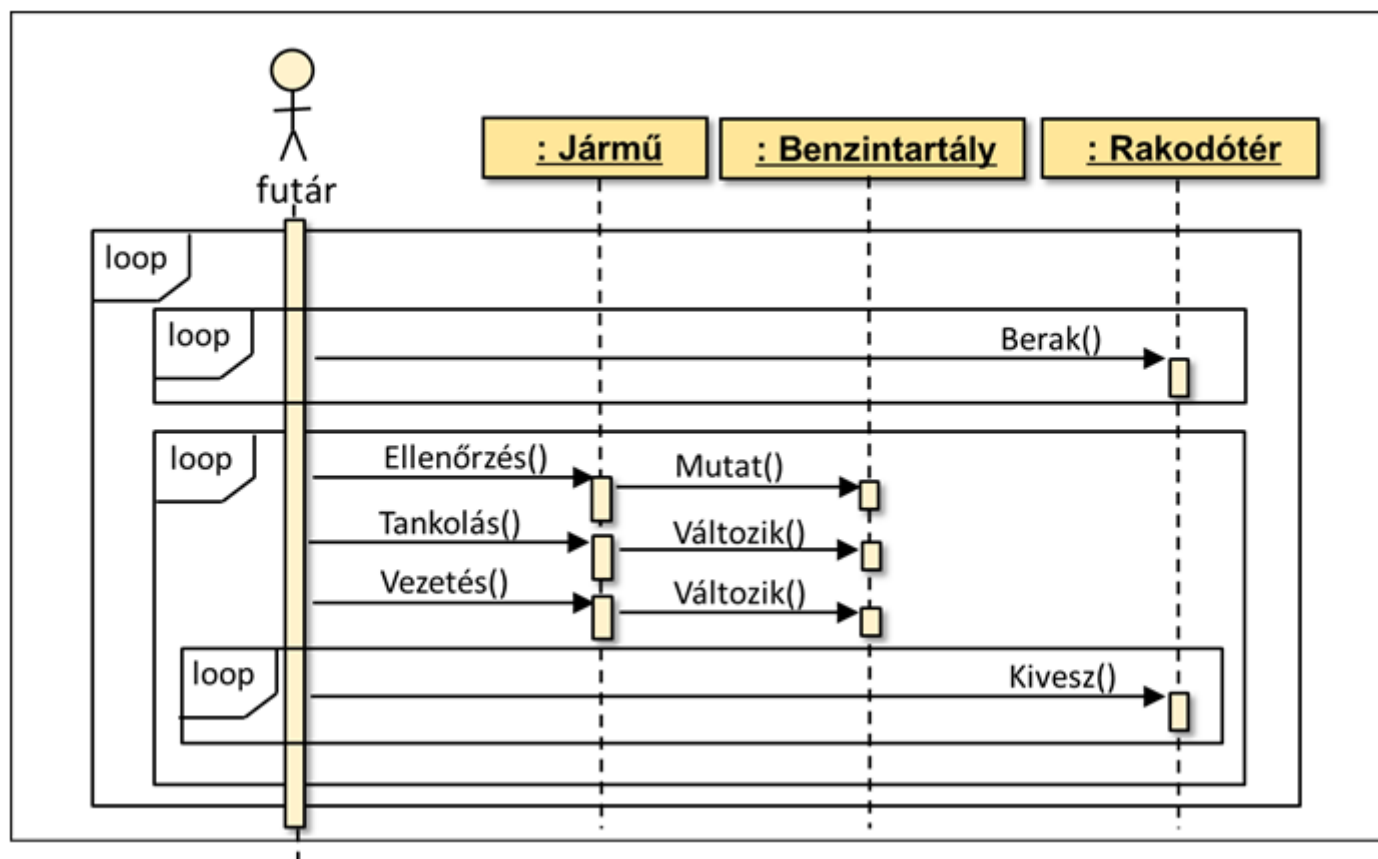
A használati eset diagram specialitása, hogy mindenféle kapcsolatra láthatunk benne példát.



A feladat megoldásához szükséges objektumokat viszonylag könnyű beazonosítani, a tevékenységeket pedig a kommunikációs diagram rendeli az objektumokhoz: ezáltal kialakítja az objektumok felelősségi köreit.

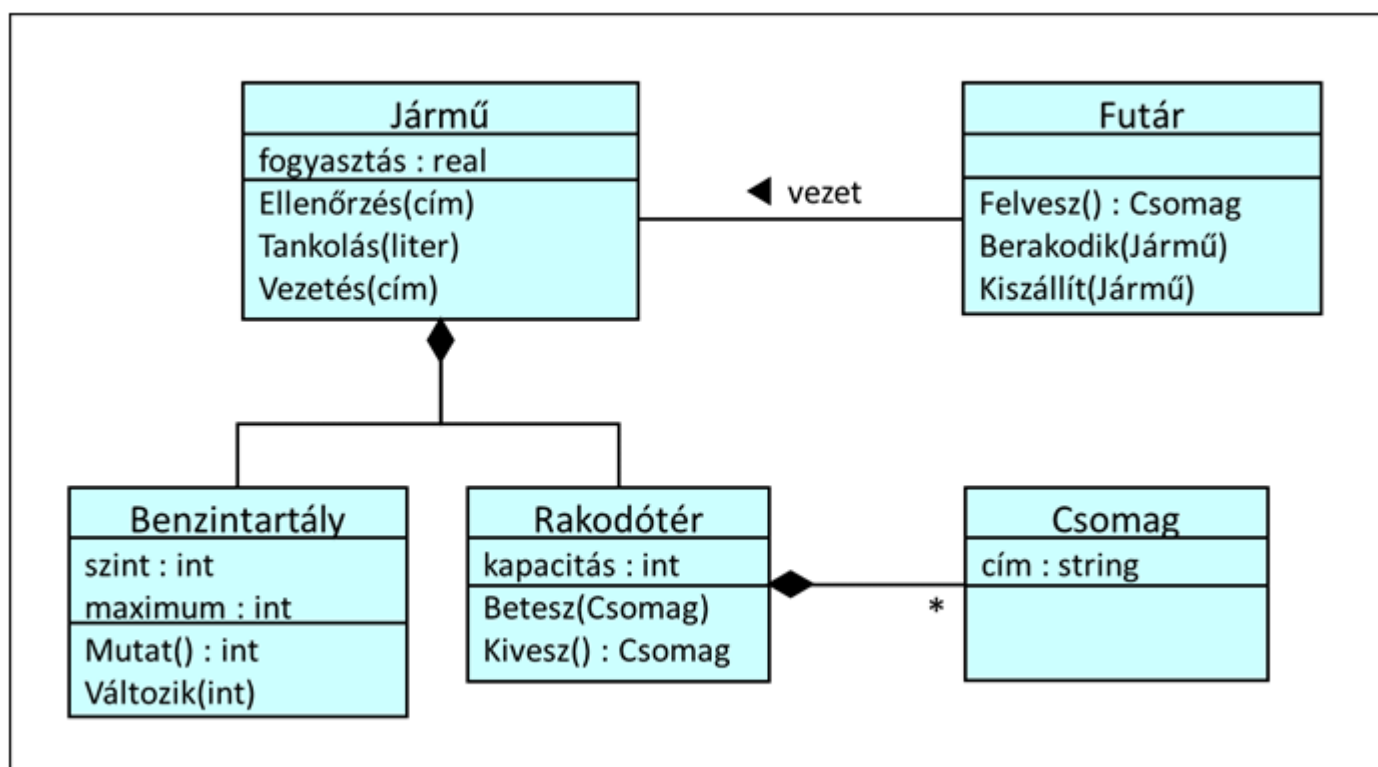


A szekvencia diagram pontosítja a metódusok végrehajtási sorrendjét, egyben ad egy lehetséges tesztelési forgatókönyvet is.



Az objektumokat, azok kapcsolatrendszerét, az objektumok adatait és metódusait legáttekinthetőbben az osztály diagram mutatja meg.

(A Járмű és Futár között lehetne függőség is, hiszen egy eseti kapcsolatot jön létre a futár és egy jármű között.)



Az elemzés szintű osztály diagram pontosításával: a kapcsolatokat nyilvántartó szerepnevek feltüntetésével, láthatósági tulajdonságok megadásával, a metódusok szignatúrájának és törzsének definiálásával kapjuk meg a tervezési szintű osztály diagramját.

