

Készítette

Németh Anna Krisztina

E-mail: cuewak@inf.elte.hu

Csoportszám: 6

Feladat

Készítsünk programot, amellyel a következő játékot játszhatjuk. Adott egy $n \times n$ elemből álló játékpálya, amelyben Maci Lacival kell piknikkosarakra vadásznunk. A játékpályán az egyszerű mezők mellett elhelyezkednek akadályok (pl. fa), valamint piknikkosarak.

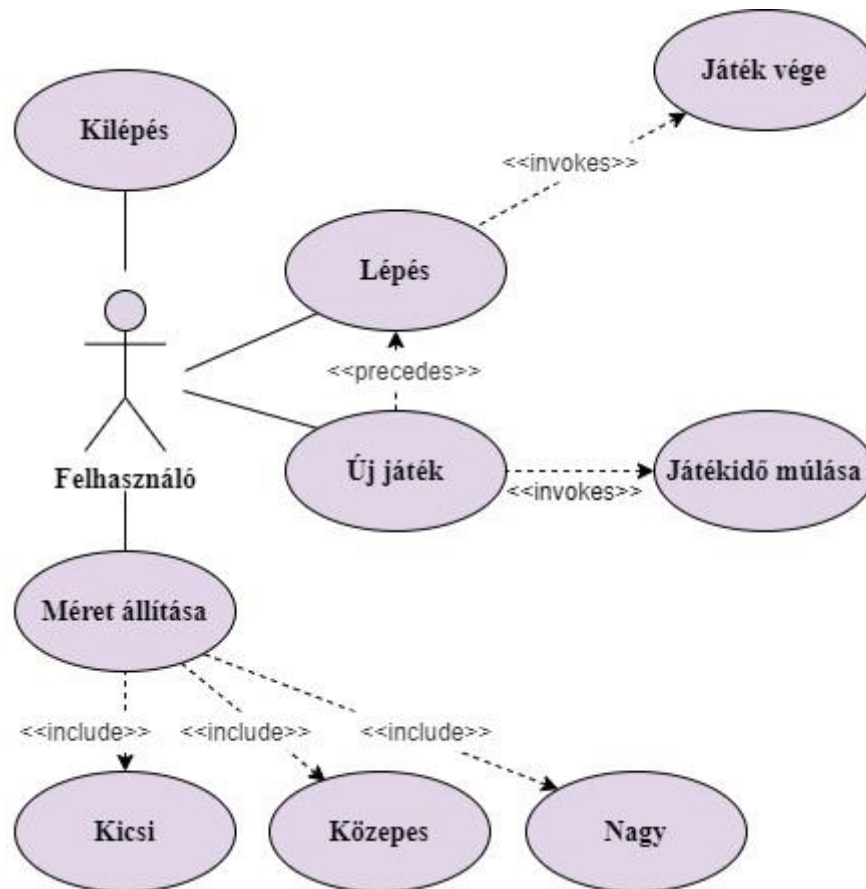
A játék célja, hogy a piknikkosarakat minél gyorsabban begyűjtsük. Az erdőben vadőrök is járőröznek, akik adott időközönként lépnek egy mezőt (vízszintesen, vagy függőlegesen). A járőrözés során egy megadott irányba haladnak egészen addig, amíg akadályba (vagy az erdő szélébe) nem ütköznek, ekkor megfordulnak, és visszafelé haladnak (tehát folyamatosan egy vonalban járőröznek). A vadőr járőrözés közben a vele szomszédos mezőket látja (átlósan is, azaz egy 3×3 -as négyzetet).

A játékos kezdetben a bal felső sarokban helyezkedik el, és vízszintesen, illetve függőlegesen mozoghat (egyesével) a pályán, a piknikkosárra való rálépéssel pedig felveheti azt. Ha Maci Lacit meglátja valamelyik vadőr, akkor a játékos veszít.

A pályák méretét, illetve felépítését (piknikkosarak, akadályok, vadőrök kezdőpozíciója) tároljuk fájlban. A program legalább 3 különböző méretű pályát tartalmazzon. A program biztosítson lehetőséget új játék kezdésére a pálya kiválasztásával, valamint játék szüneteltetésére (ekkor nem telik az idő, és nem léphet a játékos). Ismerje fel, ha vége a játéknak, és jelezze, győzött, vagy veszített a játékos. A program játék közben folyamatosan jelezze ki a játékidőt, valamint a megszerzett piknikkosarak számát.

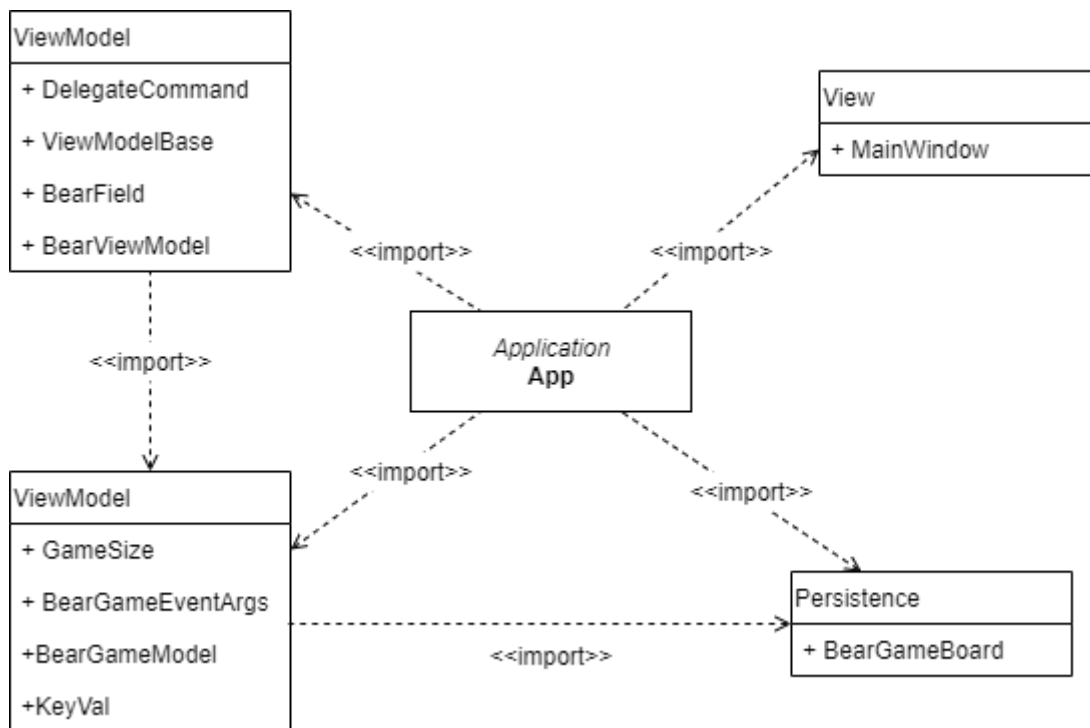
A feladat elemzése

- A játékot három mérettel játszhatjuk: kicsi (10×10), közepes (15×15) és nagy (20×20). A program induláskor közepes méretet állít be, és automatikusan új játékot indít.
- A feladatot egyablakos asztali alkalmazásként Windows Forms grafikus felülettel valósítjuk meg.
- Az ablakban elhelyezünk egy menüt a következő menüpontokkal: Fájl (Új játék, Kilépés), Beállítások (Kis pálya, Közepes pálya, Nagy pálya). Az ablak alján megjelenítünk egy státuszsort, amely az eltelt időt, illetve az összegyűjtött piknikkosarak számát jelzi.
- A játéktáblát egy nyomógombokból álló rács reprezentálja. A játékos a bal felső sarokból indul, és a WASD billentyűkkel tudja mozgatni Maci Laci karakterét. Csak olyan mezőre léphet, amin nincs akadály.
- A játék automatikusan feldob egy dialógusablakot, amikor vége a játéknak (összegyűjtöttük-e az összes kosarat, vagy elkaptak a vadőrök).

Felhasználói esetek diagramja**Tervezés:**

- Programszerkezet:
- A programot háromrétegű architektúrában valósítjuk meg. A megjelenítés a *View*, a modell a *Model*, míg a perzisztencia a *Persistence* névtérben helyezkedik el.
- Perzisztencia:
- Az adatkezelés feladata a Maci Laci játéktáblával kapcsolatos információk tárolása.
- A *BearGameBoard* osztály egy érvényes Maci Laci pálya adatait tartalmazza. A tábla alapértelmezés szerint 15×15 -ös, 5 piknikkosárral és 3 őrrrel, de a pálya mérete a konstruktorban paraméterezhető. A tábla lehetőséget a pályaméret lekérdezésére és módosítására (*BoardSize*), valamint a pálya megadott indexű elemeinek lekérdezésére és módosítására (*this indexer*).

Az alkalmazás csomagdiagramja



- Modell:
- A modell lényegi részét a *BearGameModel* osztály valósítja meg, amely a tábla tevékenységeit, valamint a játék egyéb paramétereit, például az időt (*_gameTime*) és az összegyűjtött piknikkosarakat (*_basketCount*) szabályozza. A típus lehetőséget ad új játék kezdésére (*NewGame*), valamint Maci Laci mozgatására (*MoveBear*). Az örök az idő telésével mozognak (*AdvanceTime*, *MoveGuards*). Új játék előtt kiválaszthatjuk a pálya méretét, egyébként a jelenlegi pályán kezdődik új játék.
- A játékállapot változásáról a *GameAdvanced* esemény, míg a játék végéről a *GameOver* esemény tájékoztat. Az események argumentuma (*BearGameEventArgs*) tárolja a győzelem állapotát, a kosarak számát, valamint a játékidőt.
- A modell a *LoadFields* metódus segítségével olvassa be a pályákat. A pályák adatai szöveges fájlokban vannak eltárolva. Ezeket az adatokat a programban bármikor be lehet tölteni. A fájl első sora megadja a tábla méretét (*n*), valamint a piknikkosarak számát. A fájl többi része izomorf leképezése a pályának, azaz összesen *n* sor következik, és minden sor *n* számot tartalmaz szóközzel választva. A számok 0-4 közöttiek lehetnek, ahol 0: fű, 1: fa, 2: ör, 3: piknikkosár, 4: Maci Laci.
- A pálya méretét a *GameSize* felsorolási típuson át kezeljük.

Nézet:

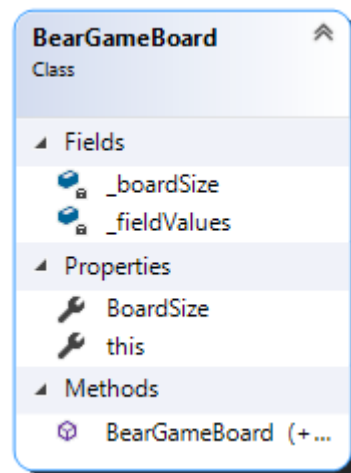
- A nézet csak egy képernyőt tartalmaz, a *MainWindow* osztályt. A nézet egy rácsban tárolja a játékmezőt, a menüt és a státuszsort. A játékmező egy *ItemsControl* vezérlő, ahol

dinamikusan felépítünk egy rácsot, amely gombokból áll. Minden adatot adatkötéssel kapcsolunk a felülethez, továbbá azon keresztül szabályozzuk a gombok színét is.

Környezet:

- Az App osztály feladata az egyes rétegek példányosítása (App_Startup), összekötése, a nézetmodell, valamint a modell eseményeinek lekezelése, és ezáltal a játék, az adatkezelés, valamint a nézetek szabályozása.
- A játék léptetéséhez tárol egy időzítőt is (_timer), amelynek állítását is szabályozza.

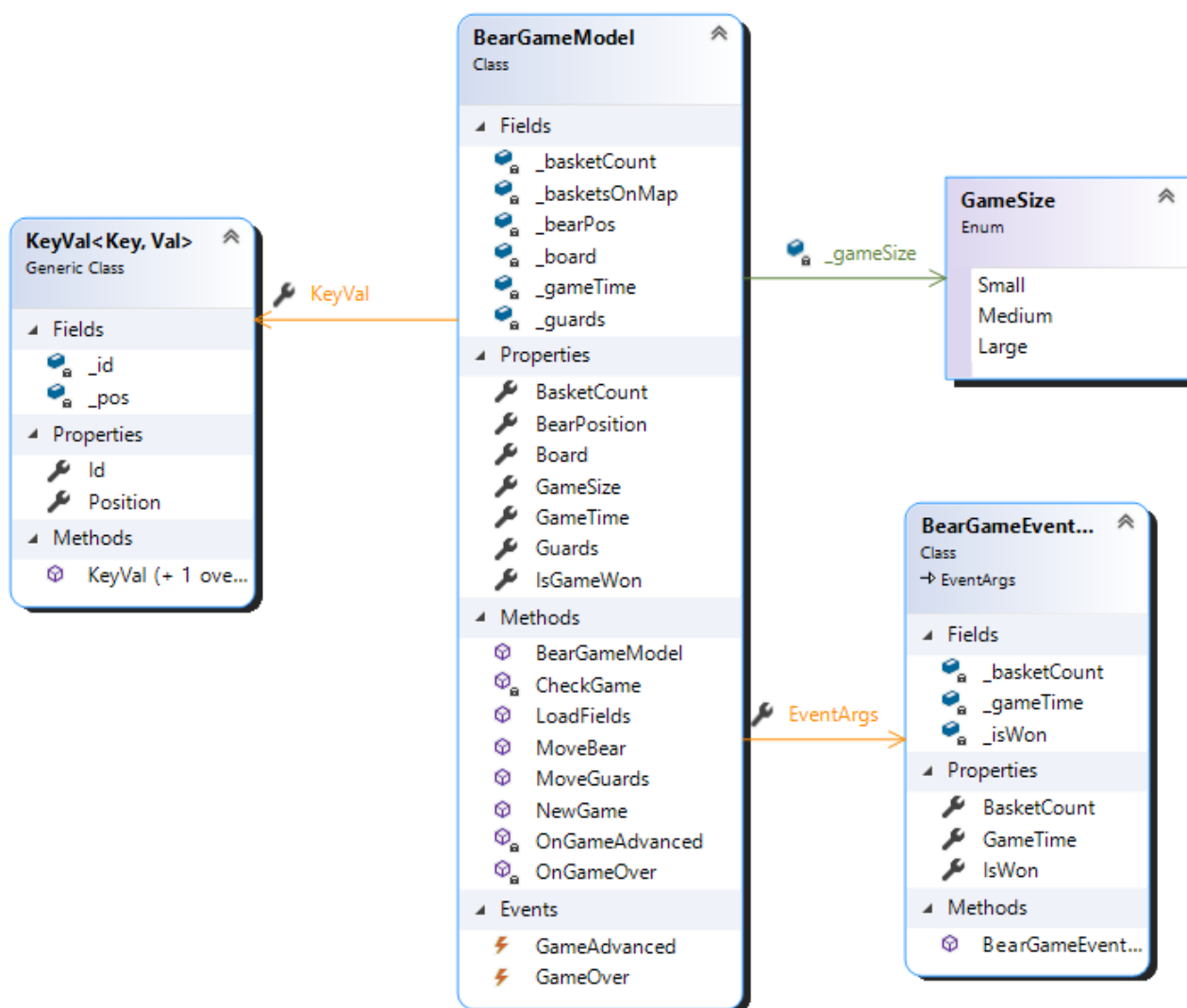
Az alkalmazás osztálydiagramja



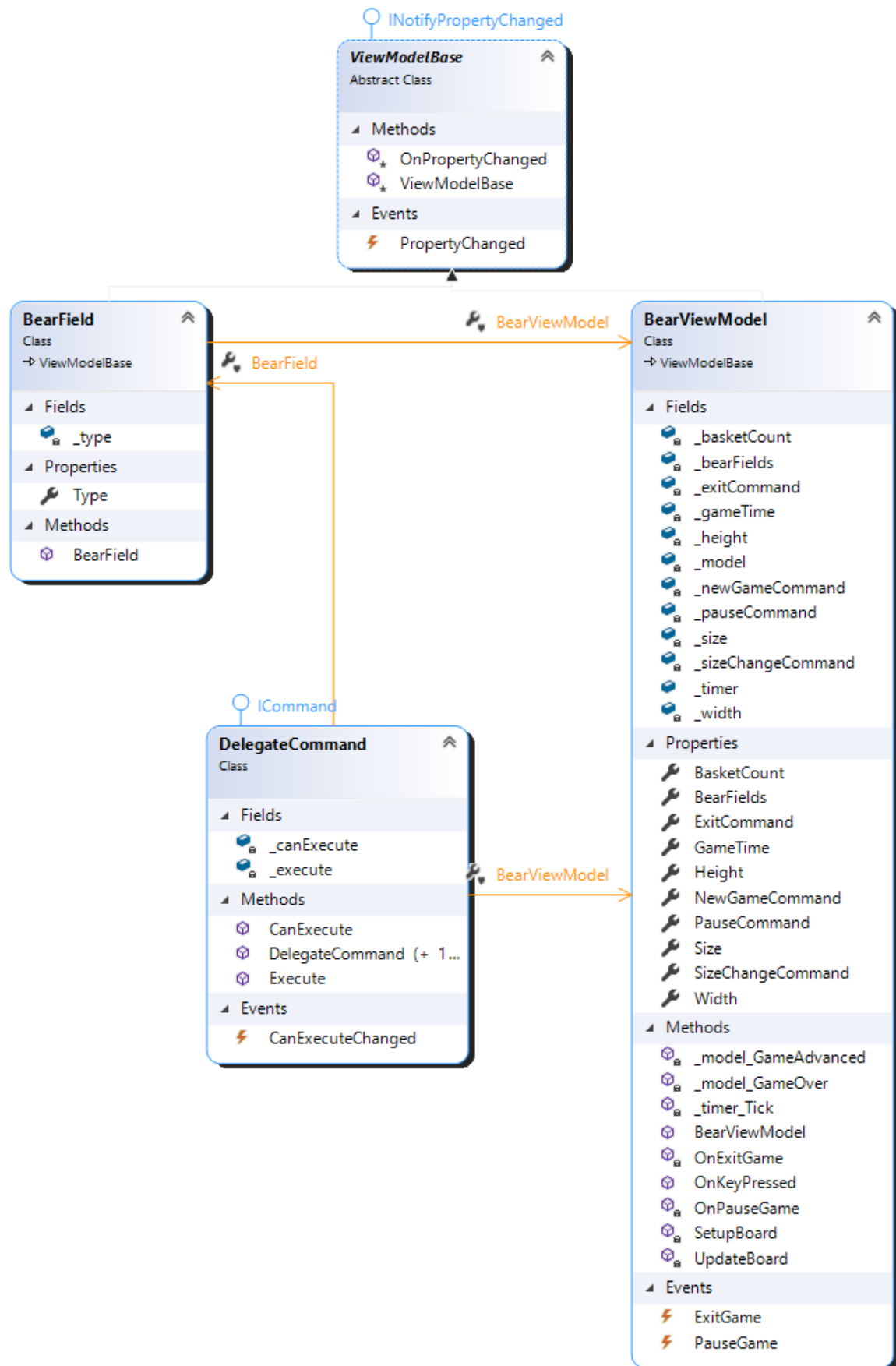
A Persistence csomag osztálydiagramja

Nézetmodell (5. ábra):

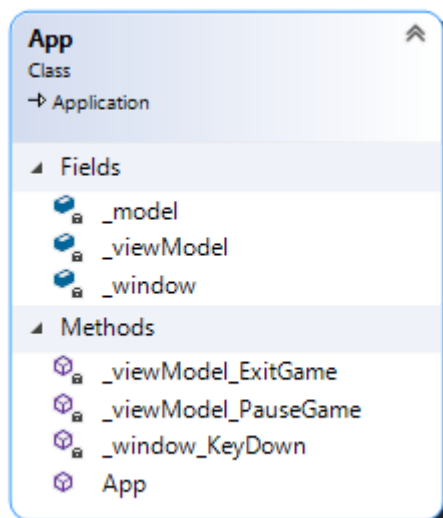
- A nézetmodell megvalósításához felhasználunk egy általános utasítás (DelegateCommand), valamint egy ős változásjelző (ViewModelBase) osztályt.
- A nézetmodell feladatait a BearViewModel osztály látja el, amely parancsokat biztosít az új játék kezdéséhez, valamint a kilépéshez. A parancsokhoz eseményeket kötünk, amelyek a parancs lefutását jelzik a vezérlőnek. A nézetmodell tárolja a modell egy hivatkozását (_model), de csupán információkat kér le tőle, illetve a játéknehézséget szabályozza. Direkt nem avatkozik a játék futtatásába.
- A játékmező számára egy külön mezőt biztosítunk (BearField), amely eltárolja a mező típusát. A mezőket egy felügyelt gyűjteménybe helyezzük a nézetmodellbe (BearFields).



A Model csomag osztálydiagramja



A nézetmodell osztálydiagramja



A vezérlés osztálydiagramja

Tesztelés:

- A modell funkcionalitása egységtesztek segítségével lett ellenőrizve a *ModelTest* osztályban. • Az alábbi tesztesetek kerültek megvalósításra:
- *NewGameSmallTest*, *NewGameMediumTest*, *NewGameLargeTest*: Új játék indítása, a mezők kitöltése, valamint a lépésszám ellenőrzése a méret függvényében.
- *MoveTest*: Játékbeli lépés hatásainak ellenőrzése.
- *MoveGuardsTest*: Az örök léptetésének ellenőrzése.