

# 1.Beadandó feladat dokumentáció

## Készítette:

Nagy Levente (uji470)

E-mail: [levente0517@gmail.com](mailto:levente0517@gmail.com)

## Feladat:

### 8. Aszteroidák

Készítsünk programot, amellyel az aszteroidák játékot játszhatjuk.

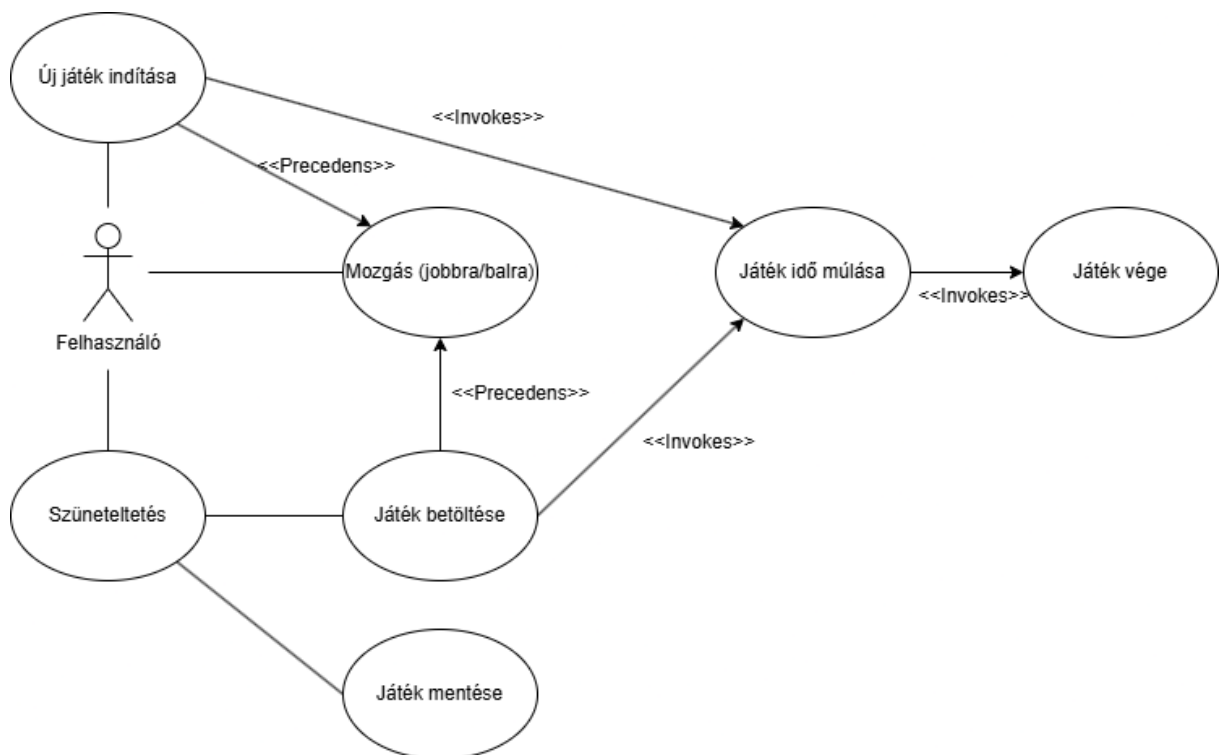
A feladatunk az, hogy egy űrhajó segítségével átnavigáljuk egy aszteroidamezőn. Az űrhajóval a képernyő alsó sorában tudunk balra, illetve jobbra navigálni. A képernyő felső sorában meghatározott időközönként véletlenszerű pozícióban jelennek meg az aszteroidák, amelyek folyamatosan közelednek állandó sebességgel a képernyő alja felé. Az idő múlásával egyre több aszteroida jelenik meg egyszerre, így idővel elkerülhetetlenné válik az ütközés. A játék célja az, hogy az űrhajó minél tovább elkerülje az ütközést.

A program biztosítson lehetőséget új játék kezdésére, valamint játék szüneteltetésére (ekkor nem telik az idő, és nem mozog semmi a játékban). Ismerje fel, ha vége a játéknak, és jelenítse meg, mennyi volt a játékidő. Ezen felül szüneteltetés alatt legyen lehetőség a játék elmentésére, valamint betöltésére.

## Elemzés:

- Indításkor → a játék elkezdődik, a pálya  $n \times m$ -es, így az  $m$ . sorban van az űrhajó, az aszteroidák pedig az 1. sor,  $n$ . random négyzetében jelennek meg. Az aszteroidák rögzített sebességgel esnek, azaz  $m := 1 + x$  -lesz a helyzetük ahol az  $x :=$  hány másodperce jelentek meg. Az űrhajó rögzített az  $m$ .-dik sorban és csak jobbra, balra tud mozogni, szóval azt változtatja az  $n$  oszlopban hol helyezkedik el. Amennyiben  $\text{űrhajó}(n,m) == \text{aszteroida}(n,m) \rightarrow \text{game over}$ .
- A játék idővel gyorsuljon fel → több aszteroida túlélő idővel arányosan
- Amikor az aszteroidák elhagyják a képernyőt, azaz túlsordulna a mátrixon, töröljük őket.
- A feladatot egyablakos asztali alkalmazásként Windows Forms grafikus felülettel valósítjuk meg.
- Az ablak tetejére elhelyezünk egy menüt a következő menüpontokkal:
  - File (Új játék kezdése, Játék szüneteltetése [ ez idő alatt mentés, betöltés ])
- Játék végét fel kell ismernünk és utána kiírni mennyi volt a túlélő idő.

- Felhasználói esetek:



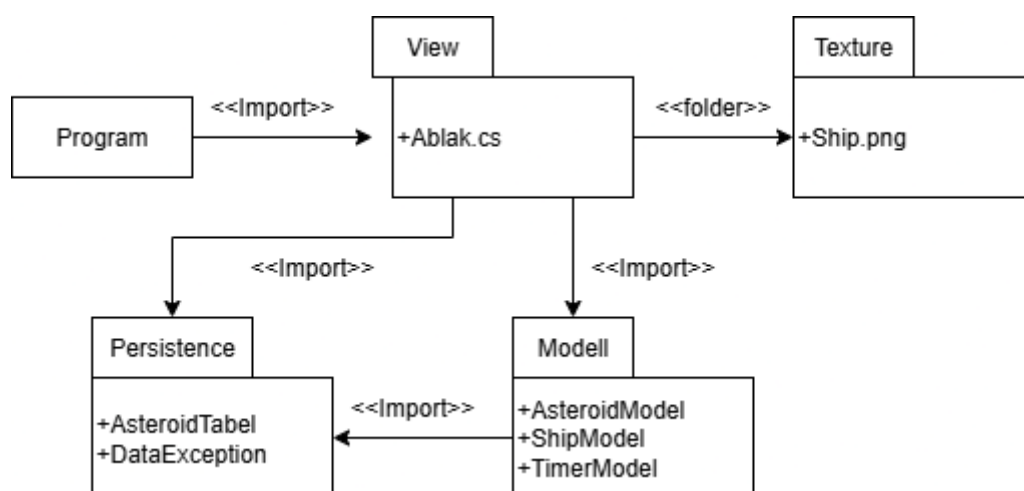
1.Ábra felhasználói esetek diagramja

ui.: A játék idő múlása azért eredményezi a Játék végét idővel, mivel a feladat azt állította egy bizonyos idő után elkerülhetetlen, hogy valaki veszítsen.

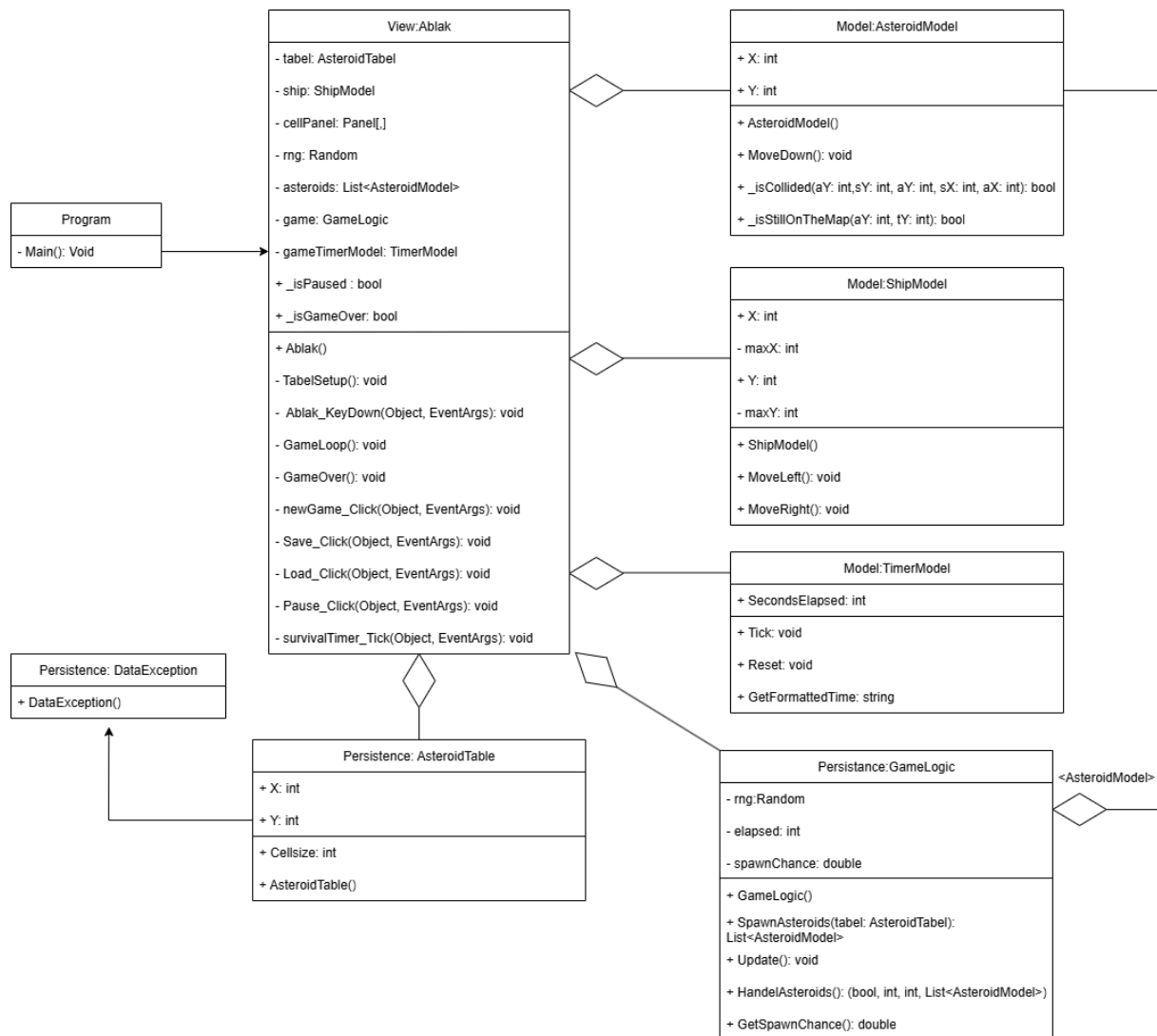
## Tervezés:

- **Programszerkezet:**
  - o A Program 3 rétegű architektúrában valósíjuk meg. A megjelenés a **View**, valamint ezen belül egy **Texture** mappa ahol a hajó textúráját tároljuk, a modellt a **Model**, és a perzisztencia a **Persistence** névtérben helyezkedik el. A program csomagszerkezete a 2. ábrán.
  - o 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:**
  - o Az adatkezelés feladata a Pályaával kapcsolatos információk tárolása
  - o Az AsteroidModel fogja ezt tárolni
  - o Amikor elmentünk egy játékot megakarjuk jegyeztetni az aszteroidák helyzetét (n,m), az űrhajó helyzetét (n,m) valamint az időt. Ezt egy .txt fájlba fogjuk eltárolni, amelybebből aztán be is fogjuk tölteni majd.

- A fájl első sora megadja az időt, az azt követő sor az űrhajó helyzetét, és az utána való sorok, az aszteroidák számát illetve azok helyzetét.
  - Az exceptionokat az **AsteroidException** osztály fogja kezelni.
  - A játék logikáját a **GameLogic** osztály fogja kezelni.
- **Modell:**
- A hajó adatait illetve mozgásait a **ShipModel**, az aszteroidák megjelenését és esését az **AsteroidModel** és az idő múlását, feljegyzését a **TimerModel** osztály fogja megvalósítani. Ezen kívül még ez fogja kezelni az idő múlását is.
- **Nézet:**
- Az idő ugyebár fontos lesz mivel ez határozza meg mennyi aszteroida fog jönni, mennyi lett a túlélő idő, ehhez lehet majd kötni az aszteroidák esését is. De ezt az időt meg is tudjuk állítani (**Pause** - al), amellyel megfagy a játék, az idő nem megy tovább → ergó, nem esnek az aszteroidák, illetve nem tudunk majd mozogni se.
  - A nézetben fogjuk tudni elmenteni (Save) illetve betölteni (Load) a játékot, amelyeket a felsősorban találunk az alkalmazásban.
  - Itt tudjuk triggerelni a (New Game) -et is amellyel teljesen újraindul a játék.



2.Ábra: Az alkalmazás csomagdiagramja



3. ábra: Az alkalmazás osztálydiagramja

## Tesztelés:

A tesztelés során végeztünk tesztek a **Model** csomagosztály minden osztályára, valamint a Perzisztencia GameLogic osztályára.

Elsőnek inicializáltuk a változókat, osztályokat, majd elvégeztük a tesztelést.

Megnéztük az aszteroidák mozgását, valamint hogy a művelet amellyel megnézzük, hogy összeütköztek e hajóval, valamint hogy ha kiérünk a mátrixból azt érzékelik e működnek-e.

A hajónál a jobbra, balra mozgást.

A Timer modellnél megnéztük, működik e a formálás valamint a léptetés, azaz a Tick.

A játék Logikájánál ellenőriztük a spawnChance kiszámításának működését valamint a, hogy a frissítés működik e.