**Feladat**

Készítsünk programot, amellyel a következő kétszemélyes játékot lehet játszani. Adott egy mezőből álló tábla, ahol a két játékos bábúi egymással szemben helyezkednek el, két sorban (pont, mint egy sakktáblán, így mindkét játékos 2bábuval rendelkezik, ám mindegyik bábu ugyanolyan típusú). A játékos bábúival csak előre léphet egyenesen, vagy átlósan egy mezőt (azaz oldalra, és hátra felé nem léphet), és hasonlóan ütheti a másik játékos bábúját előre átlósan (egyenesen nem támadhat). Az a játékos győz, aki először átér a játéktábla másik végére egy bábuval.

A program biztosítson lehetőséget új játék kezdésére a táblaméret megadásával (6×6, 8×8, 10×10), és ismerje fel, ha vége a játéknak. Ekkor jelenítse meg, melyik játékos győzött, majd automatikusan kezdjen új játékot.

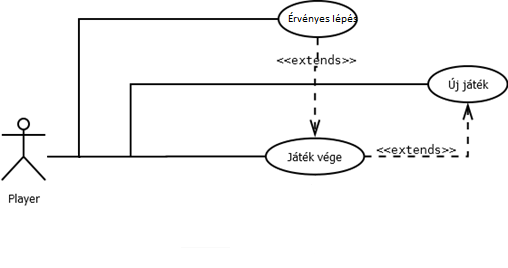
**Elemzés**

* A szoftvernek Windows operációs rendszeren kell futnia, ablakos felhasználói interfésszel kell rendelkeznie
* A program egy többszemélyes játék, azaz két játékosnak kell felváltva figurákkal lépkedni egy fix méretű táblán.
* Egy mezőnek tulajdonképpen három féle értéke lehet: üres, fekete- vagy fehér- figura áll rajta
* Kell tudni új játékot indítani, akár menet közben is, a pályaméret változhat megadott keretek között a felhasználó döntésétől függően
* A játék folyamán jelentős szerepe van annak a nem triviális kérdésnek, hogy ha egy játékos egy bizonyos mezőre lép, annak milyen következményei vannak:
  + ahhoz, hogy eldönthessük véget ér-e a játék, szükségünk van arra az információra, van-e értelmes lépése a játékosoknak
  + ahhoz, hogy eldönthessük, ki következik, ugyanígy tudni kell előre ezt az információt
  + ahhoz, hogy eldönthessük, egy lépés helyes-e egy olyan játékostól, akinek egyébként van helyes lépése, szintén tudnunk kell ezeket
* Nem igényel adatbázis-kapcsolatot, nem szükséges az adatokat se felolvasni, se menteni sehova. Nincs különösebb idő- és tárigénye

**Felhasználói esetek**

A játékosok felváltva léphetnek, minden lépés után kiértékelődik annak következménye (nyert-e valamelyik játékos). Ez járhat a játszma végével, ami pedig folytatódhat új játék kezdésével. Új játékot menet közben is lehet kezdeni.

A lehetőségeket az alábbi diagramon tudjuk áttekinteni:

****

**Tervezés**

**Osztályszerkezet**

* A model-view architektúrának megfelelően a projekt osztályai két diszjunkt halmazra bomlanak, az egyiket a model a másikat a view csomag jelöli
* A belépési pont egy a se nem model se nem view csomagba tartozó Launcher osztályban van, ez felel a két komponens létrehozásáért, felparaméterezéséért és a játék indításáért
* A view fő osztálya a GameFrame, ez kezdetben kap egy referenciát a model re. A különféle felhasználói interakcióból származó események bekövetkeztét ezen a referencián keresztül továbbítja a modelnek
* A GameFrame lesz a megjelenítendő ablak, ezért ez egy JFrame leszármazott
* 2 nagy részből áll: egy felső információkat megjelenítendő részből (InfoPanel) a központi rácsból, amin a játéktábla van (GamePanel) Tartalmaz még ezen felül egy menüt is, amin keresztül új játékot indíthatunk
* A menü, az InfoPanel és a GamePanel külön osztáylokba kerültek, amiket egyszer példányosítunk és adunk a frame-hez
* Minden modelbeli esemény a frame felé érkezik, ha az valamelyik fenti komponenst (is) érinti, a frame fogja azt propagálni a komponens felé
* A MenuBar két menüpontot tartalmaz: az egyik a model startNewGame() metódusát hívja, a másik megállítja a játék menetét.
* A model fő osztálya a GameEngine. Ez tartalmazza az adatok reprezentációját, ez fogadja a felületről érkező hatásokat és ez adja vissza azok hatásait

Az alábbi ábrán az entitás-osztályok kapcsolatait láthatjuk.

