

Beadandó feladat dokumentáció

Készítette:

Juhász Zsombor

QGZPQA

Email: juhaszjuhi03@gmail.com

Feladat:

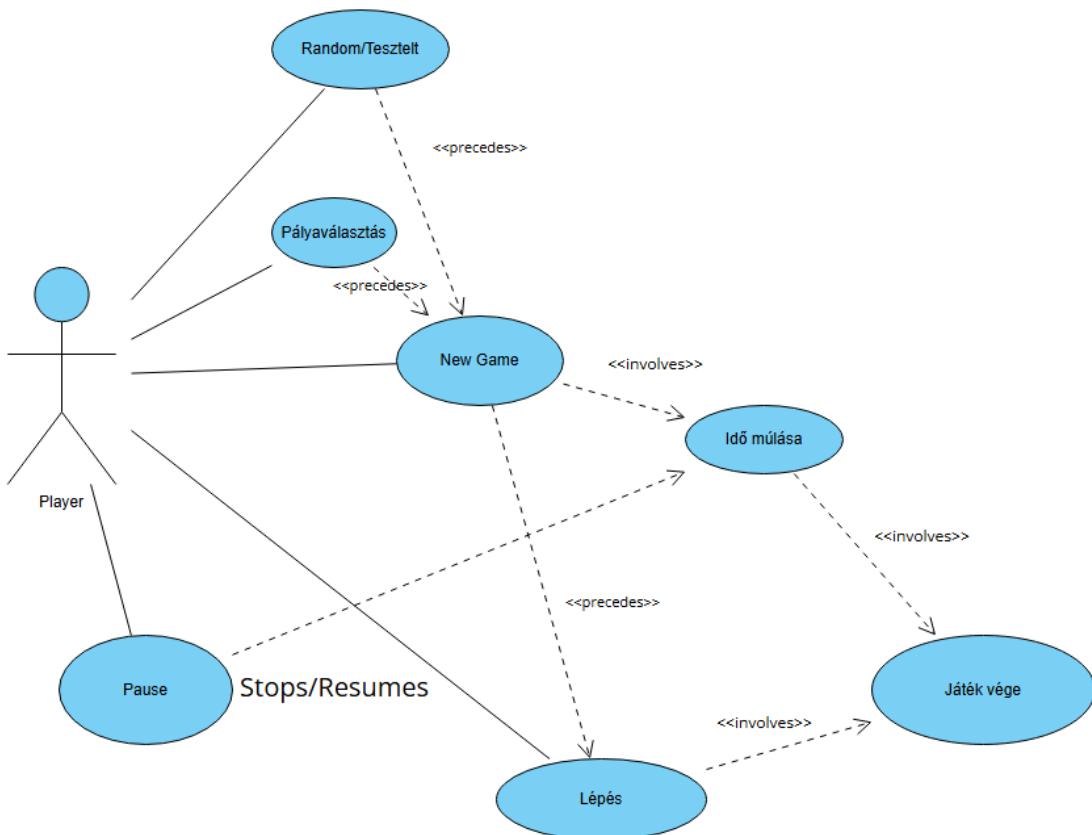
Készítsünk programot, amellyel a következő játékot játszhatjuk. Adott egy $n \times n$ elemből álló játékpálya, amelyben Maci Lacival kell piknikkosarakra vadásznunk. A játékpályán az egyszerű mezők mellett elhelyezkednek akadályok (pl. fa), valamint piknikkosarak. A játék célja, hogy a piknikkosarakat minél gyorsabban begyűjtsük. Az erdőben vadőrök is járóöznek, akik adott időközönként lépnek egy mezőt (vízszintesen, vagy függőlegesen). A járóözés során egy megadott irányba haladnak egészen addig, amíg akadályba (vagy az erdő szélébe) nem ütköznek, ekkor megfordulnak, és visszafelé haladnak (tehát folyamatosan egy vonalban járóöznek). A vadőr járóözés 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, vadőrok 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.

Elemzés:

- A feladatot egyablakos asztali alkalmazásként Windows Forms grafikus felülettel valósítjuk meg.
- Az ablakban elhelyeztem egy „New Game” gombot amivel a játékot megkezdjük.
- A játék megkezdése után megjelenik a „Pause” opció, amivel a játékot megállíthatjuk. Ilyenkor az idő nem telik és a játékos nem tud mozogni.
- A játéktábla méretét 3 opciónál választhatjuk ki egy combobox segítségével. Amennyiben úgy próbáljuk elindítani a játékot, hogy nem választottunk, egy felugró ablakba futunk.
- Van egy checkbox amivel kiválaszthatjuk, hogy a tesztelt, garantáltan végigjátszható pályákat akarjuk játszani, vagy a random generált, olykor lehetetlen nehézségű pályát.

- A játék elvesztése, vagy megnyerése esetén egy ablak ugrik fel.
- A játék során folyamatosan ki van írva a begyűjtött kosarak száma, és az eltelt idő a játék kezdete óta.

Felhasználói esetek:



Tervezés:

Programszerkezet:

- A programot háromrétegű architektúrában valósíjuk meg. A megjelenítés a View, a modell a Model, míg a perzisztencia a Persistence névtérben helyezkedik el.
- A program szerkezetét két projektre osztjuk implementációs megfontolásból: a Persistence és Model csomagok a program felületfüggetlen projektjében, míg a View csomag a Windows Formstól függő projektjében kap helyet

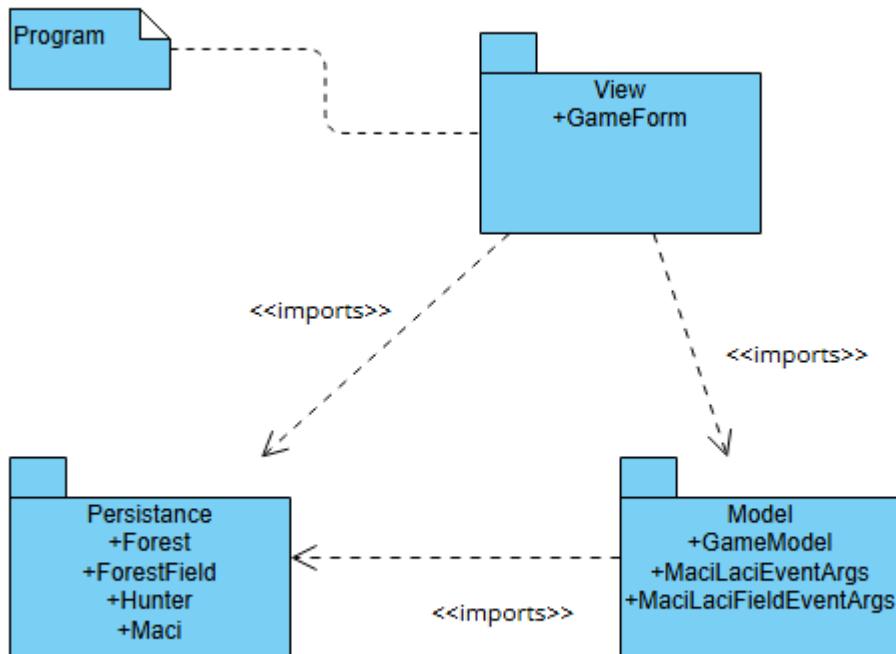
Perzisztencia:

- A Forest tábla lesz a játéktábla, ami generál egy konstruktörben megadott méretű pályát.
- A Maci lesz a mi Player által irányított karakterünk.
- A Hunters osztályból fognak majd a vadőrök példányosulni.
- A ForestField enumerátort arra használjuk, hogy a táblában megnevezzük a mezőket, ezzel megkönyebbítve a megjelenítést.

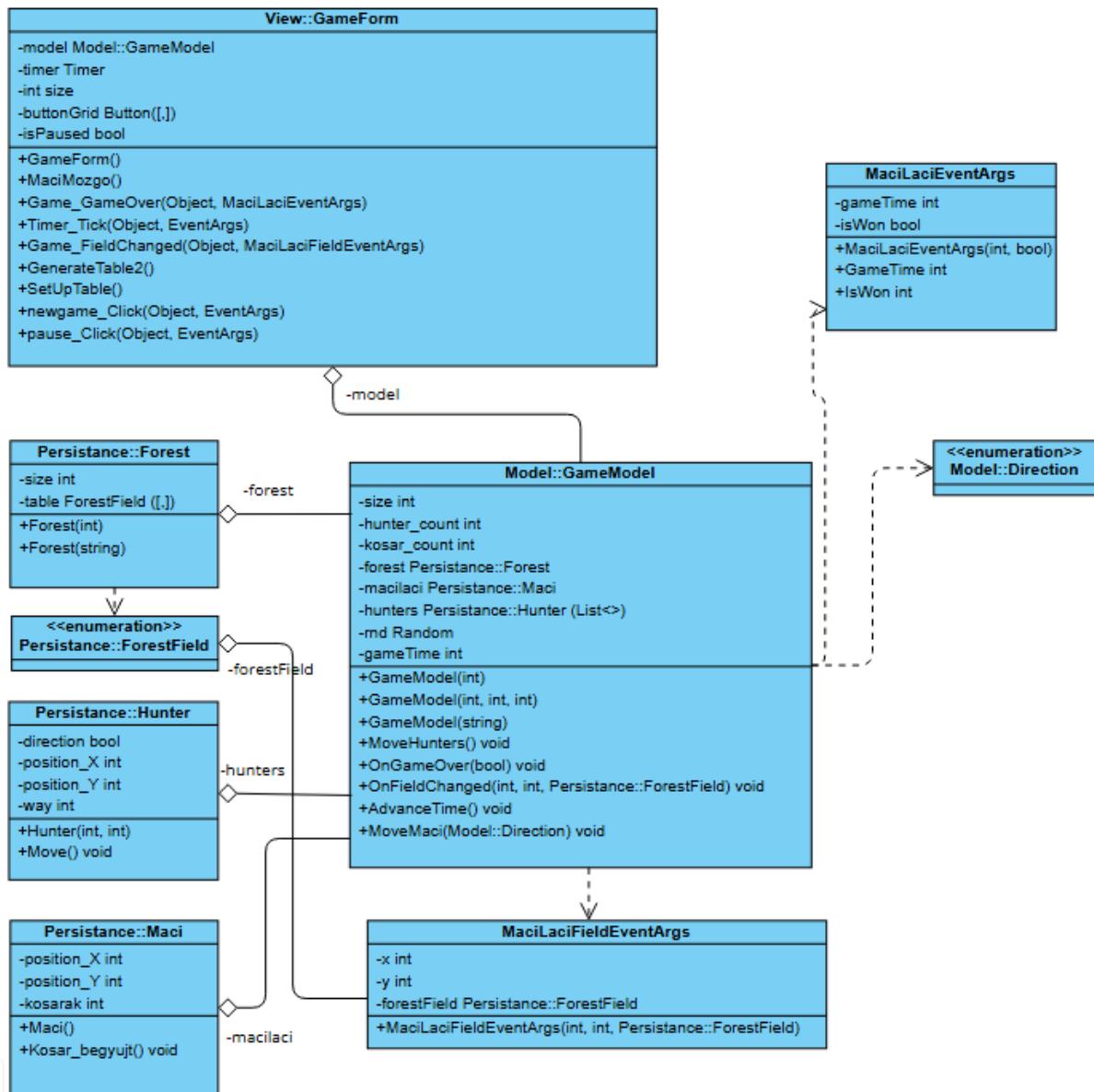
Model:

- A GameModel a lelke az egész projektnek, ebben történik maga a játék majd, melyet a View fog megjeleníteni.
- A játék folyamán Eventek váltódhatnak ki, például a játék megnyerését vagy elvesztését követően.
- Saját eventek a MaciLaciEventArgs, amit akkor használok, ha vége a játéknak, és a MaciLaciFieldEventArgs, ami a mezők változását jelzi a Viewnak, hogy frissítse a kijelzőt.

Az alkalmazás csomagdiagramja:



Az alkalmazás osztálydiagramja:



Tesztelés:

- A modell funkcionálisitása egységtesztek segítségével lett ellenőrizve a UnitTest1 osztályban.
- A következőket tesztem:
 - Inicializáció hibamentessége
 - Mozgástezt, tud-e a játékos mozogni, megfelelően történik-e a mozgás
 - Objektumokba ütközés tesztelése
 - Vadászok mozgása
 - Kosarak felvétele, megkapja-e a pontot a játékos