**1. Beadandó feladat dokumentáció**

Készítette:

Komornik Márton

E-mail:kamrcikok@gmail.com

**Feladat: (19.)**

Készítsünk programot, amellyel a potyogós amőba játékot lehet játszani, vagyis az amőba azon változatát, ahol a jeleket felülről lefelé lehet beejteni a játékmezőre. A játékmező itt is 𝑛 × 𝑛 -es tábla, és ugyanúgy X, illetve O jeleket potyogtathatunk a mezőre. A játék akkor ér véget, ha betelik a tábla (döntetlen), vagy valamelyik játékos kirak 4 egymás melletti jelet (vízszintesen, vagy átlósan). A program minden lépésnél jelezze, hogy melyik játékos következik, és a tábla egy üres mezőjére kattintva helyezhessük el a megfelelő jelet. Természetesen csak a szabályos lépéseket engedje meg a program. A program biztosítson lehetőséget új játék kezdésére a táblaméret megadásával (10 × 10, 20 × 20, 30 × 30), játék szüneteltetésére, valamint játék mentésére és betöltésére. Ismerje fel, ha vége a játéknak, és jelenítse meg, melyik játékos győzött (a táblán jelölje meg a győztes 4 karaktert). A program folyamatosan jelezze külön-külön a két játékos gondolkodási idejét (azon idők összessége, ami az előző játékos lépésétől a saját lépéséig tart, ezt is mentsük el és töltsük be).

**Elemzés:**

* A játékot három táblamérettel játszhatjuk, 10x10, 20x20, és 30x30 méretű táblákon.  
  A program indításkor 10x10 táblát gyárt alapértelmezettként.
* A feladatot egyablakos asztali alkalmazásként Windows Forms grafikus felülettel valósítjuk meg.
* Az ablakban elhelyezünk egy menüt a következő menüpontokkal: File (Új játék, Játék betöltése, Játék mentése, Kilépés), Beállítások (10x10, 20x20,30x30). Az ablak alján megjelenítünk egy státuszsort, amely a játékosok gondolkodási idejét jelzi.
* A játéktáblát egy NxN-es nyomógombokból álló rács reprezentálja. A nyomógomb egérkattintás hatására a legalsó elérhető helyre a soron következő játékos jelét helyezi az adott oszlopban.
* A játék automatikusan feldob egy dialógusablakot, amikor vége a játéknak (döntetlen helyzet vagy egy játkos 4 jelet gyűjtött a 8 irány egyikében). Szintén dialógusablakokkal végezzük el a mentést, illetve betöltést, a fájlneveket a felhasználó adja meg.
* A felhasználói esetek az 1. ábrán láthatóak.

A képen szöveg, diagram, Betűtípus, képernyőkép látható

Automatikusan generált leírás

**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 csomagszerkezete a 2. ábrán látható.
  + 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**:

* Az adatkezelés feladata az Ambőba táblával kapcsolatos információk tárolása, valamint a betöltés/mentés biztosítása.
* A PotyogosAmobaTable osztály egy érvényes Amőba táblát.
* A hosszú távú adattárolás lehetőségeit az IPotyogosAmobaDataAccess interfész adja meg, amely lehetőséget ad a tábla betöltésére (LoadAsync), valamint mentésére (SaveAsync). A műveleteket hatékonysági okokból aszinkron módon valósítjuk meg.
* Az interfészt szöveges fájl alapú adatkezelésre a PotyogosAmobaFileDataAccess osztály valósítja meg. A fájlkezelés során fellépő hibákat a PotyogosAmobaDataException kivétel jelzi.
* A program az adatokat szöveges fájlként tudja eltárolni, melyek az stl kiterjesztést kapják. Ezeket az adatokat a programban bármikor be lehet tölteni, illetve ki lehet menteni az aktuális állást.
* A fájl első sora megadja a tábla méretét, a játékosok gondolkodási idejét, valamint a soron következő játkost. A fájl többi része izomorf leképezése a játéktáblának, azaz összesen n sor következik, és minden sor n számot tartalmaz szóközökkel választva. A számok 0-2 közöttiek lehetnek, ahol 0 reprezentálja a még üres mezőt.

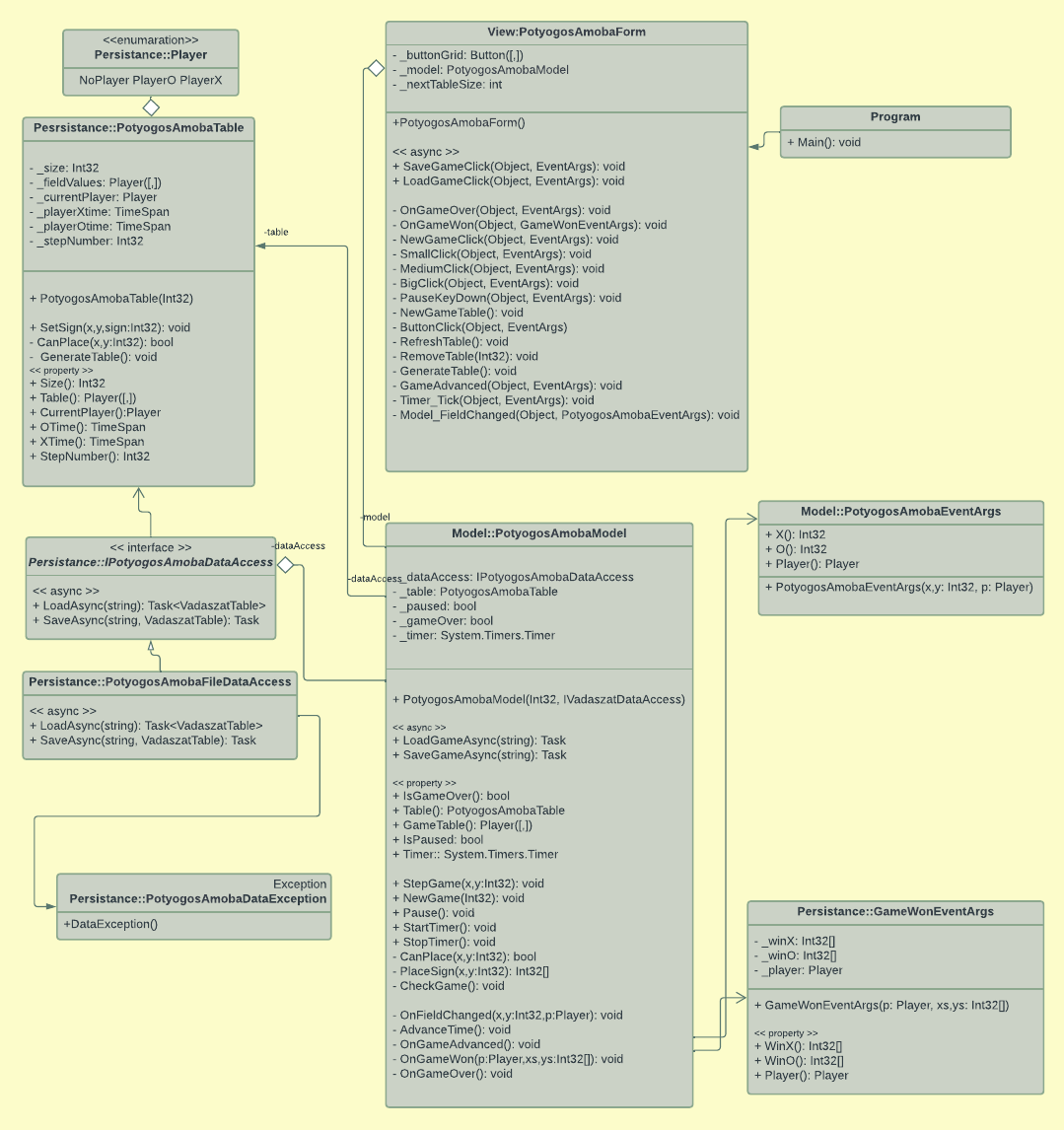
• **Modell**:

* + A modell lényegi részét a PotyogosAmobaModel osztály valósítja meg, amely szabályozza a tábla tevékenységeit, valamint a játék egyéb paramétereit, úgymint az idő (\_XTime) és a lépések (\_stepCount). A típus lehetőséget ad új játék kezdésére (NewGame), valamint lépésre (StepGame). Új játéknál megadható a kiinduló játéktábla mérete. Az idő előreléptetését időbeli lépések végzéséve (AdvanceTime) tehetjük meg.
  + A mezők állapotváltozásáról a FieldChanged esemény tájékoztat. Az esemény argumentuma (PotyogosAmobaFieldEventArgs) tárolja a megváltozott mező pozícióját.
  + A játékállapot megváltozásáról a GameAdvanced esemény, míg a játék végéről a GameOver esemény tájékoztat.
  + A képen szöveg, képernyőkép, Betűtípus, Téglalap látható

    Automatikusan generált leírásA modell példányosításkor megkapja az adatkezelés felületét, amelynek segítségével lehetőséget ad betöltésre (LoadGameAsync) és mentésre (SaveGameAsync)

**• Nézet:**

* A nézetet a PotyogosAmobaForm osztály biztosítja, amely tárolja a modell egy példányát (\_model), valamint az adatelérés konkrét példányát (\_dataAccess).
* A játéktáblát egy dinamikusan létrehozott gombmező (\_buttonGrid) reprezentálja. A felületen létrehozzuk a megfelelő menüpontokat, illetve státuszsort, valamint dialógusablakokat, és a hozzájuk tartozó eseménykezelőket. A játéktábla generálását (GenerateTable), illetve az értékek beállítását (RefreshTable) külön metódusok végzik.
* A játék időbeli kezelését egy időzítő végzi (\_timer), amelyet mindig aktiválunk játék során, illetve inaktiválunk, amennyiben bizonyos menüfunkciók futnak.

• A program teljes statikus szerkezete a 3. ábrán látható.

**Tesztelés:**

* A modell funkcionalitása egységtesztek segítségével lett ellenőrizve az AmobalTest osztályban.
* Az alábbi tesztesetek kerültek megvalósításra:
  + New10TableTest
  + New20TableTest
  + New30TableTest : E három új játék inditását, a konstruktorok helyes működését ellenőrzi az adott mérettel
  + StepGameTest: Játékbeli lépés hatásainak ellenőrzése, játék megkezdése előtt, valamint után. Több lépés végrehajtása azonos játékmezőn, esemény kiváltásának ellenőrzése.
  + GameWonTest: A Játék megnyerésének felismerését teszteli
  + LoadTest: A mentett játék hibamentes betöltése kerül ellenőrzésre