

3. Beadandó feladat dokumentáció

Prucs Ákos András

Neptun kód: ENPCTE

E-mail: enpcte@inf.elte.hu

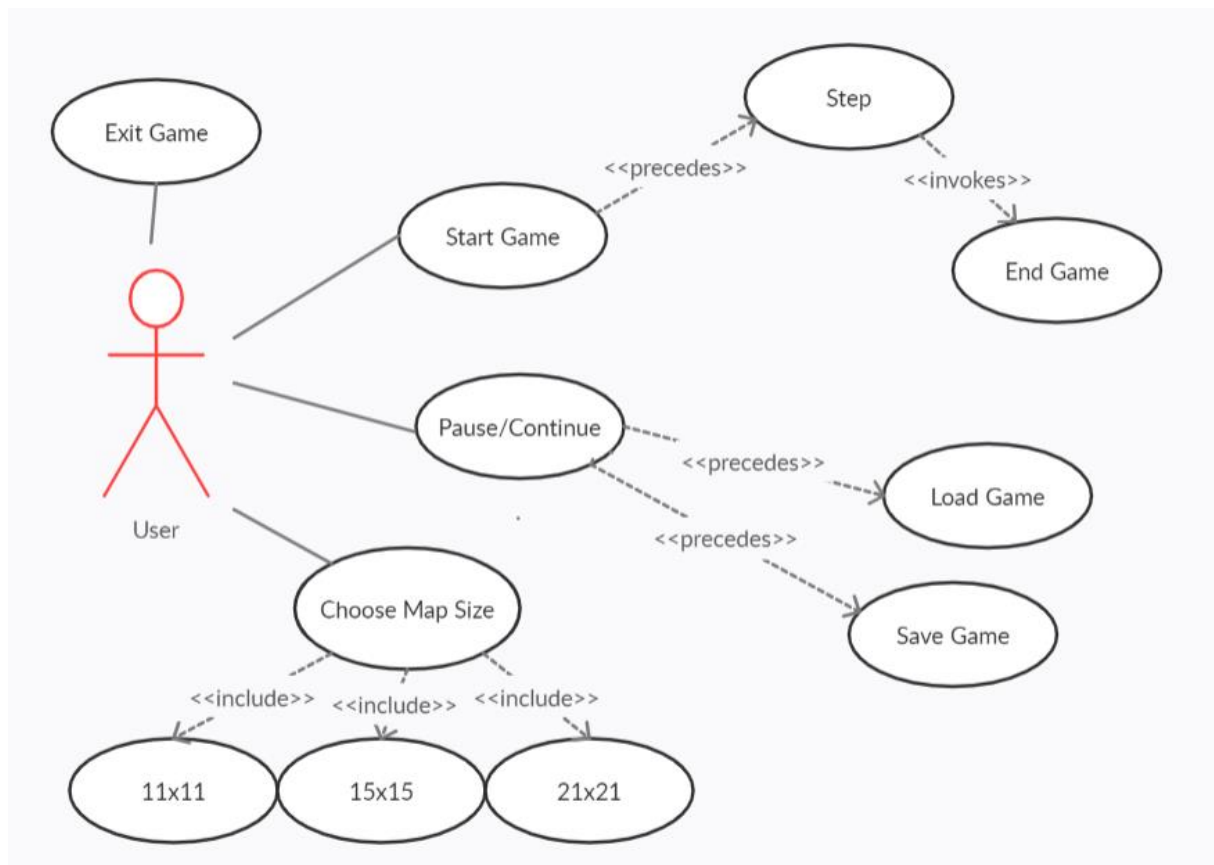
Feladat:

Menekülj

Készítsünk programot, amellyel a következő játékot játszhatjuk. Adott egy $n \times n$ elemből álló játékpálya, ahol a játékos két üldöző elől próbál menekülni, illetve próbálja őket aknára csalni. Kezdetben a játékos játékpálya felső sorának közepén helyezkedik el, a két üldöző pedig az alsó két sarokban. Az ellenfelek adott időközönként lépnek egy mezőt a játékos felé haladva úgy, hogy ha a függőleges távolság a nagyobb, akkor függőlegesen, ellenkező esetben vízszintesen mozognak a játékos felé. A pályán véletlenszerű pozíciókban aknák is elhelyezkednek, amelyekbe az ellenfelek könnyen beleléphetnek, ekkor eltűnnek (az akna megmarad). A játékos vízszintesen, illetve függőlegesen mozoghat (egyesével) a pályán, és célja, hogy az ellenfeleket aknára csalja, miközben ő nem lép aknára. Ha sikerül minden üldözőt aknára csalnia, akkor győzött, ha valamely ellenfél elkapja (egy pozíciót foglal el vele), vagy aknára lép, akkor veszített. A program biztosítson lehetőséget új játék kezdésére a pályaméret megadásával (11×11 , 15×15 , 21×21), valamint játék szüneteltetésére (ekkor nem telik az idő, és nem léphet senki). Ismerje fel, ha vége a játéknak, és jelenítse meg, hogy győzött, vagy veszített-e a játékos. Ezen felül szüneteltetés alatt legyen lehetőség a játék elmentésére, valamint betöltésére. A program játék közben folyamatosan jelezze ki a játékidőt

Elemzés:

A feladat egy asztali alkalmazásként Windows Forms grafikus felülettel valósítom meg. A program indításakor a felhasználó beállíthatja a legenerálandó pálya méretét, majd a START gombra kattintva elkészül a játéktábla. Ezt követően a billentyűzet nyilaival léphet a táblán, amíg a játék a végéhez nem ér. Továbbá, a felhasználónak lehetősége van szüneteltetni a játékot, ekkor a táblán lévő egységek automatikus léptetése leáll. Szünet állapotban két új opció nyílik meg, a játék elmentése és egy mentés betöltése. Mentés esetén a kiválasztott helyre mentődik egy .txt kiterjesztésű állomány, benne a játékállás adataival. Egy játék betöltésekor egy kiválasztott mentés betöltődik, és a CONTINUE gombra kattintva folytatódik a játék.



Felhasználói esetek diagramja

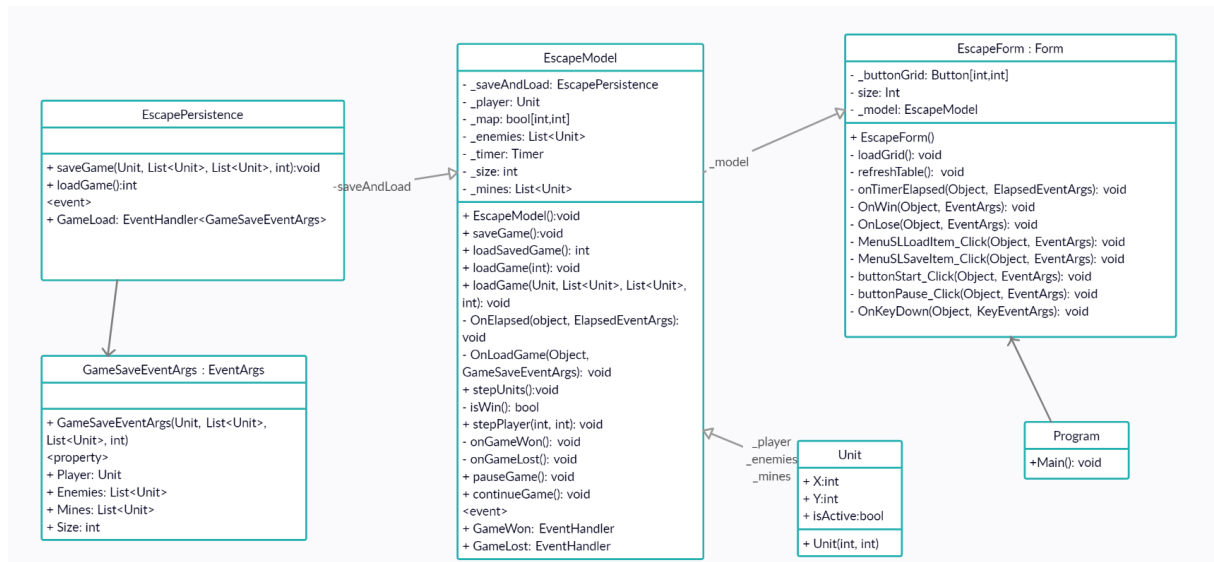
Tervezés:

A program három rétegű architektúrában van megvalósítva. A három rétege a megjelenésért felelős View, a logikáért felelős Model és a játék mentés és betöltéséért felelős Persistene.

A Model a loadGame() utasítással hozza létre a játék elemeit. A táblát egy boolean tömbben tárolja, a táblán szereplő egységeket pedig egy Unit (tartalmazza a pozíciót és az aktívsgot) típussal tárolja. Időközönként a Model lépteti az egységeket a stepUnits() utasítással, továbbá a játékos léptetésének utasítását a View-től kapja meg, ez a stepPlayer() utasítás.

A Persistence-ért felelős osztály mindössze a mentés és betöltésért felelős utasításokat tartalmazza. Mentéskor a Model-től megkapja a paramétereket, betöltéskor pedig egy származtatott EventArgs osztállyal továbbítja a paramétereket vissza a Model-be.

A Form-on (View) a START gomb minden egyes indításnál letörli az előző táblát, majd pedig egy újat generál. A győzelem és vereséget a Model eseménykezelőkön keresztül közvetíti a View felé.



Az alkalmazás osztálydiagramja