

Eva Dokumentáció

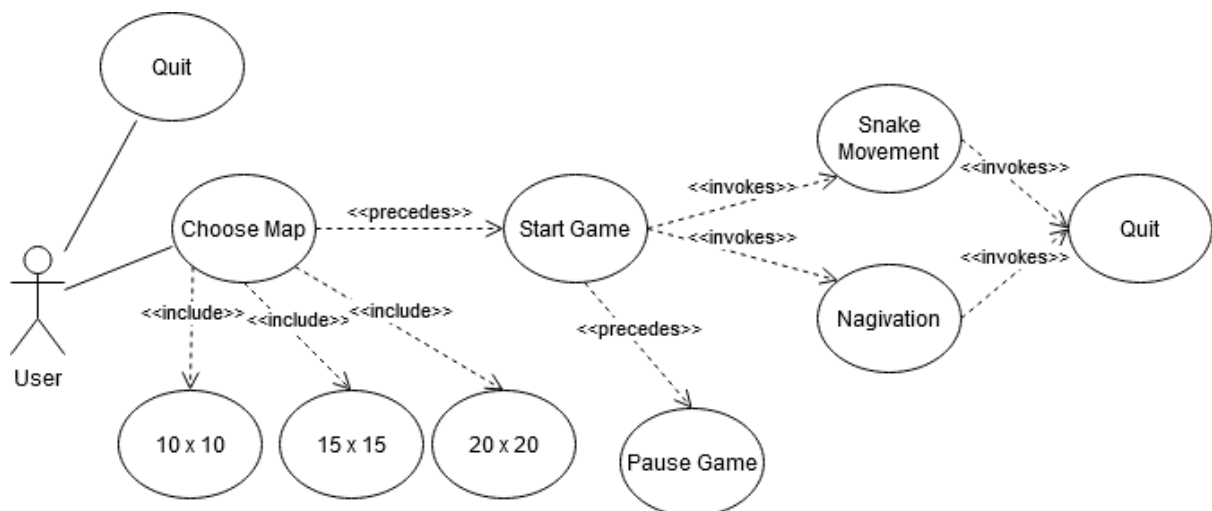
Assignment 03 Snake

Feladat:

Készítsük programot, amellyel 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 lesz 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

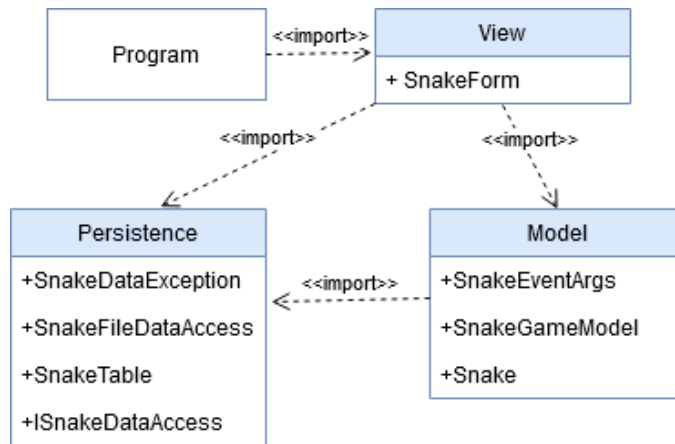
- A feladatot ablakos asztali alkalmazásként valósítjuk meg Windows Forms felhasználásával.
- A játékban lehetőség van 3 különböző " $n \times n$ " méretű ahol " n " lehet 10, 15, 20 és különböző felépítésű pálya kiválasztására ahol mindegyik pályára igaz, hogy fileban tároljuk őket és a kiválasztásakor ezeket beolvassuk majd generáljuk.
- Az ablak felépítése, egy
 - **menuStrip** – Ahol a File legró ablak és Start/Pause nyomógomb található, a File továbbnyílik a pályák választásához valamint egy kilépési opcióhoz.
 - **Panel** – Ezen a területen jelenítjük meg a generált pályát.
 - **StatusStrip** – Itt követjük nyomon a kért statisztikákat, tehát hogy mennyi tojást fogyasztott a játékos.
- A kígyó irányítása az "A" és "D" gombokkal történik amik a kígyó irányát változtatják a kígyó fejéhez relative balra illetve jobbra.
- A játék elvesztésekor a játék az adott játékmenetből statisztikát jelez majd kilép.



Tervezés

Programszerkezet:

Megvalósítása harm rétegben teljesül meg. A megjelenítésért a **View**, a játék logikájának és adatainak kezelését a **Model** bonyolítja le, míg a **Perszisztencia** a tábla adatait, illetve a betöltést biztosítja



Perszisztencia:

A SnakeTable osztály reprezentálja numerikusan a játéktáblát.

Minden mezőnek van egy numerikus értéke (**_mapValues**) ezek lehetnek 0-1-2-3-4 pályától, véletlenszerű tojáslerakástól és kígyópozíciótól függően.

A pályák tárolását az ISnakeDataAccess-el érjük el amivel lehetőségünk van a pályák betöltésére (**LoadAsync**) ez aszinkron módon történik.

A filekezelés alatt előforduló hibákat a SnakeDataException fogja jelezni.

A fileok kiterjesztése txt, mivel csak n^2+1 számot tárolunk.

A file legelső sora a betöltendő pálya mérete, a többi n sor pedig a pályán elhelyezkedő falak helyzetét adja meg.

Modell:

A modell logikai részét a **SnakeGameModel** osztály valósítja meg, míg a kígyó koordinátáit, testi mozgását, evés és ezáltal növekedés kezelését a Snake osztály kezeli.

Példányosításkor megkapja paraméterként az adatkezelési felületet így elérhető lesz a betöltés

A játék kezdését a ,mivel fileból töltjük a pályákat, a **LoadGameAsync** végzi, a léptetést az AdvanceGame függvény végzi ami elvégzi a kígyó mozgását és kezeli a mozgás következményeit.

Állapotváltozások esetén az **GameAdvanced** és a **GameOver** esemény hívódik.

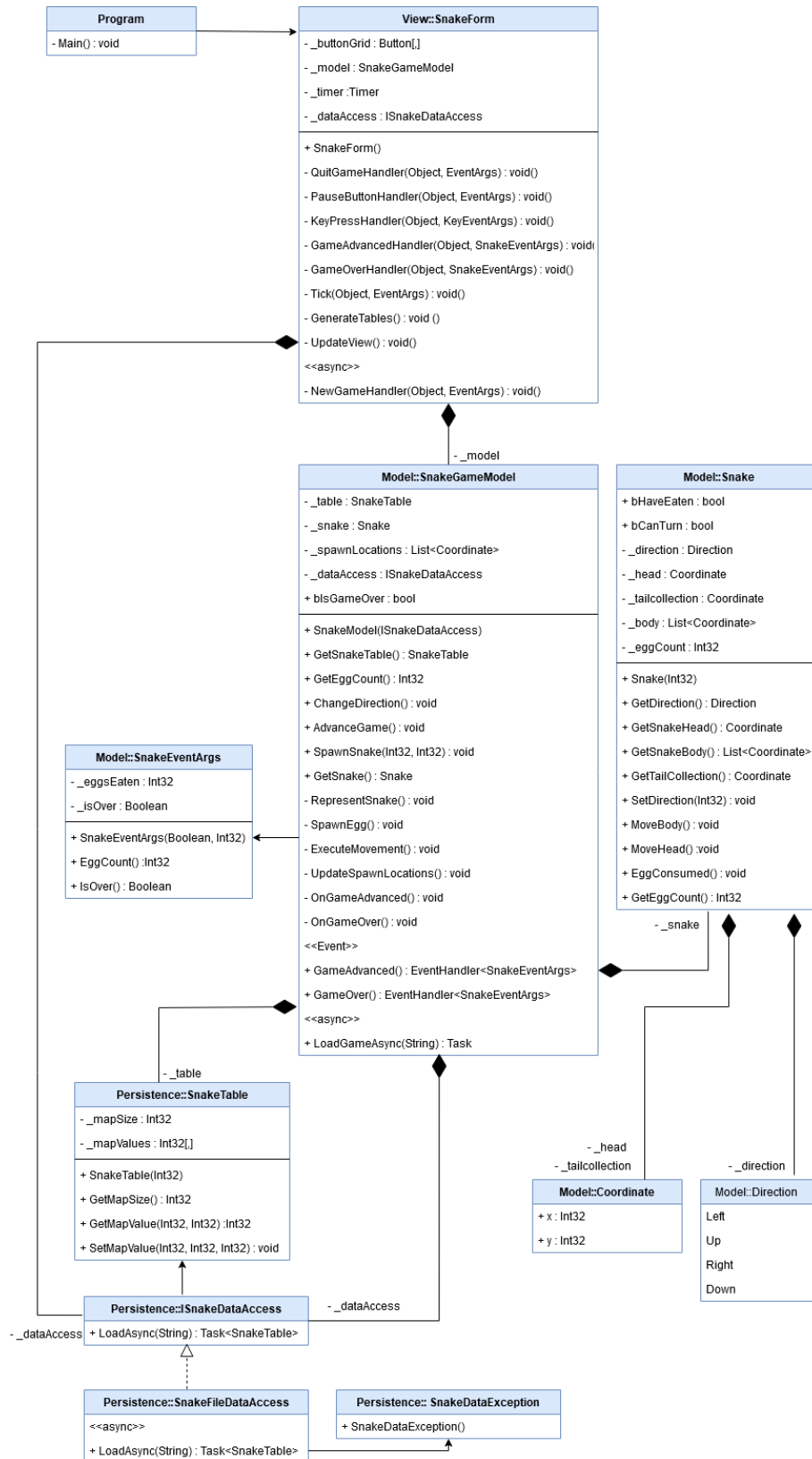
Nézet:

A SnakeForm osztály végzi a vizuális reprezentációját a játéknak.

Ez tartalmazza a _modellt és az adatelérés példányát is, valamint egy **Timer** típust (**_timer**) ami ütemezi a játék sebességét.

Egy kikapcsolt gombmező reprezentálja a játékpályát ami reagál a modell **SnakeTabl** numerikus táblájának értékeire.

Az **“A”** és **“D”** gombokkal tud irányítani a játékos, ezt is a nézet kezeli.



Tesztelés

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

Tesztesetek:

TestSnakeInitial – A kígyó kezdőértékeit teszteli

TestSnakeMovement – A kígyó mozgását és annak következményeit teszteli

TestEggConsumption – Egy tojás elfogyasztását teszteli

TestWallHit – Egy falnak ütközést tesztel

Készítette: Márton Zoltán _ B44T65