

11. Prototípus

66 – *[simon_balazst_szeretnenk_konzulensnek]*

Konzulens:

Simon Balázs

Csapattagok:

Kiss Andor	TXC54G	kissandor4@gmail.com
Konrád Márk	JSPDME	konrad0816@gmail.com
Glávits Balázs Róbert	NMZC9G	glavits.balazs@gmail.com
Máté Botond	ELOYOV	m.botond7@gmail.com
Lant Gábor	P35E36	lant.gabor98@gmail.com

2020. május 3.

Tartalomjegyzék

11 Grafikus felület specifikációja	4
11.1 A grafikus interfész	4
11.2 A grafikus rendszer architektúrája	6
11.2.1 A felület működési elve	6
11.2.2 A felület osztály-struktúrája	7
11.3 A grafikus objektumok felsorolása	7
11.3.1 Osztály1	7
11.3.2 Osztály2	7
11.4 Kapcsolat az alkalmazói rendszerrel	8
11.5 Napló	8

Ábrák jegyzéke

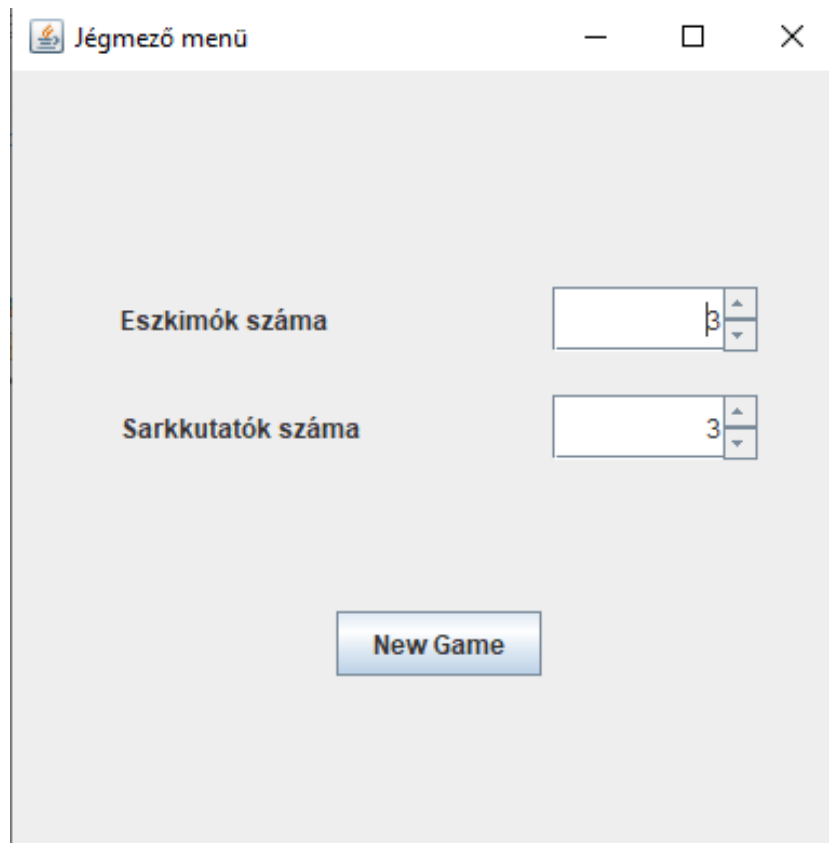
11.1 A játék menüjét bemutató ábra	4
11.2 Kép a játékról, eszközökkel	5
11.3 Kép a játékról, eszközök nélkül	5
11.4 Kép a játékban a hóviharról	6

11. Grafikus felület specifikációja

11.1. A grafikus interfész

[A menürendszer, a kezelői felület grafikus képe. A grafikus felület megjelenését, a használt ikonokat, stb screenshot-szerű képekkel kell bemutatni. Az építészetben ez a homlokzati terv.]

A játék indításakor az alábbi menü jön be:



11.1. ábra. A játék menüjét bemutató ábra

A két Spinner segítségével kiválasztható a játékosok száma, kutatókra és eszkimókra lebontva. A New Game gomb segítségével indítható el a játék, ami működés közben valahogy így néz ki:



11.2. ábra. Kép a játékról, eszközökkel



11.3. ábra. Kép a játékról, eszközök nélkül

A képen láthatók a játékosok jobb oldalt, akik részt vesznek a játékban. Az ikonjuk alapján megkülönböztethetők egyértelműen. Az ikon alatt pirossal a testhő, sárgával a játékos energiája látható. Az aktuálisan kiválasztott játékos, akit zöld keret vesz körül, lehet irányítani, és láthatóak a tárgyai a bal oldalon. Az éppen használatban lévő tárgyakat szintén zöld keret veszi körül. Az élelem és sátorzacskó fogyócikek fölött számláló helyezkedik el, ami a kiválasztott játékosnak éppen birtokában lévő mennyiséget jelzi belőlük. A 2020. május 3.

tárgyak között látható törékeny falapát, míg törhetetlen lila kristálylapát is. A képernyő alján az akciógombok helyezkednek el, ezekkel lehet a modellben specifikált use-casek szerint irányítani az éppen kiválasztott játékost. Emellett ha már nem szeretne akciót végezni, véget vethet körének a játékos. A játékból való kilépés és a hang kikapcsolása is ezen a gombsoron kapott helyet. A képernyő közepén található a lényeg, maga a játék megjelenítése. A jégmezőt a hideg, fagyos óceán veszi körül, míg a tükörsima jégen félelmetes jegesmedve szörny, és tőle rettegő, onnan menekülni vágyó játékosok láthatók. Kapucniban az iglulakó eszkimók találhatók, míg sapkában a sarkkutatók fedezik fel a jégtáblák rejtelseit. Felfedezett jégtábláik teherbírását zászlóval jelölik: zölddel, ha az nem törhet el, míg pirossal és egy számmal, ha az eltörhet a számot meghaladó játékosok lába alatt. A jégen található tárgyak is, sátorzacskó, törhetetlen lapát, emellett vannak épületek is lerakva, iglu és sátor is. A jégmezőn található hóval nem fedett vizesgödör, hóval fedett és sima jégtábla is. A hőmennyiség az üres jégtől kezdve 5 réteg hóig terjedhet, mindegyik látható a pillanatképünkön. A jégmezőn néha feltámad a hóvihár, ez az alábbihoz hasonló animációval lesz majd megjelenítve:



11.4. ábra. Kép a játékban a hóviharról

11.2. A grafikus rendszer architektúrája

[A felület működésének elve, a grafikus rendszer architektúrája (struktúra diagramok). A struktúra diagramokon a prototípus azon és csak azon osztályainak is szerepelnie kell, amelyekhez a grafikus felület létrehozó osztályok kapcsolódnak.]

11.2.1. A felület működési elve

*[Le kell írni, hogy a grafikai megjelenésért felelős osztályok, objektumok hogyan kapcsolódnak a meglévő rendszerhez, a megjelenítés során mi volt az alapelv. Törekedni kell az MVC megvalósításra. Alapelvek lehetnek: **push** alapú: a modell értesíti a felületet, hogy változott; **pull** alapú: a felület kérdezi le a modellt, hogy változott-e; **kevert**: a kettő kombinációja.]*

A feladat megvalósítása során törekedtünk az MVC architektúrára. A modellt a korábbi fázisokban elkészült program biztosítja. A modell önmagában csak adatokat szolgáltat ezért minden adata hozzáférhető getterek segítségével. A View pullolja a modell adatait, míg a Controller pusholja azt. A modell önmagában nem observable, mert minden változást a Controller vált ki futása során.

A View egy pálya nézet. Megkapja a Modellt mint dependency injection. Update hatására kiolvassa a Modell állapotát és megjeleníti azt. A View implementálja a GameObservert.

A Controller felelős a UI elemek kezeléséért. Megkapja a Modell-t és a View-t, mint dependency injection. Feliratkozhat a View eventjeire, ha az is irányítható. Fontos feladata az felhasználótól érkező inputok lekezelése. Írja a modell állapotát, frissíti a View-t.

A főprogram felelős a játék főmenüjének megjelenítéséért és az ehhez kapcsolódó inputok lekezeléséért. A főmenüben meg lehet adni a játékosok számát, neveit. Modellt épít a Proto-val. Implementálja a GameObserver-t, ezzel képes a játék végét kezelni. View-t és Controller-t nyit és zár.

11.2.2. A felület osztály-struktúrája

[Osztálydiagram. Minden új osztály, és azon régiek, akik az újakhoz közvetlenül kapcsolódnak.]

11.3. A grafikus objektumok felsorolása

[Az új osztályok felsorolása. Az régi osztályok közül azoknak a felsorolása, ahol változás volt. Ezek esetén csak a változásokat kell leírni.]

11.3.1. Osztály1

- Felelősség
[Mi az osztály felelőssége. Kb 1 bekezdés. Ha szükséges, akkor state-chart is.]
- Ősosztályok
*[Mely osztályokból származik (öröklési hierarchia)
Legősebb osztály → Ősosztály2 → Ősosztály3...]*
- Interfészek
[Mely interfészeket valósítja meg.]
- Attribútumok
[Milyen attribútumai vannak]
 - attribútum1: attribútum jellemzése: mire való, láthatósága (UML jelöléssel), típusa
 - attribútum2: attribútum jellemzése: mire való, láthatósága (UML jelöléssel), típusa
- Metódusok
[Milyen publikus, protected és privát metódusokkal rendelkezik. Metódusonként precíz leírás, ha szükséges, activity diagram is a metódusban megvalósítandó algoritmusról.]
 - int foo(Osztály3 o1, Osztály4 o2): metódus leírása, láthatósága (UML jelöléssel)
 - int bar(Osztály5 o1): metódus leírása, láthatósága (UML jelöléssel)

11.3.2. Osztály2

- Felelősség
[Mi az osztály felelőssége. Kb 1 bekezdés. Ha szükséges, akkor state-chart is.]
- Ősosztályok
*[Mely osztályokból származik (öröklési hierarchia)
Legősebb osztály → Ősosztály2 → Ősosztály3...]*
- Interfészek
[Mely interfészeket valósítja meg.]

- Attribútumok

[Milyen attribútumai vannak]

- attribútum1: attribútum jellemzése: mire való, láthatósága (UML jelöléssel), típusa
- attribútum2: attribútum jellemzése: mire való, láthatósága (UML jelöléssel), típusa

- Metódusok

[Milyen publikus, protected és privát metódusokkal rendelkezik. Metódusonként precíz leírás, ha szükséges, activity diagram is a metódusban megvalósítandó algoritmusról.]

- int foo(Osztály3 o1, Osztály4 o2): metódus leírása, láthatósága (UML jelöléssel)
- int bar(Osztály5 o1): metódus leírása, láthatósága (UML jelöléssel)

11.4. Kapcsolat az alkalmazói rendszerrel

[Szekvencia-diagramokon ábrázolni kell a grafikus rendszer működését. Konzisztens kell legyen az előző alfejezetekkel. Minden metódus, ami ott szerepel, fel kell tűnjön valamelyik szekvenciában. Minden metódusnak, ami szekvenciában szerepel, szereplnie kell a valamelyik osztálydiagramon.]

11.5. Napló

Kezdet	Időtartam	Résztevők	Leírás
2010.03.21. 18:00	2,5 óra	Horváth Németh Tóth Oláh	Értekezlet. Döntés: Horváth elkészíti az osztálydiagramot, Oláh a use-case leírásokat.
2010.03.23. 23:00	5 óra	Németh	Tevékenység: Németh implementálja a tesztelő programokat.
...