Eva Dukumentáció

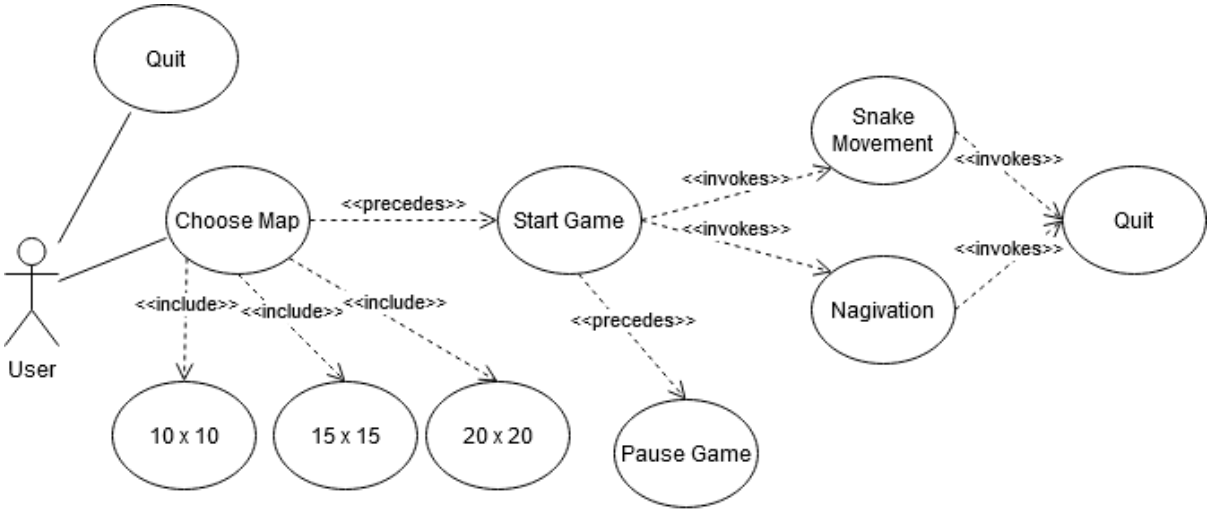
Assignment 04 Snake

Feladat:

Készítsük programot, amellyela klasszikus kígyó játékot játszhatjuk.Adott egy $n \times n$elemből álló játékpálya, amelyben akadályok (falak) találhatóak. A játékos egy kezdetben 5 hosszú kígyóval indul a képernyő közepén, amely vízszintesen, illetve függőlegesen halad rögzített időközönként a legutoljára beállított irányba. A kígyóval elfordulhatunk balra, illetve jobbra. A pályán véletlenszerű pozícióban mindig megjelenik egy tojás, amelyet a kígyóval meg kell etetni. Minden etetéssel eggyel nagyobb 1ess a kígyó. A játék célja, hogy a kígyó minél tovább elkerülje az ütközést az akadályokkal, a pálya szélével, illetve saját magával.A pályák méretét, illetve felépítését (falak helyzete) 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 mozog a kígyó). Továbbá ismerje fel, ha vége a játéknak. Ekkor jelenítse meg, hány tojást sikerült elfogyasztania a játékosnak.

Elemzés

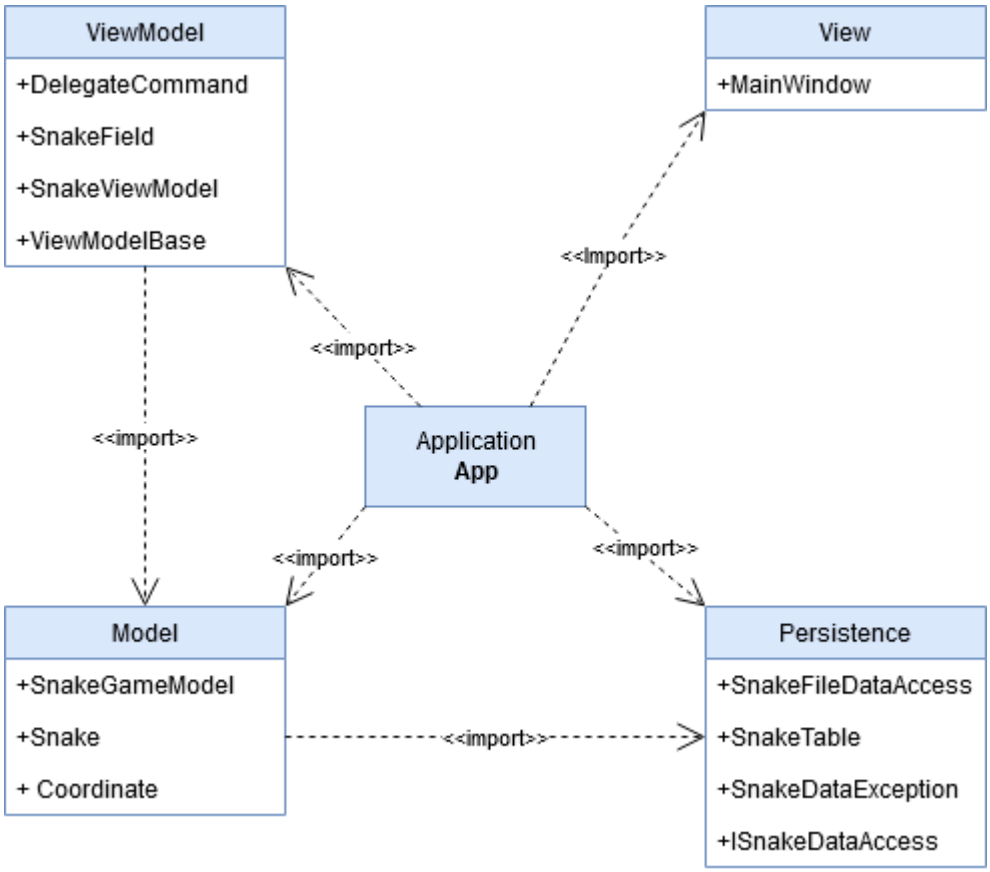
- A feladatot ablakos asztali alkalmazásként valósítjuk meg WPF felhasználásával.
- A játékban lehetőség van 3 különböző "$n \times n$" méretű ahol "n" lehet *11*, *15*, *23* és különböző felépítésű pálya kiválasztására ahol mindegyik pályára igaz, hogy fileban tároljuk őket és a kiválasztásakor ezeket beolvassuk majd generáljuk.
- Az ablak felépítése, egy
    - **Menu** – Ahol a File leugró ablak és Start/Pause nyomógomb található, a File továbbnyílik a pályák választásához valamint egy kilépési opcióhoz.
    - **ItemsControl** – Ezen a területen jelenítjük meg a generált pályát.
    - **StatusBar** – Itt követjük nyomon a kért statisztikákat, tehát hogy mennyi tojást fogyasztott a játékos.
- A kígyó irányítása az "↓", "→", "↑" és "←" gombokkal történik.
- A játék elvesztésekor a játék az adott játékmenetből statisztikát jelez majd újraindul.



Tervezés

Programszerkezet:

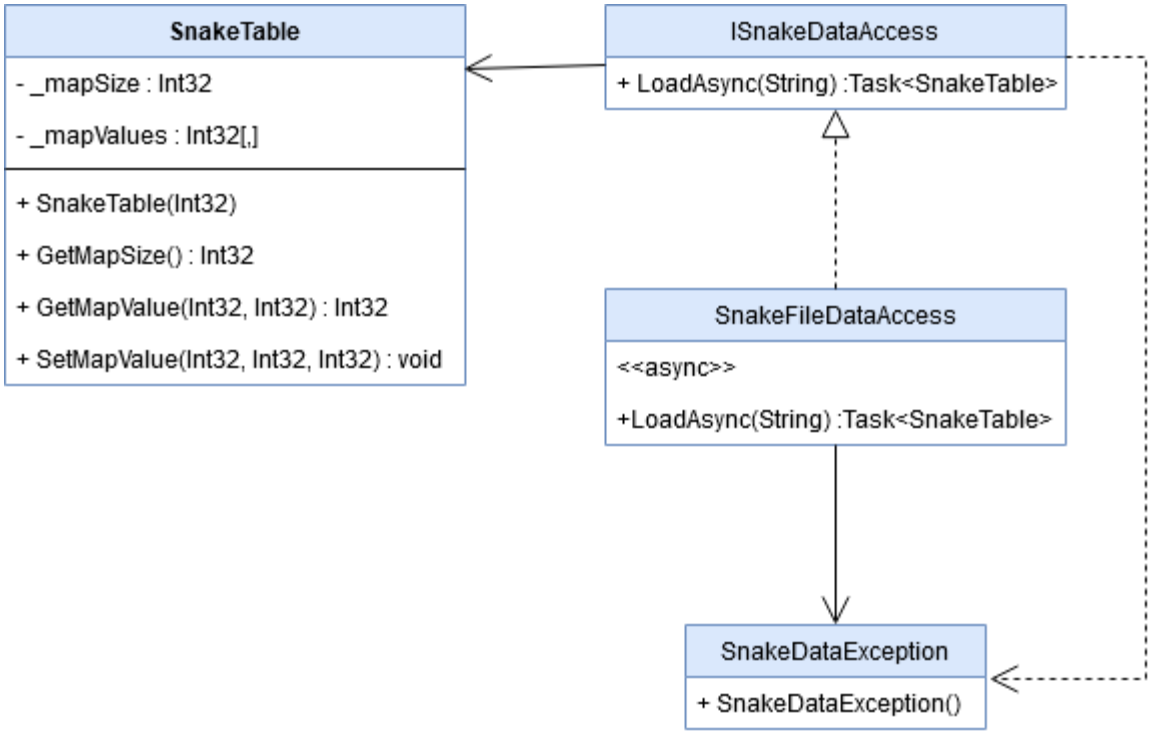
MVVM architektúrában valósítjuk meg. Az App példányosítja a modellt, nézetmodellt, nézetet és kezeli az ezek közötti kommunikációt.



Perszisztencia:

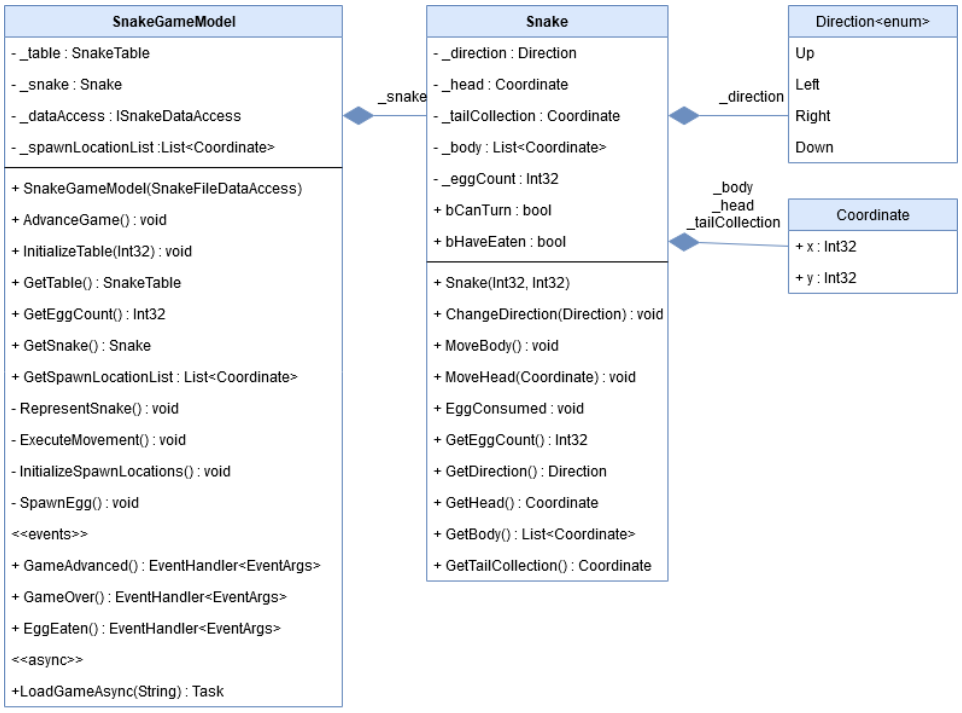
A perszisztencia törődik az program adatkezelésével, valamint lebonyolítja a különböző pályák betöltését.

- A SnakeTable egy Int-eket tartalmazó helyes megfeleltetése a vizuális táblának aminek celláinak értékei alapján fut a játék.
  3 különböző méretű pálya áll rendelkezésre mindegyik NxN mérteű N = {11, 15 ,23}.
  Ennek megadása a construktorban paraméterként történik.
  Innentől a mezők 5 értéket vehetnek fel
  0- üres, 1- a Kígyó feje, 2- a kígyó teste, 3- étel, 4- fal.
- Hosszú távú adatbetöltést a SnakeFileDataAccess valósítja meg, hibákat a SnakeDataException jelzi.
- A pályák txt kiterjesztésű fileban vannak elmentve ahol az első sor a pálya mérete a többi pedig egy soronkénti megfeleltetése a pályának



Modell:

- A modell réteg kezeli a játék logikáját
- A Snake osztály valósítja meg a kígyó mozgását, növekedését és tojások evését
- A Modell osztály pedig a tojások lerakását, pályák betöltését, és játékbeli események kiváltását kezeli.

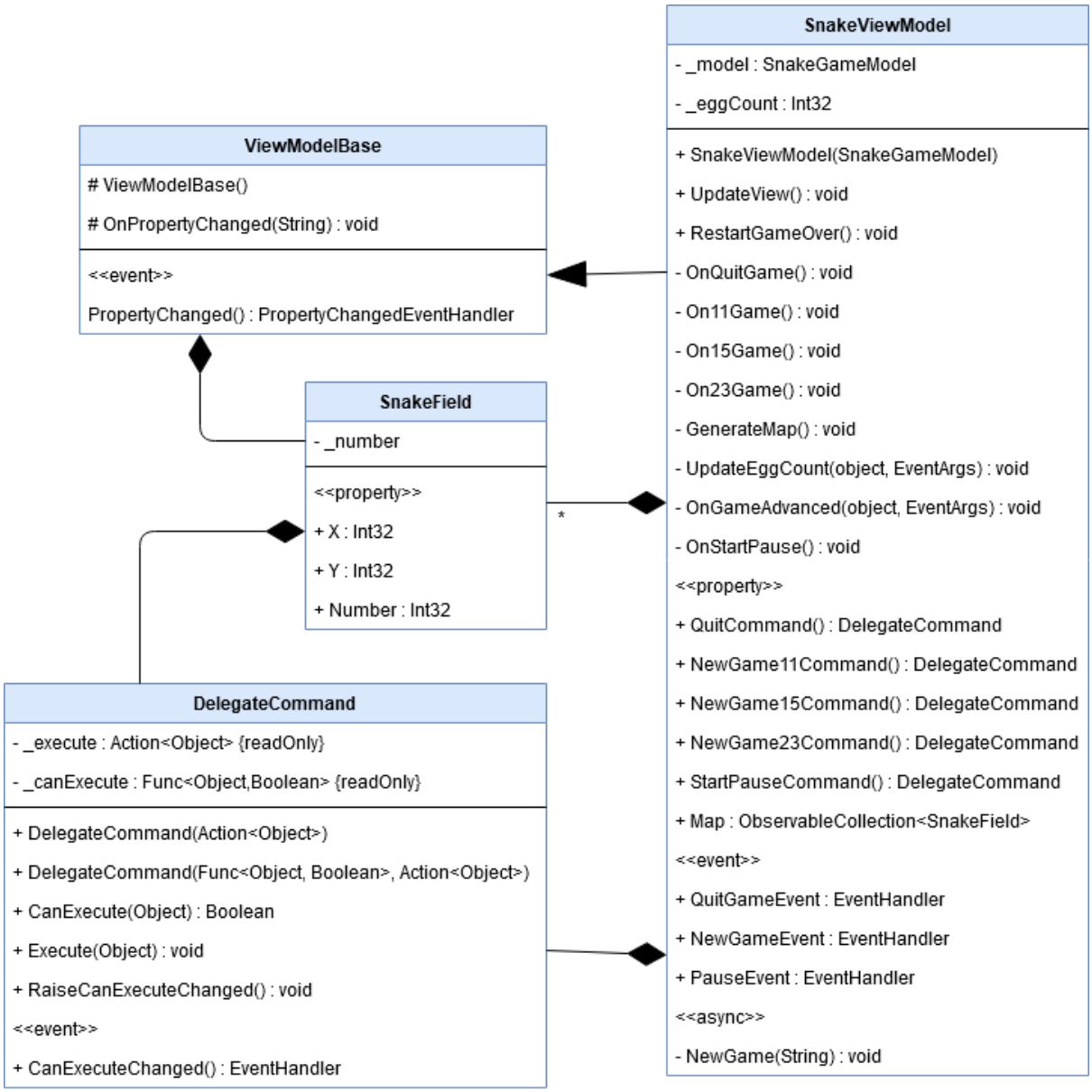


NézetModell:

A nézetmodell megvalósításához használunk egy általános utasítás osztályt (DelegateCommand) valamint egy ősosztályt a kinézeti elemek tulajdonságának megváltoztatásának felismerésére(ViewModelBase).

Ez az osztály kezeli a nézeti felület által kiváltott eseményeket, reprezentáláshoz hozzáfér a modellhez de játéklogikát nem bonyolít le.

A SnakeField osztály a vizuális megjelenítés elemeknek megfeleltetése.



Nézet:

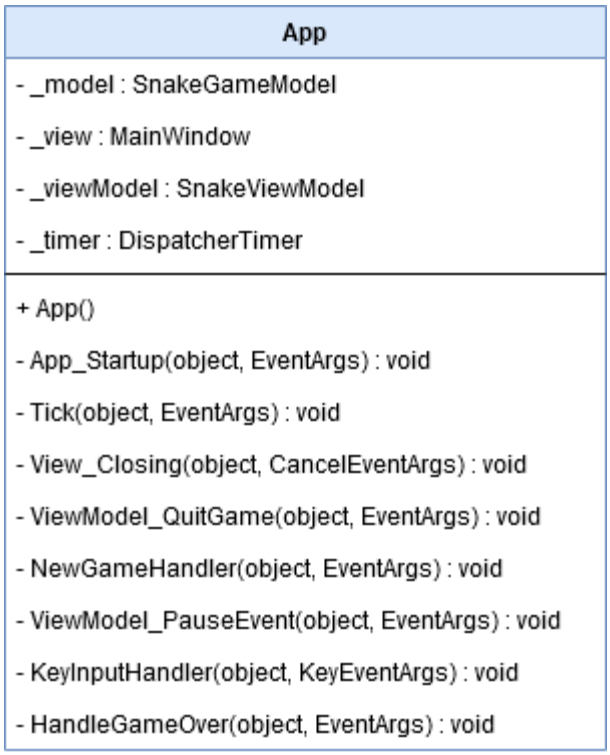
A nézet MainWindow egy rácsban tárolja a pályát, valamint létre van hozva egy menu, és egy statusbar a tojásszámok megjelenítésére.

A rács elemei “Border” amelyeknek a hátterét kötöttük a Nézetmodell SnakeField-jéhez.

Környezet:

A környezet példányosítja és köti össze a többi réteget, továbbá eseményeket kezel.

Itt található a _timer is ami ütemezi a játék menetét.



Tesztelés

A modell működését a ***SnakeTests*** osztállyal teszteljük.

Tesztesetek:

TestInitial – A kígyó kezdőértékeit teszteli

TestMovementOnAdvanceGame – A kígyó mozgását és annak következményeit teszteli

TestEating – Egy tojás elfogyasztását teszteli

TestWallHit –Egy falnak ütközést tesztel

Készítette: Márton Zoltán _ B44T65