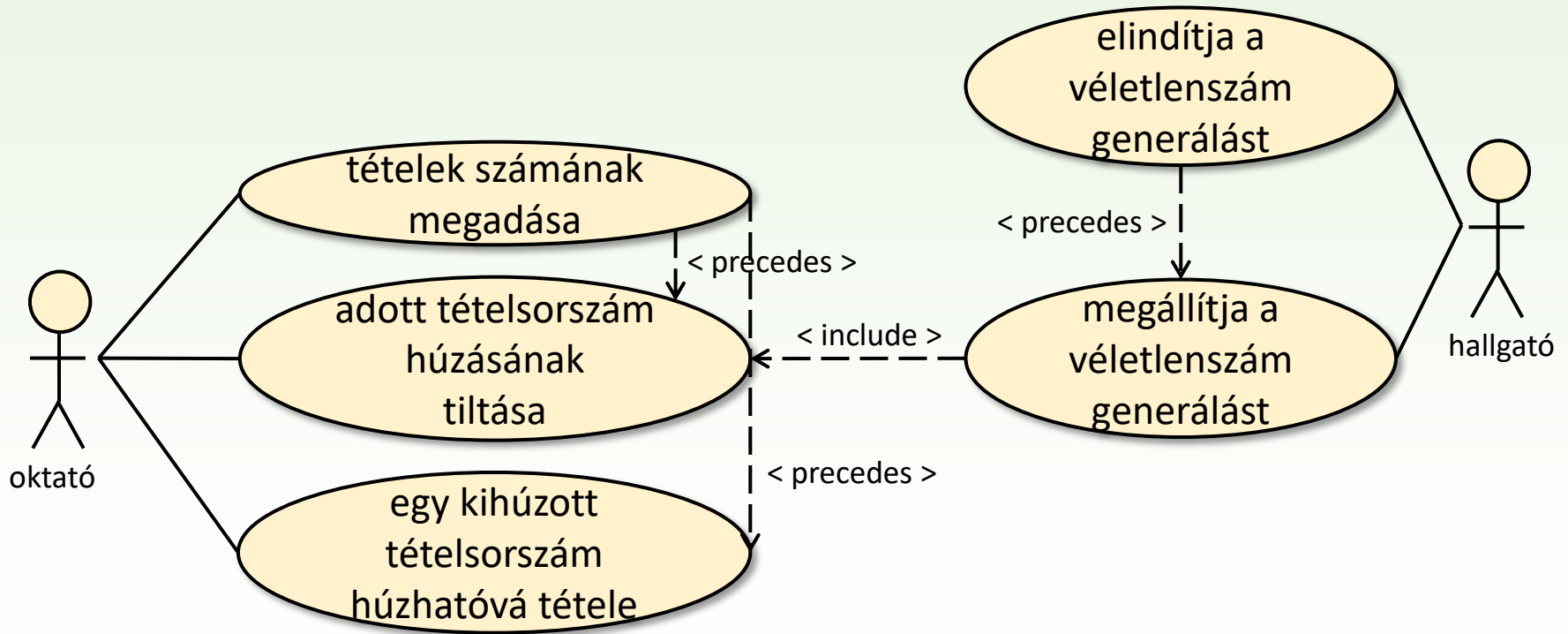
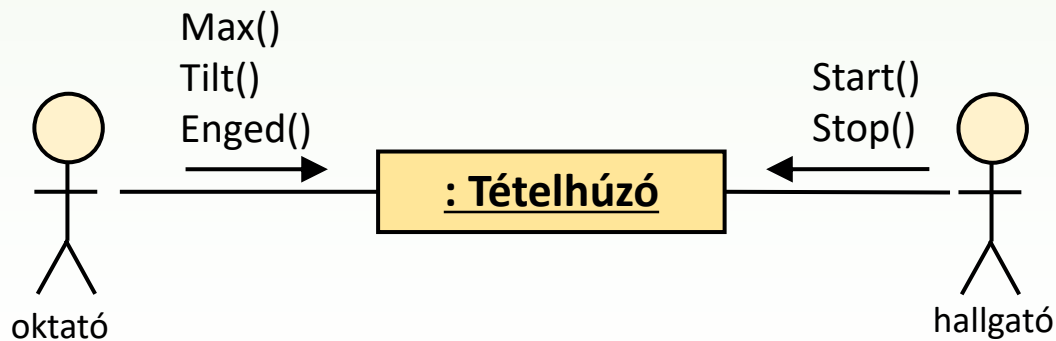
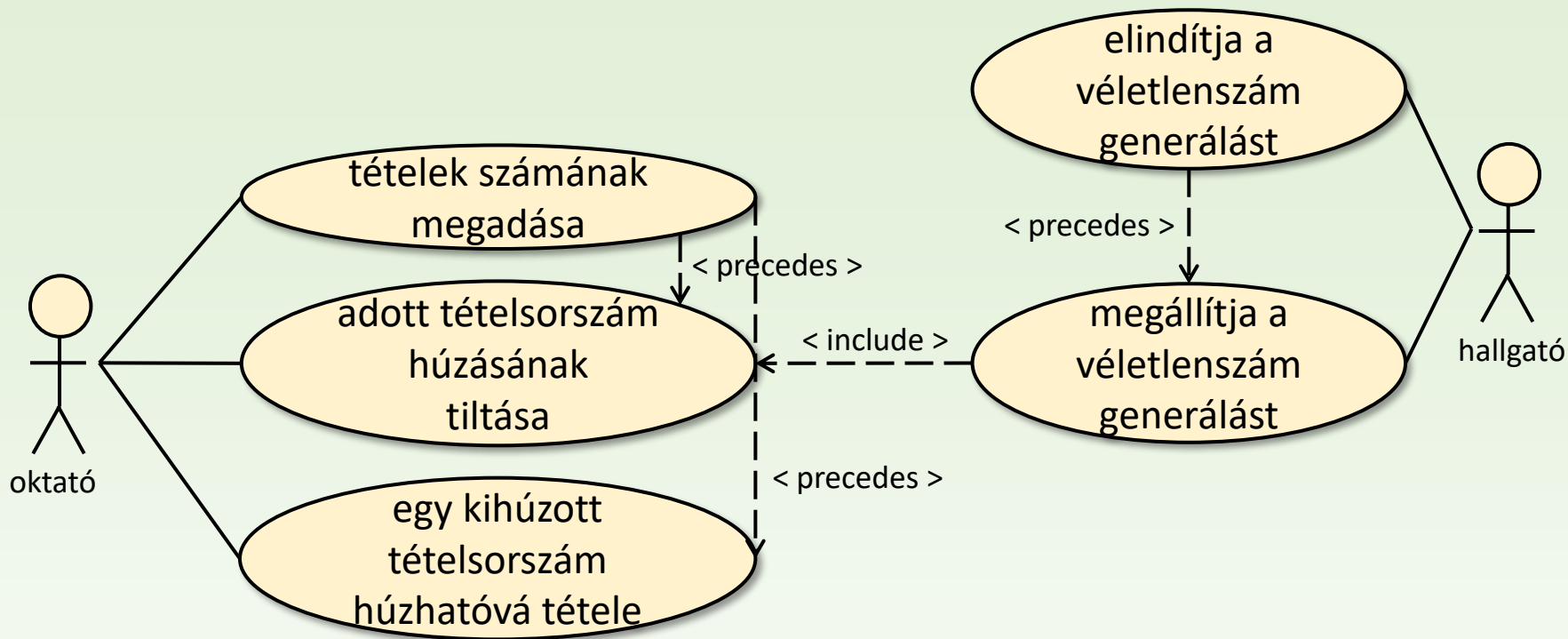


1. Egy vizsgán a hallgatók egy tételsorszám-generátor segítségével „húzzhatják” a tételüket. A vizsga előtt a vizsgáztató megadja a tételek számát, és minden tételnél be/kikapcsolhatja, hogy az húzható-e vagy sem. Egy hallgató a tételhúzásakor elindítja a generátort, és amikor leállítja, az a húzható tételek egyikének számát mutatja. Ez a tétel ezután már nem-húzható, kivéve, ha azt az oktató vissza nem teszi a húzhatók közé.



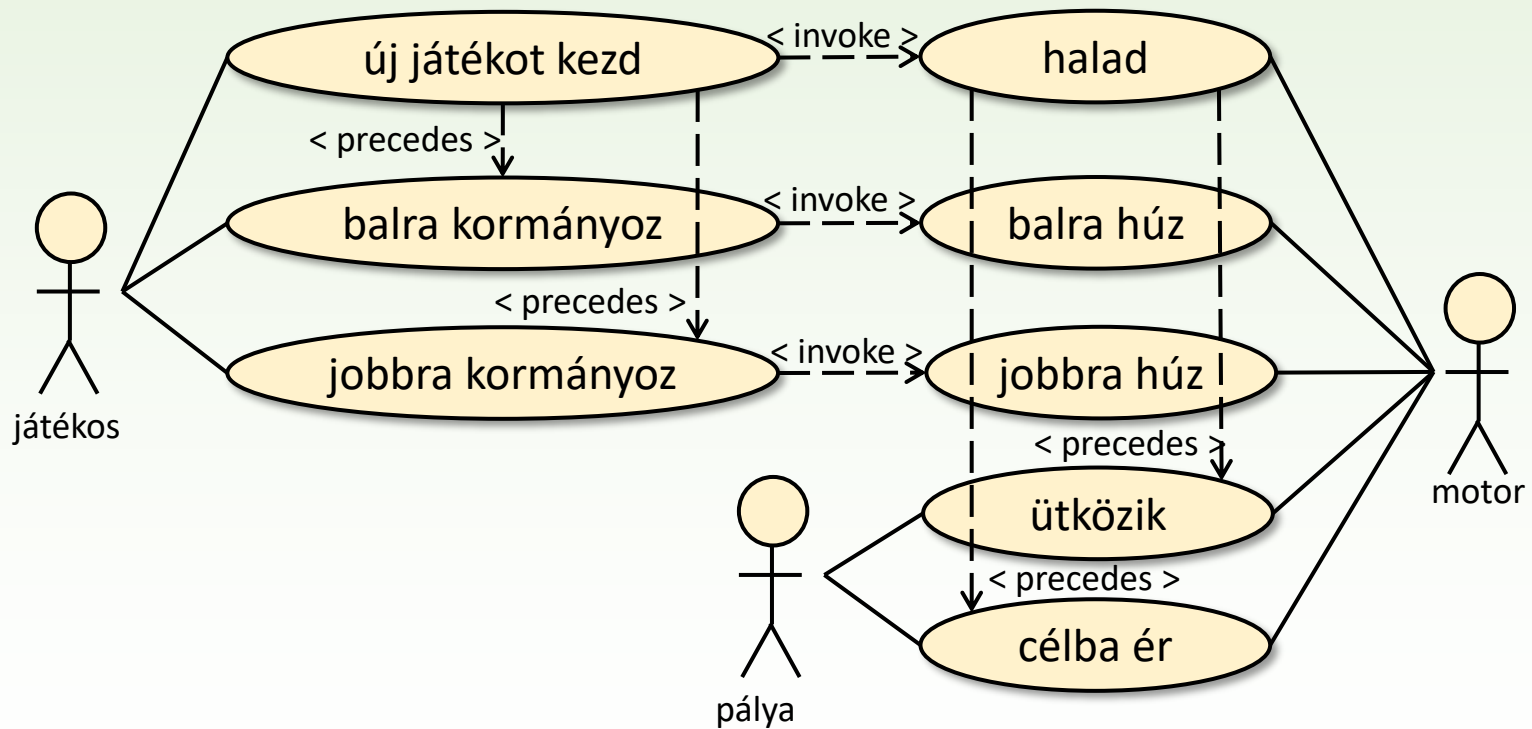
As oktató		Leírás
Tételek számának megadása	GIVEN	Az alkalmazás fut.
	WHEN	Tételek számát (\mathbb{N}^+) beírja.
	THEN	Az 1 .. tételszám közötti tételsorszámok „húzható” státuszúak.
Adott tételsorszám kihúzásának letiltása	GIVEN	Az alkalmazás fut, a tételek számát már megadtuk.
	WHEN	Az egyik (1 .. tételszám közötti) tételsorszámot megjelöli.
	THEN	A megjelölt tételsorszám „nem-húzható” státuszú lesz.
Adott tételsorszám kihúzásának engedése	GIVEN	Az alkalmazás fut, a tételek számát már megadtuk.
	WHEN	Az egyik nem-húzható tételsorszámot megjelöli.
	THEN	Az adott tételsorszám „húzható” státuszú lesz.

As hallgató		Leírás
elindítja a véletlenszám generálást	GIVEN	Az alkalmazás fut.
	WHEN	Véletlenszám generálást elindítja.
	THEN	A „húzható” tételsorszámok véletlen sorrendű, gyors váltakozása egy kijelzőn.
megállítja a véletlenszám generálást	GIVEN	Véletlenszám generálás már fut.
	WHEN	Véletlenszám generálást megállítja.
	THEN	A megállításkor kijelzett tételsorszám látható, státusza „kihúzott”-ra változik.



TételHúzó
<ul style="list-style-type: none"> - max : nat - húzható : nat[]
<ul style="list-style-type: none"> + Max(n:nat) + Tilt(i:nat) + Enged(i:nat) + Start() + Stop() : nat

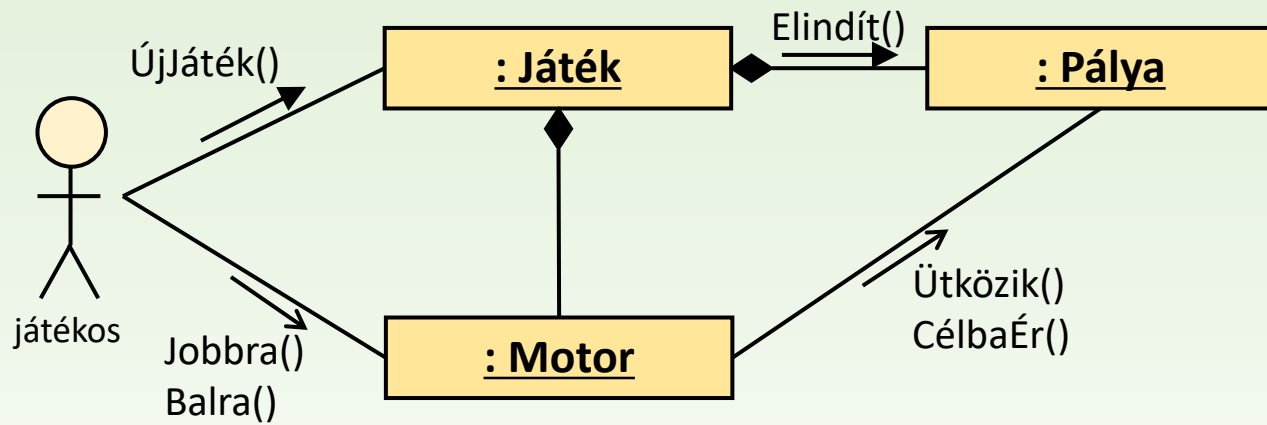
2. Egy motoros játékban a játékos feladata, hogy minél messzebb jusson a pályán a motorjával, elkerülve az akadályokat. Miután új játékot kezd, az akadályokat kerülgetve balra, illetve jobbra húzhat a motorjával addig, amíg akadálnak nem ütközik vagy célba nem ér.



As játékos		Leírás
Új játék	GIVEN	Az alkalmazást telepítették, elindították.
	WHEN	A felhasználó új játékot indít.
	THEN	A motor halad.
Balra / Jobbra	GIVEN	A motor halad.
	WHEN	Balra/Jobbra húzás billentyűjét lenyomták.

As motor		Leírás
Halad	GIVEN	A motort elindították.
	WHEN	Nincs semmilyen beavatkozás.
	THEN	Látszik a motor felülnézeti képe a képernyő közepén, amely alatt az akadályokat is tartalmazó útpálya egyenletes sebességgel hátrafelé úszik.
Balra / Jobbra húz	GIVEN	A motor halad.
	WHEN	A felhasználó a balra/jobbra húzás billentyűjét lenyomva tartja.

As pálya		Leírás
Ütközik	GIVEN	A motor halad.
	WHEN	Feltűnik a motor képe alatt hátrafelé úszó pályán egy olyan akadály, amelyre ráfut a motor.
	THEN	A motor nem halad tovább, üzenet jelenik meg az ütközésről.
Célba ér	GIVEN	A motor halad.
	WHEN	Feltűnik a motor képe alatt hátrafelé úszó pálya vége, amelyre ráfut a motor.
	THEN	A motor nem halad tovább, üzenet jelenik meg a célba érésről.

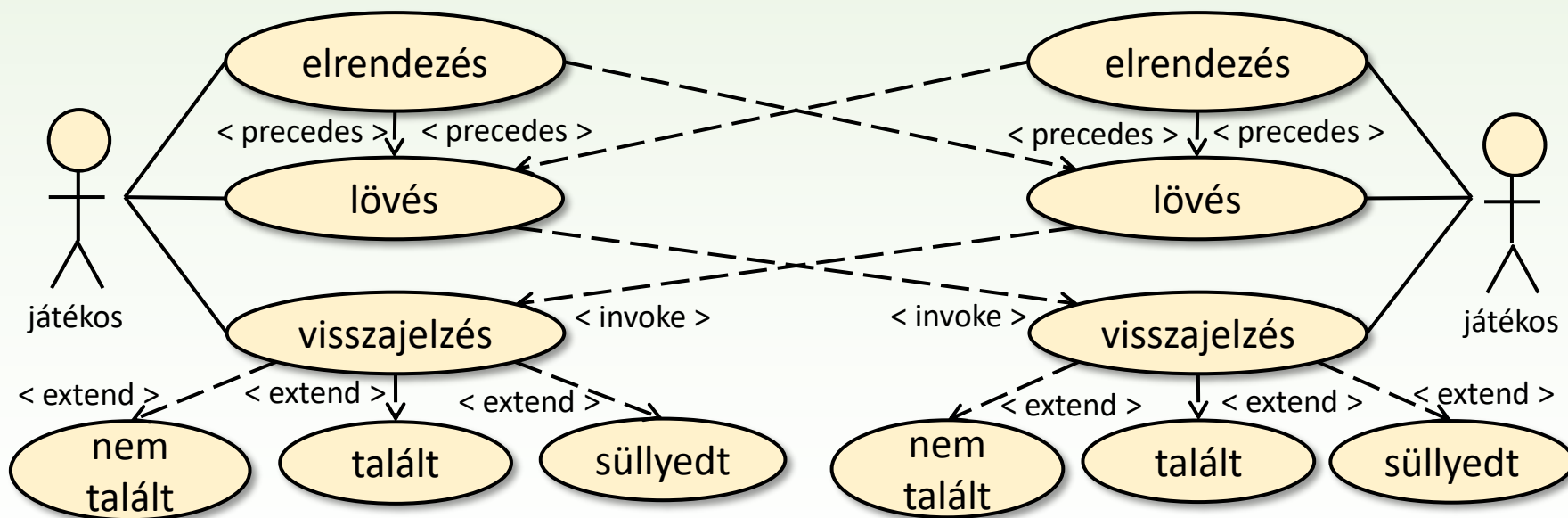


Játék
- pálya : Pálya
- motor : Motor
+ ÚjJáték()

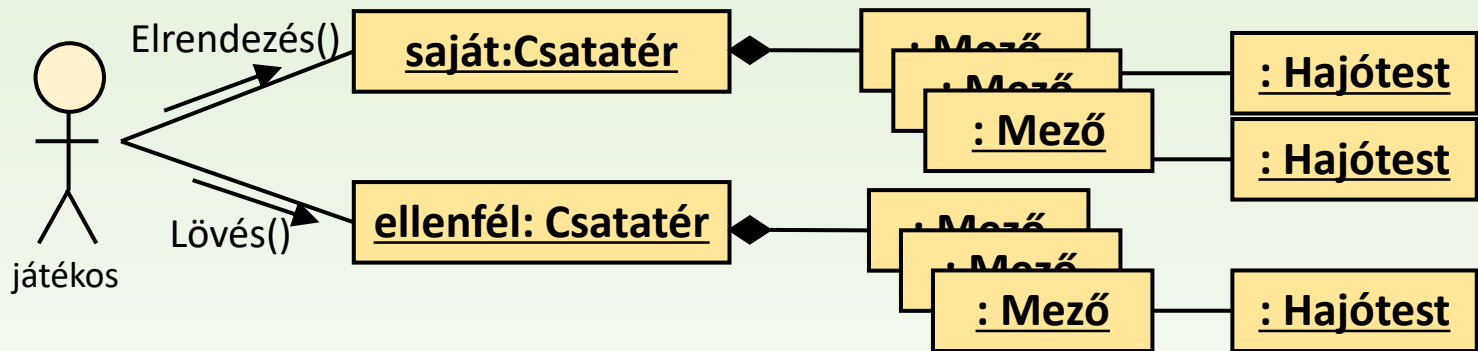
Motor
- poz : nat
+ Jobbra()
+ Balra()

Pálya
+ Ütközik(m:Motor)
+ CélbaÉr(m:Motor)

3. A torpedó játékot két játékos játssza. Először mindketten elhelyezik hajóikat a saját négyzetrácsos táblájukon, amit a másik játékos nem láthat. Egy hajó a tábla egy vagy több szomszédos mezőjét foglalja el. Ha egy hajó több mezőt is elfoglal, akkor a hajó által elfoglalt mindegyik mező legalább egy oldalával érintkezik a hajó által elfoglalt valamelyik másik mezővel. Ezután a játékosok felváltva lőnek az ellenfél táblájára annak egy megcélzott mezőjének megadásával. A lövés után a másik játékos visszajelez, hogy eltalálták-e egy hajóját vagy sem, illetve, hogy így elsüllyedt-e az eltalált hajó (azaz annak minden mezőjét már találat érte).

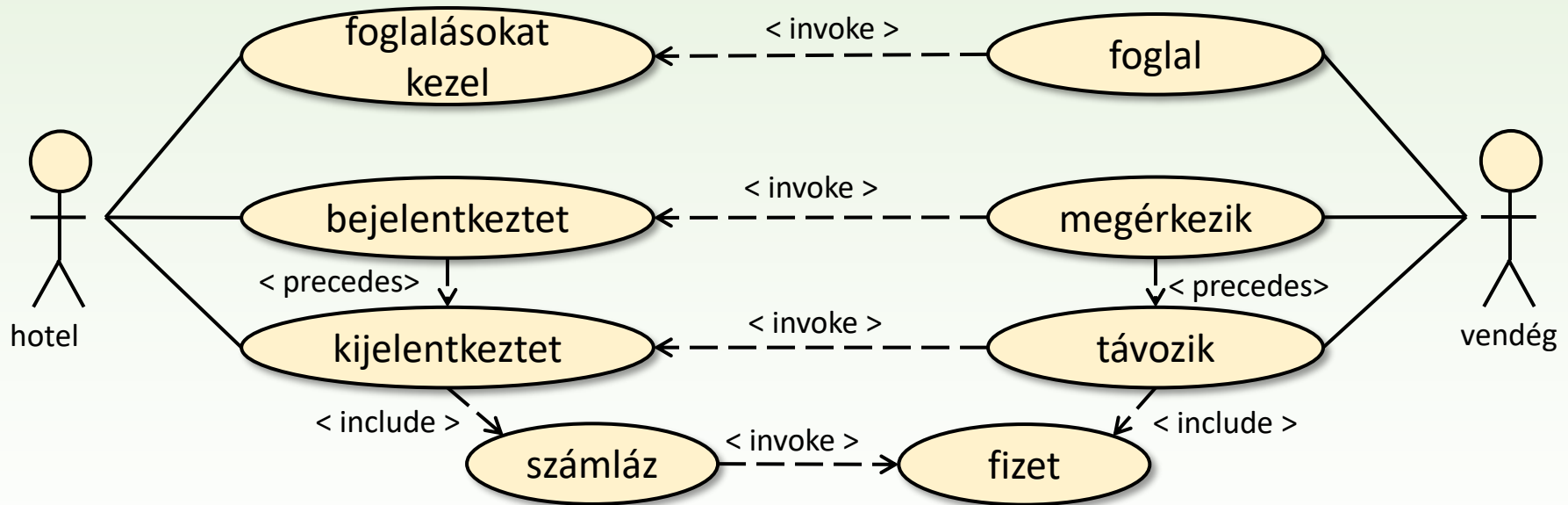


As játékos		Leírás
Elrendezés	GIVEN	Az alkalmazás elindult, létrejött a kapcsolat az ellenféllel, látja a saját üres tábláját, és a felhelyezhető hajókat.
	WHEN	Kiválaszt egy hajót és felhelyezi a tábla mezőire.
	THEN	A táblán látható a most felhelyezett hajó is.
Lövés	GIVEN	Az alkalmazás fut, a saját táblája mellett látja az ellenfél tábláját, és azon az általa korábban meglőtt mezőket, azt is, ha ezen egy ellenséges hajótest volt, és azt is, ha az egymáshoz kapcsolódó meglőtt ellenséges hajótestek egy teljes hajót tesznek ki, azaz a hajó elsüllyedt.
	WHEN	Lő az ellenfél táblájának egyik mezőjére.
	THEN	A megcélzott mező azonosítója eljut az ellenfélhez, aki visszajelez.
Visszajelzés	GIVEN	Az alkalmazás fut, a saját tábláján látja az ellenfél korábbi lövéseit, azokat is, amelyek egy hajótestet találtak el.
	WHEN	Az ellenfél lő a saját tábla egy mezőjére.
	THEN	Ha az eltalált mező üres, akkor „nincs találat” választ adja; ha azon egy hajótest is áll, akkor „talált” választ adja; ha ez a hajótest már az utolsó el nem talált hajóteste volt egyik hajójának, akkor a „talált, süllyedt” választ adja.



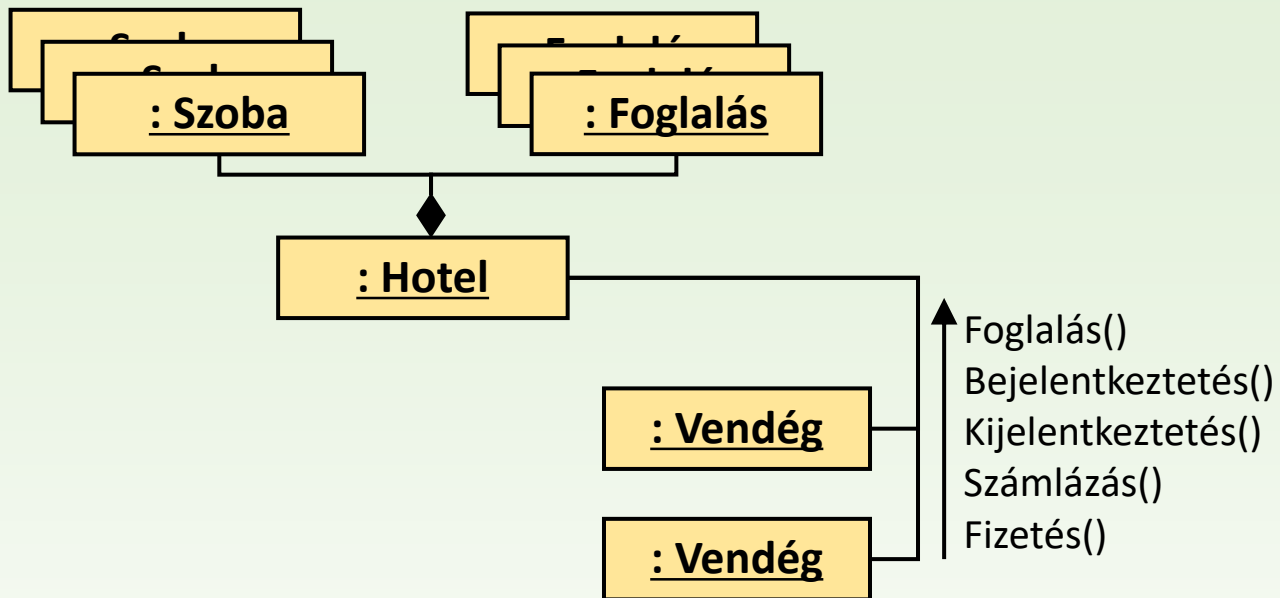
Csatatér
- terep : Mező[][]
+ Elrendezés(...)
+ Lövés(x,y:nat):Válasz

4. Egy hotelben a vendégek előre foglalhatnak szobát, de a szabad szobákat foglalás nélkül is kivehetik. Amikor megérkeznek a recepció segítségével jelentkezhetnek be; távozáskor a recepció jelentkezteti ki őket, és ő készíti el a számlát is, amit ki kell fizetniük.



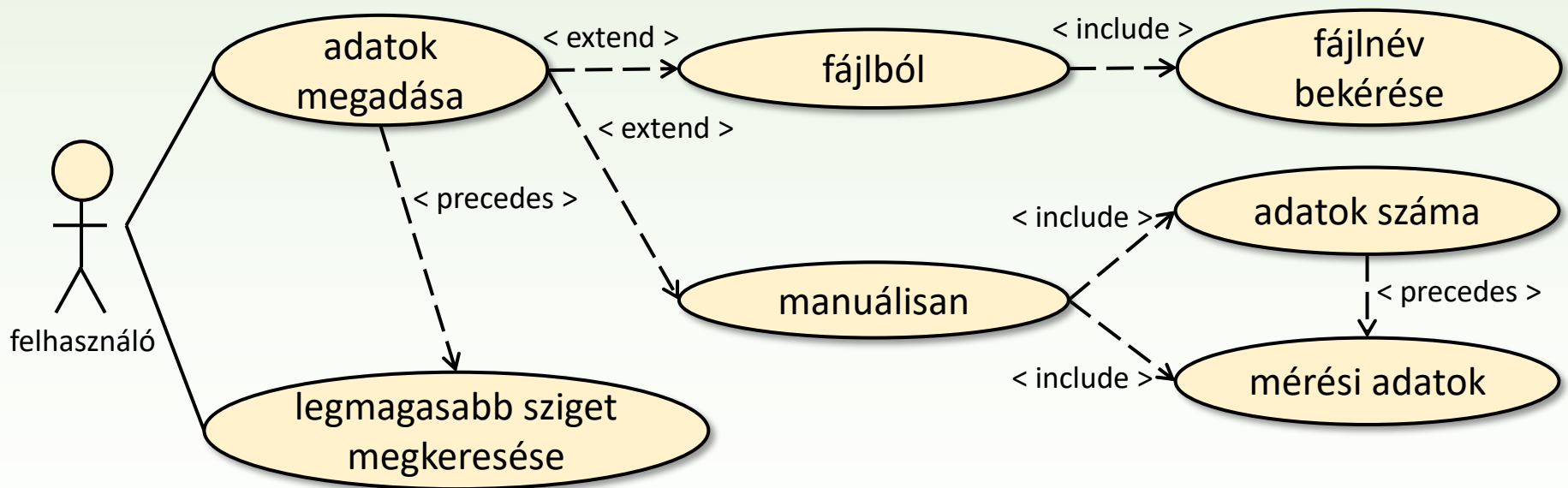
As vendég		Leírás
Foglalás (sikeres)	GIVEN	adott a lefoglalni kívánt időpont és szoba
	WHEN	a vendég foglalását ki lehet elégíteni
	THEN	létrejön a foglalás, értesítjük a vendéget
Foglalás (sikertelen)	GIVEN	adott a lefoglalni kívánt időpont és szoba
	WHEN	a vendég foglalását nem lehet elégíteni
	THEN	nem jön létre foglalás, értesítjük a vendéget
Fizetés (sikeres)	GIVEN	a vendég távozni akar
	WHEN	adott a szállodában igénybe vett szolgáltatásokról készült számla
	THEN	a vendég fizet

As hotel		Leírás
Bejelentkezés (sikeres)	GIVEN	a vendég szállást akar
	WHEN	van foglalása vagy van szabad szoba
	THEN	a vendég megkapja a kulcsát, a szálloda használati információkat
Bejelentkezés (sikertelen)	GIVEN	a vendég szállást akar
	WHEN	nincs foglalása és nincs szabad szoba
	THEN	a vendéget udvariasan elküldjük
Kijelentkezés	GIVEN	a vendég a hotelben lakik
	WHEN	a vendég távozni és fizetni akar
	THEN	számla készül, a vendég fizet, a szoba szabaddá (foglalhatóvá) válik

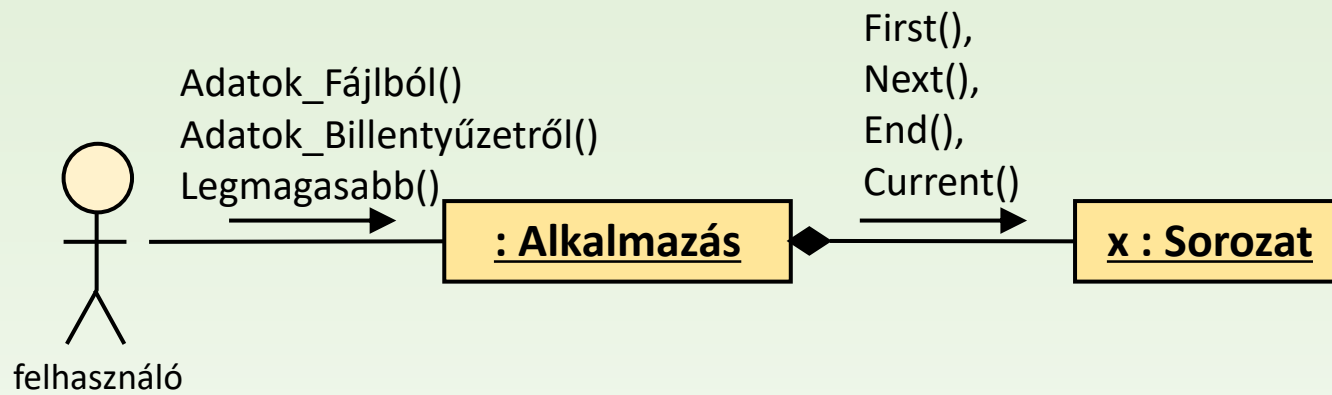


Hotel
- foglalások : Foglалás[] - szobák : Szoba[]
+ Foglалás() + Fizetés() + Bejelentkezés() + Kijelentkezés() + Számlázás()

5. Keressük a legmagasabb szigetet a Föld felszín tengerszint feletti magasságának egy adott vonalban végzett mérései alapján. Ehhez először felhasználó vagy manuálisan adja meg adatokat, vagy egy ezeket tartalmazó fájlt tölt be. Előbbi esetben először megadja a mérési pontok számát, majd magukat a mért magasság értékeket. A bevitelt követően az alkalmazás kiírja a legmagasabb sziget sorszámát.



As felhasználó		Leírás
Adatok megadása fájlból	GIVEN	Adott egy szöveges állomány, amelyben szóközökkel elválasztva találjuk az egymás után mért tengerszint feletti magasságértékeket. Az alkalmazás elindult.
	WHEN	Fájlból való adatfeltöltést választjuk.
	THEN	Beolvassuk a fájl elemeit (N), amelyeket egy sorozatba fűzünk.
Adatok megadása billentyűzetről	GIVEN	Az alkalmazás elindult.
	WHEN	Billentyűzetről való adatfeltöltést választjuk.
	THEN	Beolvassuk először a mérési pontok számát (N), majd ennek megfelelő számú értéket (N), amelyeket egy sorozatba fűzünk.
Legmagasabb sziget, ha van	GIVEN	Feltöltöttünk egy számsorozatot a tengerszint feletti magasságokkal, amelyben van legalább egy pozitív szám.
	WHEN	Keressük a legmagasabb szigetet. A sziget a sorozat egy olyan pozitív értékekből álló szakasza, amelynek eleje vagy a sorozat eleje, vagy egy 0 értékű elem után következik; a vége pedig vagy a sorozat vége, vagy egy 0 értékű elem előtt van. A sziget magassága a legmagasabb értéke.
	THEN	A legmagasabb sziget sorszámának megadása
Legmagasabb sziget, ha nincs	GIVEN	Feltöltöttünk egy számsorozatot a tengerszint feletti magasságokkal, amelyben nincs pozitív szám.
	WHEN	Keressük a legmagasabb szigetet.
	THEN	Hiba üzenet arról, hogy nincs egyáltalán sziget.



Alkalmazás
- x : real[]
+ Adatok_Fájlból() + Adatok_Billentyűzetről() + Legmagasabb() : nat