

ELTE IK, Programtervező informatikus BSc

Eseményvezérelt Alkalmazások Fejlesztése 1.

Rosanics Márton, 2017

marton.rosanics@gmail.com

Feladat: Bombázó

Készítsünk programot, amellyel a következő játékot játszhatjuk.

Adott egy $n \times n$ mezőből álló játékpálya, amelyen az ellenfelek járőröznek. A játék célja, hogy a játékosunkat úgy irányítsuk, hogy az ellenfeleket bombák segítségével minél gyorsabban legyőzze.

A játékpályán a bejárható mezők mellett falak is elhelyezkednek, amelyekre nem léphetünk. Az ellenfelek adott időközönként lépnek egy mezőt (vízszintesen, vagy függőlegesen) úgy, hogy folyamatosan előre haladnak egészen addig, amíg falba nem ütköznek. Ekkor véletlenszerűen választanak egy új irányt, és arra haladnak tovább.

A játékos figurája kezdetben a bal felső sarokban helyezkedik el, és vízszintesen, illetve függőlegesen mozoghat (egyesével) a pályán, de ha találkozik valamely ellenféllel (ugyanarra a mezőre lép), akkor meghal.

A játékos bombát rakhat le az aktuális pozíciójára, amely rövid időn belül robban megsemmisítve a 3 sugáron belül (azaz egy 7×7 -es négyzetben) található ellenfeleket (falon át is), illetve magát a játékost is, ha nem menekül onnan.

A pályák méretét, illetve felépítését (falak, ellenfelek kezdőpozíciója) tároljuk fájlban, vagy hozzuk létre véletlenszerűen (előre rögzített paraméterek mellett).

A programot legalább 3 különböző méretű pályával lehessen használni.

A program biztosítson lehetőséget új játék kezdésére a pálya kiválasztásával, valamint játék szüneteltetésére (ekkor nem telik az idő, és nem léphet a játékos).

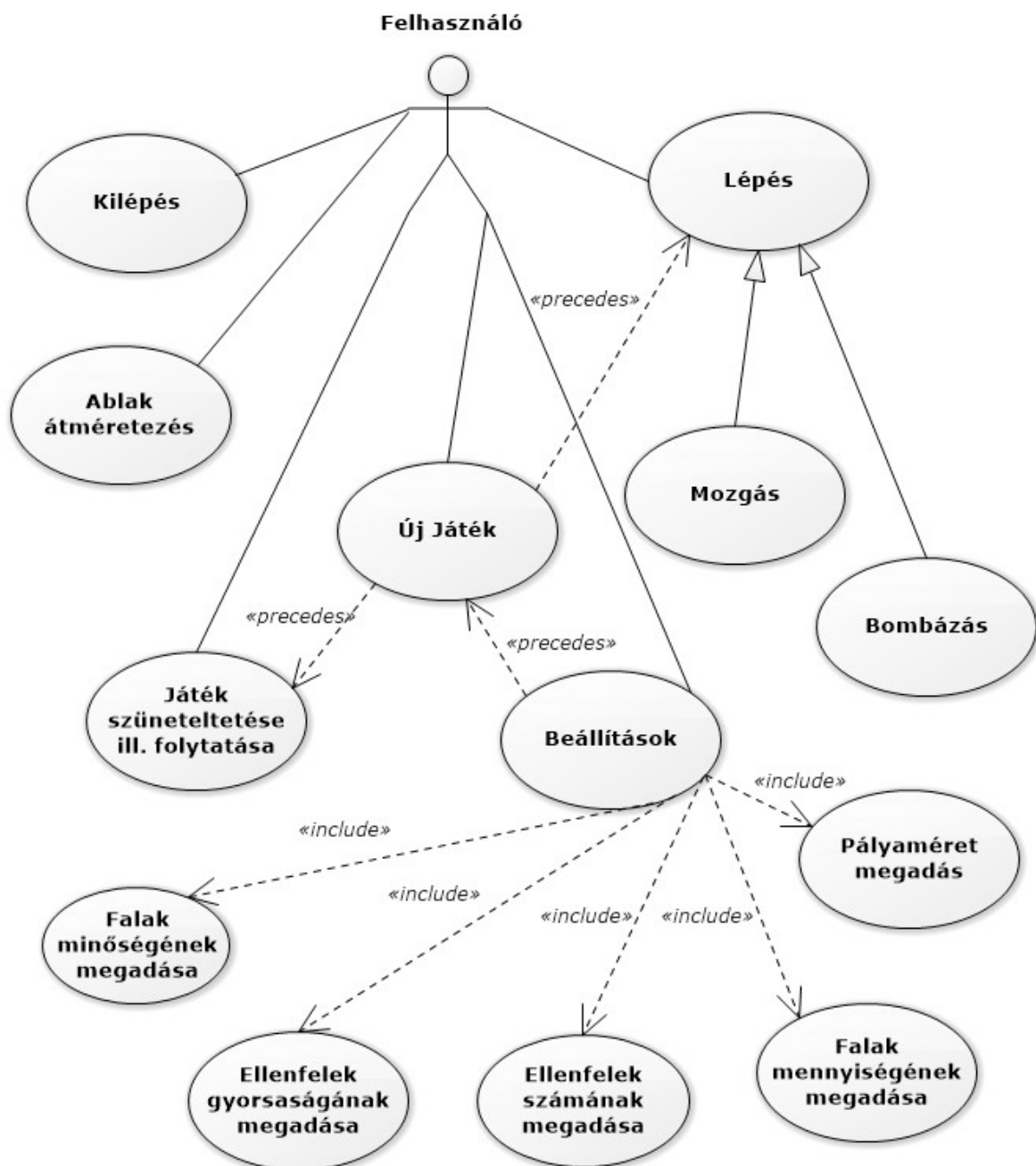
Ismerje fel, ha vége a játéknak, és jelezze, győzött, vagy veszített a játékos. A program játék közben folyamatosan jelezze ki a játékidőt, valamint a felrobbantott ellenfelek számát.

Elemzés

- Az alkalmazást egy grafikus felületen fogom megjeleníteni, ahol a játékbeállításokat csúszkákkal, játék indítást és szüneteltetést pedig nyomógombokkal érheti el a játékos.
- A játéktér négyzetes területet fog elfoglalni, így a képernyőn fennmaradó függőleges sávon (fix minimális mérettel) elhelyezhető majd az információs sáv.
- Véletlenszerű fal- és ellenfél kezdőpozíciókkal, illetve a szabadon állítható pályamérettel adódik, hogy a feladat leírásában megszabott "több különböző pályán" lehet majd játszani.
- A feladatleírás nem határozza meg, hogy a falak lerombolhatóak-e bombázás során, de extra funkcionalitásként ez is implementálom.

- Az alkalmazás tartalmazni fog egy-két ikont is, melyek erőforrásként csatolva lesznek a projecthez. (Célpont- és robbanásjelzők.)
- Külön jelölők kellene a falaknak, a padlónak, a játékosnak és az ellenfeleknek is. Ezek legegyszerűbben különböző színekkel oldhatóak meg. Esetleges UI-update esetére érdemes viszont a lépés irányát is eltárolni a modellben a játékosnál és az ellenfeleknél (hiszen máshogy nézhetnek ki ha jobbra/balra/felfelé/lefelé haladnak.)

Felhasználói esetek:



Felhasználói eset	Leírás	
Alkalmazás indítása	GIVEN:	Az alkalmazás telepítve van.
	WHEN:	Az alkalmazást elindítja
	THEN:	Üres játékfelület (tábla nélkül) megjelenik.
Kilépés	GIVEN:	Az alkalmazás el van indítva.
	WHEN:	Játékfelület ablakának bezáró ikonjára kattint.
	THEN:	Alkalmazás befejezése.
Beállítások (pályaméret, falak- és ellenfelek száma, ellenfelek gyorsasága, falak rombolhatósága)	GIVEN:	Az alkalmazást elindította.
	WHEN:	A jobb oldali panelen a csúszkákon állít, vagy a rádiógombra kattint.
	THEN:	A következő új játék indítása a megadott paraméterek szerint fog történni.
Új játék	GIVEN:	Az alkalmazást elindította.
	WHEN:	A 'Launch Mission' nyomógombra kattint.
	THEN:	Új játéktábla jelenik meg. A meccs automatikusan elindul.
Játék szüneteltetése	GIVEN:	A játék folyamatban van.
	WHEN:	A 'Freeze time!' gombra kattint.
	THEN:	Az időszámláló megáll, az ellenfelek nem haladnak. A játékos lépni és bombát hívni nem tud ilyenkor.
Játék folytatása	GIVEN:	A játék szüneteltetve van.
	WHEN:	Az 'Unfreeze time!' gombra kattint.
	THEN:	A játék folytatódik.
Játékos mozgatása	GIVEN:	A játék folyamatban van.
	WHEN:	A billentyűzet iránygombjainak valamelyikét lenyomja.
	THEN:	A játékos a megfelelő irányba ellép, ha ott nincs fal.
Célpont kiválasztása	GIVEN:	A játék folyamatban van.
	WHEN:	A 'Space' billentyűt lenyomja.
	THEN:	A játékos jelenlegi pozíciója meg lesz jelölve, mely a -kis késleltetéssel érkező- bombázott terület középpontja lesz.
Játék vége (vereség)	GIVEN:	A játék folyamatban van.

	WHEN:	A játékos a bombázott területek valamelyikén áll (vagy bombázás során odalép), vagy pedig egy ellenféllel ugyan azon a mezőn tartózkodik.
	THEN:	A játék véget ér.
Játék vége (győzelem)	GIVEN:	A játék folyamatban van.
	WHEN:	Az ellenfelek száma eléri a nullát (azaz az utolsó ellenfél is egy olyan helyen tartózkodik, ami bombázás alatt van).
	THEN:	A játék véget ér.

Rendszerterv:

A program szerkezetét két rétegre bontjuk modell/nézet architektúrának megfelelően. A modell eseményeken keresztül kommunikál a nézettel.

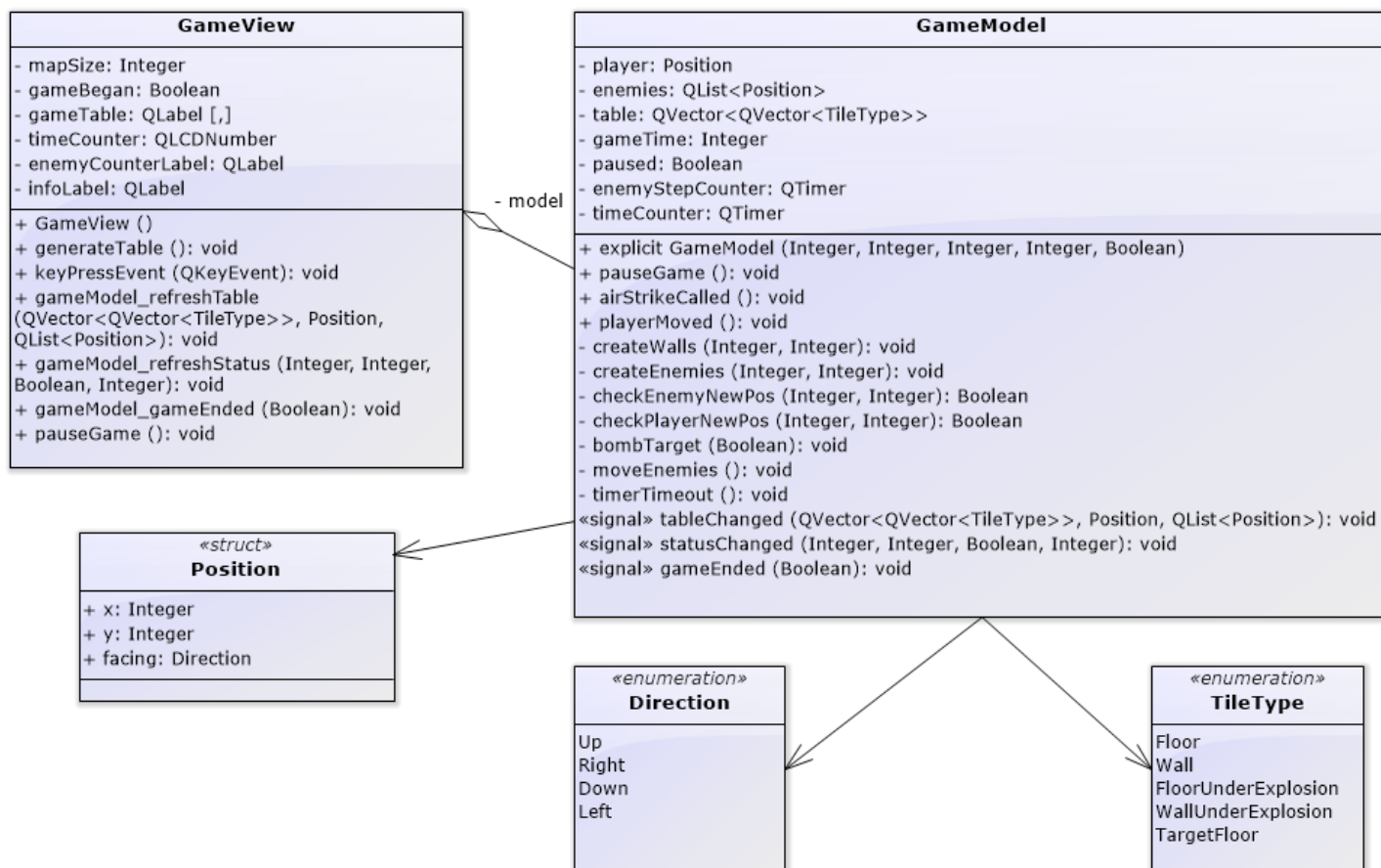
A modellt a *GameModel* osztály valósítja meg, mely a játéklogikát biztosítja.

- Az eseménykezelés megvalósítása érdekében az osztályt a *QObject*-ból származtatjuk.
- A model generálja le a játéktáblát, az ellenfelek és a falak véletlenszerű pozícióját. Itt történik az időszámlálás, és az ellenfelek mozgatása is.
- A fentiekén kívül természetesen a lépéseket (bevitt utasításokat) is a model dolgozza fel és végzi el, miközben leellenőrzi, hogy végrehajtható-e az adott utasítás.
- A model háromféle signal-t képes küldeni: *tableChanged*, mely a nézetnek elküldi a játéktábla aktuális állását, *statusChanged*, mely az információs sávnak küld adatokat, és a *gameEnded*, mely a játék végét jelzi.

A megjelenítést a *GameView* biztosítja, mely a *QWidget* leszármazottja. Ez a réteg kezeli az eseményeket is.

- Az alkalmazás 2 fő részből áll: a játéktáblából és az információs sávból. A játéktábla kezdetben üres, csak új játék indításakor telik meg.
- Az információs sávban találhatóak a játék beállításai, valamint két gomb, amikkel új játékot lehet kezdeni, ill. szüneteltetni lehet a folyamatban lévő menetet. Itt található az időszámláló is, valamint kijelzi a legyőzött ellenfelek számát. Célpont kiválasztásakor itt van feltüntetve a bomba érkezéséig hátramaradt idő. A játék végén szintén ugyanitt van feltüntetve, hogy a játékos győzött vagy veszített-e.
- A *GameView* lekezeli a *GameModel* által küldött signal-okat, valamint egyéb beviteli eseményeket (iránygombok lenyomása, space lenyomása, ablak átméretezése).

Osztályszerkezet:



Megjegyzés: A fenti osztálydiagram nem tartalmazza az osztályok összes adattagját és metódusát, csupán áttekintést biztosít a fontosabbak feltüntetésével.

Eseményvezérlés (A GameView osztály eseményei.)

Esemény	Leírás
<i>setWallNumberText</i> <i>setEnemyNumberText</i> <i>setEnemySpeedText</i>	Az adott csúszka állításakor a csúszka fölött lévő számállás frissítése. (Kiírja, mennyi a csúszka értéke.)
<i>setSliderMaxValues</i>	A pályaméretet állító csúszka módosításakor az afelett lévő számállás frissül. A pályaméret függvényében le lesz korlátozva a többi csúszka maximum értéke. (Ez biztosítja a megfelelő bemenetet.)
<i>generateTable</i>	A 'Launch Mission' gomb lenyomásakor a megadott paraméterekkel létrehoz egy új játékot. (Meghívja az új paraméterekkel a modell konstruktorát.) Legenerálja a játéktér mezőit, és frissíti az állást.
<i>keyPressEvent</i>	Billentyűzet iránygombjainak, vagy szóköz billentyűjének lenyomása során továbbítja az input-ot a modellnek.

<i>gameModel_refreshTable</i>	A modell által küldött jel után a játéktábla tartalmának frissítése.
<i>gameModel_refreshStatus</i>	A modell által küldött jel után az információs sáv tartalmának frissítése.
<i>gameModel_gameEnded</i>	A modell által küldött jel után az információs sáv tartalmának frissítése. Játékszüneteltetés lehetőségének tiltása.
<i>pauseGame</i>	A játék szüneteltetéséért felelős gomb lenyomása során a gomb feliratának megváltoztatása, valamint a modell értesítése történik.
<i>resizeEvent</i>	Ablak átméretezése során az ablak tartalmát az új mérethez igazítja.

Tesztelés

Az alkalmazás mappájában található *bomberTest* project tartalmaz egységteszteket.

Az alábbi funkcionálisok voltak (sikeresen) tesztelve:

- Játéktábla generálása, játékos kezdőpozíciója. Játékos el tud indulni.
- Játékos minden irányba tud lépni, kivéve ha falba ütközik.
- Célpontot ki lehet választani, késleltetés után megtörténik a robbanás.
- Robbanástól a játékos meg tud halni, akkor is ha a robbanás elején tartózkodik a zónában, és akkor is ha közben lép csak be.
- Robbanástól az ellenfél meg tud halni.
- Ellenféltől a játékos meg tud halni.