# Követelmény, projekt, funkcionalitás

## Bevezetés

### Cél

A dokumentáció ismerteti a projekt alapvető felépítését és követelményeit. Meghatározza a munka során követendő irányelveket.

### Szakterület

Az elkészítendő szoftver a híres Sokoban nevű játékprogram, így kifejezett szakterülete nincs, bárki játszhat vele. Egyetlen célja a végfelhasználók szórakoztatása. Ez egy logikai játék - ráadásul elég nehéz, NP komplexitású - kevés akcióval, gondolkodni szerető játékosokat céloz meg.

### Definíciók, rövidítések

**aktor:** a használati esetekben külső félként résztvevő szereplők által játszott összetartozó szerepek együttese

**BME:** Budapesti Műszaki és Gazdaságtudományi Egyetem

**Git:** egy verziókezelő rendszer, ami arra szolgál, hogy fájlok (programok, dokumentációk stb.) különböző verzióit kordában tartsa, elkönyvelje, tárolja és megossza

**GitHub:** egy Git-re épülő internetes szolgáltatás

**IIT:** Irányítástechnika és Informatika Tanszék

**JDK:** Java Development Kit-Java fejlesztőeszköz

**JRE:** Java Runtime Environment - Java futtatókörnyzet

**NP komplexitás:** egy probléma megoldásának nehézségét jellemzi, ez a kategória kifejezetten bonyolult

**proto:** prototípus - Olyan állapota a programnak, amikor amikor a grafikus felületen kívül minden belső működés meg van valósítva

**szoftver:** számítógépen futtatható program

**szkeleton:** olyan állapota a programnak, amikor még csak a belső felépítése van kész

**UML:** Unified Modeling Language - modellező eszköz

**Use-case**: leírja a rendszer és az őt felhasználó külső szereplők (aktorok) közötti akciók és reakciók (válaszok) sorozatát, az interakciókat

### Hivatkozások

Szoftver projekt laboratórium - https://www.iit.bme.hu/targyak/BMEVIIIAB02

### Összefoglalás

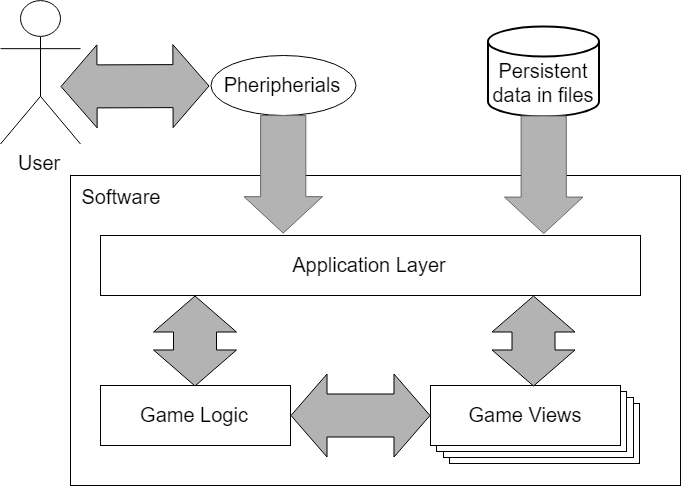
A dokumentum további részeiben található:

* Áttekintés, ami nagyvonalakban bemutatja a szoftvert
* Követelmények leírása, melyek alapján később a szoftver elkészül
* Use-case-ek felsorolása
* Szótár a nem közérthető szavak, illetve nem hagyományos értelemben használt szavak jelentésével
* Projekt megvalósításának a terve
* Napló a végzett munkáról

## Áttekintés

### Általános áttekintés

A szoftver legmagasabb szintű képe, mely esetünkben követi a Game Coding Complete c. könyvben (Mike McShaffry,‎ David Graham, ISBN: 978-1133776574) - és rengeteg játékban és játékmotorban - prezentált architektúrát. Ez a fajta felbontás nagy rugalmasságot nyújt a rendszernek.



Megjegyzés: Az esemény és üzenet ezen kontextusban ekvivalens az architektúra szempontjából.

**User**: Játékos, játékosok, akik a szoftvert használják.

**Pheripherials (perifériák)**: A számítógéphez kapcsolódó eszközök, melyek audiovizuálisan prezentálják a játékot (monitor, hangszoró) vagy a játék irányítását teszik lehetővé (billentyűzet).

**Persistent data in files**: Minden játékadat, amelyet fájl(ok)ban tárol a szoftver, mint a pályák vagy grafikus elemek.

**Application Layer**: A szoftver azon része, mely a perifériákkal és a game view-kkal tartja a kapcsolatot. Betölt file-okat, billentyűzetről olvas és a rajzolás legalacsonyabb szintjét végzi (operációs rendszer szinten). A perifériáktól kapott jelzéseket, például billentyű lenyomást elküldi a Game View-knak üzenetként.

**Game Views**: A játékra egy adott nézet. Ilyen nézet lehet a grafikai megjelenítés, a játékosok irányítása stb. Ha például az irányítás nézet megkapja az első játékos balra mozgató billentyűjének az üzenetét, akkor az átfordítja “Játékos 1 balra lép” üzenetté, és elküldi a Game Logicnak.

Elméletileg, ha készítenénk egy View-t, ami nem csinál mást, mint rögzíti a perifériális eseményeket, és egy másik View pedig ezeket időrendben lekezeli, létrehoztunk egy egyszerű visszajátszási rendszert, amely hibajavításnál például jól jöhet.

**Game Logic**: A játék funkcionalitása, logikája. A Game View-któl kapott események alapján alakítja a játékot (lépteti a játékost stb.) illetve ha valami vizuális frissítésre szorul, üzenettel jelzi a Game View-nak. Az Application Layerhez üzenettel is fordulhat, ha például perzisztens adat betöltésére van szükség, mert véget ért a pálya és a következőre kell lépni.

### Funkciók

A játék a híres Sokoban továbbgondolt változata. Egy raktárépületben ládákat tárolnak. A raktárépület padlója négyzet alakú mezőkre van osztva, ezeken állnak a mezőkkel megegyező alapterületű ládák. A padlón kívül kétféle építőelemet találunk még, falat és oszlopot.

A raktárban többen dolgoznak, egy játékos egy adott dolgozót irányít. Pontosan annyi dolgozó kezd a raktárban, mint ahány játékos játssza a játékot. Játékosból 2-4 lehet. Egy dolgozó mezőről mezőre haladhat az oldalak mentén, de kizárólag a padló típusú mezőkön, falon, oszlopon vagy másik dolgozó kiindulási mezőjén nem haladhat át. Egy mezőn egyszerre csak egy dolgozó tartózkodhat.

Egyes padló típusú mezőkön a játék kezdetén ládákat találunk, melyek szintén mezőről mezőre mozoghatnak az oldalak mentén, a padló típusú mezőkön. Mozgatásukhoz a dolgozóknak kell eltolniuk ezeket: ha egy dolgozó egy olyan padlóra akar lépni, melyen láda van és a láda képes ugyanazon irányban továbbmozogni (azaz nincs az abban az irányban lévő szomszédos mezőn fal, oszlop vagy egy játékos kiindulási mezője), amerre a dolgozó lépne, a ládát lépés közben eltolja arra a mezőre, különben a dolgozó nem léphet arra. A ládák egy ilyen eltolással mindig csak egy szomszédos mezőre kerülhetnek. A ládákat húzni nem lehet, így előfordulhat, hogy egy láda beragad és azt többé mozgatni nem lehet.

A cél a ládák előírt helyre tologatása. Az előírt helyek olyan padló típusú mezők, melyek meg vannak jelölve. Ha egy dolgozóra ládát tolunk, akkor a dolgozó automatikusan a szomszédos mezőre tolódik. Ha nem tud eltolódni (a tolás a dolgozót falba, oszlopba vagy másik játékos kiindulási mezőjére léptetné), a dolgozó életet veszít és visszakerül a kiindulási helyére. A ládák nem nyomhatók össze.

Ha egy vonalban közvetlen egymás után több láda vagy dolgozó - akár mindkettő vegyesen - áll és a legtávolabbi eltolható az adott irányba, akkor az egész lánc eltolható. A dolgozók közvetlenül nem tolhatják el egymást.

A padlón egyes helyeken lyukak találhatók, amelyekre ládát tolva a láda leesik (eltűnik), és nem kerül vissza a játékba. Ha dolgozó lép rá, vagy rátolják, életet veszít és visszakerül a kiindulási helyére. Némelyik lyuk csak akkor viselkedik lyukként, ha egy kapcsolón láda áll, egyébként padlónak tűnik. Ha egy kapcsolóra láda kerül, a kapcsoló kinyit egy vagy több lyukat. Amint a láda lekerül a kapcsolóról, a kapcsoló által kinyitott lyukak bezáródnak. Ha munkás áll a kapcsolóra, akkor nem kapcsol.

Minden dolgozó rendelkezik egy számára kijelölt kiindulási mezővel. A dolgozók itt kezdik a játékot, valamint innen folytatják életvesztés esetén is. Minden kiindulási mezőn csak az ahhoz tartozó dolgozó tartózkodhat és láda sem tolható rá.

Minden dolgozó három élettel kezd, ha valakinek ez elfogyott, az kikerül a játékból és az őt irányító játékos veszít. Egy játékos egy láda helyére tolásával egy pontot szerez. Ha egy játékos eltol a helyéről egy ládát, azzal a helyére juttató játékos - akár önmaga - pontszámát eggyel csökkenti. Ha egy játékos megmozgat úgy egy láncot, hogy a láncban valahol egy vagy több dolgozót tol meg, ezáltal a helyére tol egy vagy több ládát, akkor az így szerzett pontokat az egész láncot mozgató játékos kapja, így lehet pontot lopni.

Egyes ládák szívecskével vannak megjelölve. Ezek a ládák ugyanúgy használhatók pontszerzésre és a játék céljának elérésére, mint a hagyományos ládák, de lyukba lökésük eggyel megnöveli a mozgató dolgozó életének számát. Az életszerzés elve azonos a pontszerzés elvével, vagyis, ha a láda lyukba lökése egy láncban és nem közvetlenül történik, akkor is a láncot mozgató dolgozó szerzi az életet.

Ha az összes játékban lévő láda (ami előfordulhat, hogy több, de az is, hogy kevesebb, mint amennyi előírt hely van) egy-egy előírt helyre került, vagy már nincsen mozgatható láda, a játék azonnal véget ér. Ilyenkor az a játékos nyer, aki a legtöbb pontot szerezte. Amennyiben több játékos (akár az összes) is azonos pontszámmal rendelkezik a játék végén, közöttük holtverseny alakul ki. A játék akkor is véget ér, ha már csak egy dolgozó marad, mert a többi meghalt. Ilyenkor automatikusan az utolsónak megmaradt dolgozó játékosa nyer, a ládák helyzetétől és a többi játékos pontszámától függetlenül.

A játék a kezdés pillanatától folyamatosan zajlik, vagyis nincsen körökre osztva, a dolgozók tetszőleges időpillanatban mozoghatnak. A játékban időkorlát nincsen, akár a végtelenségig is játszható.

### Felhasználók

Használatához nincs szükség szakértelemre vagy előképzettségre, a játék lényege percek alatt megtanulható, és a gondolkodni - vagy egymást szétnyomni - vágyókat pedig rövid ideig elszórakoztathatja.

### Korlátozások

A szoftvernek a JDK beépített könyvtárait kell használnia a funkciók megvalósításához és az elkészült programnak futnia kell a sztenderd JRE futtatókörnyezetben.

### Feltételezések, kapcsolatok

Szoftver projekt laboratórium tárgyhonlap, ahol megtalálható a feladatkiírás, a dokumentáció sablonjai, továbbá a leadás ütemezése.

*https://www.iit.bme.hu/targyak/BMEVIIIAB02*

Szoftver projekt laboratórium ideiglenes oldal, ahol elérhetőek az imént említett segédanyagok a tárgyholnap elérhetetlensége esetén.

*http://devil.iit.bme.hu/~balage/projlab/*

## Követelmények

### Funkcionális követelmények

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Use-case** | **Komment** |
| 1.01 | Vannak ládák | bemutatáskor | alapvető | megrendelő | View warehouse |  |
| 1.02 | A pálya négyzet alakú mezőkre van osztva | bemutatáskor | alapvető | megrendelő | View warehouse |  |
| 1.03 | Vannak falak és oszlopok | bemutatáskor | alapvető | megrendelő | View warehouse |  |
| 1.04 | A játékosok dolgozókat irányítanak | bemutatáskor | alapvető | megrendelő | Move worker |  |
| 1.05 | A játékot egyszerre több játékos játssza | bemutatáskor | alapvető | megrendelő | Move worker |  |
| 1.06 | A dolgozók a padlókón mozoghatnak | bemutatáskor | alapvető | megrendelő | Move worker |  |
| 1.07 | Egy mezőn egyszerre csak egy dolgozó állhat | bemutatáskor | fontos | csapat | Move worker |  |
| 1.08 | A dolgozók nem tolhatják el közvetlenül egymást | bemutatáskor | fontos | megrendelő | Move worker |  |
| 1.09 | A ládák falra, oszlopra és kiindulási mezőkre nem tolhatók | bemutatáskor | alapvető | megrendelő | Push crates |  |
| 1.10 | A dolgozók eltolhatják a ládákat | bemutatáskor | alapvető | megrendelő | Move worker, Push crates |  |
| 1.11 | A ládák beragadhatnak | bemutatáskor | fontos | csapat | Push crates |  |
| 1.12 | Cél a ládák előírt helyre mozgatásával minél több pontot szerezni | bemutatáskor | alapvető | megrendelő | Push crates |  |
| 1.13 | Egy eltolt láda másik ládát és dolgozót is tovább tolhat | bemutatáskor | fontos | megrendelő | Push workers, Push crates |  |
| 1.14 | A dolgozók falra, oszlopra vagy másik dolgozó kiindulási mezőjére tolva életet veszítenek | bemutatáskor | fontos | megrendelő | Move worker, Push crates, Push workers | A megerendelő esetében meghalnak |
| 1.15 | A ládák nem nyomhatók össze | bemutatáskor | alapvető | megrendelő | Push crates |  |
| 1.16 | A dolgozók ládákat és más dolgozókat láncban is eltolhatnak | bemutatáskor | fontos | csapat | Move worker, Push crates, Push workers |  |
| 1.17 | Vannak lyukak | bemutatáskor | fontos | megrendelő | View warehouse |  |
| 1.18 | A ládák lyukra tolva leesnek | bemutatáskor | fontos | megrendelő | Push crates |  |
| 1.19 | A dolgozók lyukra lépve életet vesztenek | bemutatáskor | fontos | megrendelő | Move worker | A megrendelő esetében meghalnak |
| 1.20 | Életvesztés esetén a dolgozók visszakerülnek a kiindulási mezőjükre | bemutatáskor | fontos | csapat | Move worker |  |
| 1.21 | A ládák kapcsolóra tolva kinyitnak egy vagy több lyukat | bemutatáskor | fontos | megrendelő | Push crates, Control switches |  |
| 1.22 | A ládák kapcsolóról eltolva becsuknak egy vagy több, általuk kinyitott lyukat | bemutatáskor | fontos | megrendelő | Push crates, Control switches |  |
| 1.23 | Minden dolgozó saját kiindulási mezőről indul | bemutatáskor | alapvető | csapat | Start game |  |
| 1.24 | A dolgozók három élettel kezdenek | bemutatáskor | opcionális | csapat | Start game |  |
| 1.25 | A játékos veszít, ha a dolgozójának nincs több élete | bemutatáskor | alapvető | csapat | Finish game |  |
| 1.26 | A játékosok pontot szereznek egy láda helyre tolásával | bemutatáskor | alapvető | megrendelő | Move worker, Push crates |  |
| 1.27 | A játékosok pontot veszíthetnek | bemutatáskor | opcionális | csapat | Move worker, Push crates |  |
| 1.28 | A játékosok pontot lophatnak | bemutatáskor | fontos | csapat | Move worker, Push crates |  |
| 1.29 | Egyes ládák szívecskével vannak megjelölve | bemutatáskor | opcionális | csapat | View warehouse |  |
| 1.30 | A dolgozók életet szerezhetnek szívecskével jelölt láda lyukba tolásával | bemutatáskor | opcionális | csapat | Move worker, Push crates |  |
| 1.31 | Ha minden láda előírt helyen van, a játéknak vége | bemutatáskor | alapvető | megrendelő | Push crates |  |
| 1.32 | Ha nincs mozgatható láda, a játéknak vége | bemutatáskor | alapvető | megrendelő | Push crates |  |
| 1.33 | Ha már csak egy dolgozó van életben, ő nyer és a játéknak vége | bemutatáskor | fontos | csapat | Move worker |  |
| 1.34 | Ha a játék végére több dolgozó marad életben, a legtöbb pontot szerző játékos(ok) nyer(nek). | bemutatáskor | alapvető | megrendelő | Move worker |  |
| 1.35 | Azonos pontszám esetén döntetlen van több játékos között. | bemutatáskor | fontos | csapat |  |  |
| 1.36 | A játék folyamatos, nincsenek körök | bemutatáskor | opcionális | csapat |  |  |
| 1.37 | A játéknak nincs időkorlátja | bemutatáskor | opcionális | csapat |  |  |
| 1.38 | A játékot el lehet indítani | bemutatáskor | alapvető | megrendelő | Start game |  |
| 1.39 | A játékból ki lehet lépni | bemutatáskor | alapvető | megrendelő | Exit game |  |

### Erőforrásokkal kapcsolatos követelmények

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| 2.01 | Számítógép (PC) | bemutatáskor | alapvető | megrendelő | Célhardver |
| 2.02 | Eclipse | nincs | opcionális | csapat | JAVA IDE |
| 2.03 | JRE | bemutatáskor | alapvető | megrendelő | JAVA futtatókörnyezet |
| 2.04 | Slack | nincs | opcionális | csapat | Kommunikációs platform |
| 2.05 | WhiteStarUML | nincs | opcionális | csapat | UML modellező |
| 2.06 | Google Drive | nincs | alapvető | csapat | Webes tárhely és dokumentumkezelő |
| 2.07 | Microsoft Office | nincs | opcionális | csapat | Dokumentumkezelő |
| 2.08 | Git | nincs | alapvető | csapat | Verziókezelő |
| 2.09 | Github | nincs | alapvető | csapat | Webes Git platform |
| 2.10 | GIMP | nincs | opcionális | csapat | Raszteres képszerkesztő |
| 2.11 | Inkscape | nincs | opcionális | csapat | Vektoros képszerkesztő |
| 2.12 | Egér | nincs | alapvető | megrendelő | Periféria |
| 2.13 | Billentyűzet | nincs | alapvető | megrendelő | Periféria |
| 2.14 | Monitor | nincs | alapvető | megrendelő | Periféria |

### Átadással kapcsolatos követelmények

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| 3.01 | Szkeleton átadása | bemutatáskor | alapvető | megrendelő | márc. 12. |
| 3.02 | Prototípus átadása | bemutatáskor | alapvető | megrendelő | ápr. 23. |
| 3.03 | Grafikus változat átadása | bemutatáskor | alapvető | megrendelő | máj. 14. |
| 3.04 | Teljes program átadása | bemutatáskor | alapvető | megrendelő | máj. 18. |
| 3.05 | Installálási útmutatást és kezelési leírás biztosítása | bemutatáskor | alapvető | megrendelő |  |

### Egyéb nem funkcionális követelmények

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Azonosító** | **Leírás** | **Ellenőrzés** | **Prioritás** | **Forrás** | **Komment** |
| 4.01 | A forrásprogramnak a laboratóriumban rendszeresített JDK alatt lefordíthatónak és futtathatónak kell lennie | bemutatáskor | alapvető | megrendelő |  |
| 4.02 | A programnak működnie kell a laboratórium számítógépein | bemutatáskor | alapvető | megrendelő |  |

## Lényeges use-case-ek

### Use-case leírások

|  |  |
| --- | --- |
| **Use-case neve** | View warehouse |
| **Rövid leírás** | A játékosok megtekintik a pályát. |
| **Aktorok** | Player |
| **Forgatókönyv** | 1. A játékosok megfigyelik a pályát. 2. A rendszer kirajzolja a pályát a ládákkal ill. a játékosokat azok kiindulási helyeivel együtt, továbbá kijelöli a ládák előírt helyeit. |

|  |  |
| --- | --- |
| **Use-case neve** | Move worker |
| **Rövid leírás** | A játékosok az általuk használt dolgozót irányítják. |
| **Aktorok** | Player |
| **Forgatókönyv** | 1. Egy játékos a saját dolgozójával lép egy vele szomszédos padlóra, ha az nem más dolgozó kiindulási helye. |
| **Alternatív forgatókönyv** | 1.A. A szomszédos mezőn láda tartózkodik, amely eltolható, ekkor a dolgozó eltolja azt a lépés irányába szomszédos mezőre. |
| **Alternatív forgatókönyv** | 1.B A szomszédos mező lyuk, a dolgozó életet veszít és visszakerül a kiindulási helyére vagy ha nics több élete, akkor a játékosa veszít. |

|  |  |
| --- | --- |
| **Use-case neve** | Push crates |
| **Rövid leírás** | A dolgozók ládákat tologatnak. |
| **Aktorok** | Player |
| **Forgatókönyv** | 1. Egy dolgozó eltol egy ládát, amely az eltolás irányába szomszédos mezőre kerül, ha az nem fal, oszlop vagy kiindulási mező. |
| **Alternatív forgatókönyv** | 1.A A láda egy lánc vége, ekkor az egész lánc eltolódik, ha a másik végén játékos vagy eltolható láda van. |
| **Alternatív forgatókönyv** | 1.B A szomszédos mező egy kijelölt hely, az eltolást végző játékos pontot kap. |
| **Alternatív forgatókönyv** | 1.B.1 A láda egy láncban szerepel, a pontot a lánc végén álló játékos kapja. |
| **Alternatív forgatókönyv** | 1.C A szomszédos mező egy lyuk, a láda leesik. |
| **Alternatív forgatókönyv** | 1.C.1 A leesett láda szívecskével volt megjelölve, az eltolást végző dolgozó életet kap. |
| **Alternatív forgatókönyv** | 1.D Az egyik kijelölt helyről eltolják a ládát, az érte pontot kapó játékos pontot veszít. |

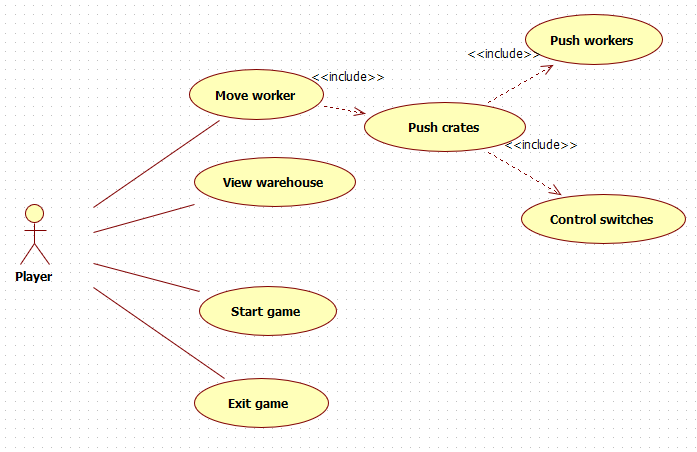
|  |  |
| --- | --- |
| **Use-case neve** | Push workers |
| **Rövid leírás** | A dolgozók közvetetten más dolgozókat tologatnak. |
| **Aktorok** | Player |
| **Forgatókönyv** | 1. Egy dolgozó eltol egy ládát, amely láncot alkot egy vagy több dolgozóval, ekkor a dolgozók is eltolódnak. |
| **Alternatív forgatókönyv** | 1.A Ha a lánc végén egy dolgozó áll és az eltolás irányában fal, oszlop, lyuk vagy más kiindulási hely található, a lánc eltolódik és az eltolt dolgozó életet veszít. |

|  |  |
| --- | --- |
| **Use-case neve** | Control switches |
| **Rövid leírás** | A dolgozók a kapcsolót irányítják. |
| **Aktorok** | Player |
| **Forgatókönyv** | 1. Egy dolgozó egy ládát tol egy kapcsolóra, ezzel aktiválva bizonyos lyukakat. 2. Egy dolgozó egy ládát eltol egy kapcsolóról, ezzel bizonyos lyukakat padlókká alakítva. |

|  |  |
| --- | --- |
| **Use-case neve** | Start game |
| **Rövid leírás** | A játékosok elindítják a játékot. |
| **Aktorok** | Player |
| **Forgatókönyv** | 1. A játékosok megadják a létszámukat és elindítják a játékot. |

|  |  |
| --- | --- |
| **Use-case neve** | Exit game |
| **Rövid leírás** | A játékosok bezárják a programot. |
| **Aktorok** | Player |
| **Forgatókönyv** | 1. Egy játékos bezárja a játékot a kilépés gombra kattintva. |

### Use-case diagram



## Szótár

**dolgozó:** Egy entitás, amely a játéktérben található és egy játékos által irányítható. A dolgozó pontokkal és életekkel rendelkezik, továbbá ládákat ill. egy láncban más dolgozókat is el tud tolni. Egy dolgozó egy mező méretű. A dolgozók és a játékosok kölcsönösen megfeleltethetők egymásnak.

**élet:** Egy szám, ami az egyes dolgozókhoz van rendelve. Ha ez eléri a nullát, az őt irányító játékos veszít és nem folytathatja a játékot. A játék kezdetén minden dolgozónak három élete van.

**előírt hely**: ld. láda (előírt) helye.

**eltol:** Egy lépés, aminek keretében egy dolgozó, vagy egy láda a szomszédos mezőre mozdít egy vele szomszédos mezőn tartózkodó dolgozót vagy ládát az eltolás irányába.

**eltolható láda:** Egy láda eltolható egy lánc végén, ha a tolás irányában szomszédos mező padló.

**fal:** A játéktér határait jelző mezőtípus, amelyen ládák nem tolhatók keresztül, és amelynek egy dolgozót ládával nekitolva az életet veszít.

**holtverseny:** Egy helyzet, ami akkor következik be, ha a játék végén több játékos pontszáma megegyezik. Holtverseny esetén ily módon több nyertese is lehet a játéknak.

**időkorlát:** Egy számláló, amelynek értéke az idő előrehaladásával folyamatosan csökken, és ha eléri a nullát, a játék véget ér.

**játék kezdete:** Egy esemény, ami akkor következik be, ha játéktér és a benne elhelyezkedő objektumok teljesen betöltöttek.

**játékos:** Egy valós személy, aki egy dolgozót irányít.

**játéktér:** A tér, amelyben a játékosok játszanak. A játéktér négyzet alakú mezőkből áll, amelynek különböző típusai vannak. Egy mező legfeljebb négy másik mezővel lehet szomszédos az oldalai mentén.

**játék vége:** Egy esemény, ami akkor következik be, ha az összes láda a helyére kerül, ha már egyetlen ládát sem lehet eltolni vagy egy kivétellel az összes dolgozó életeinek száma eléri a nullát.

**kapcsoló:** Olyan padlótípus, amelyre ládát rátolva az lyukká változtat egy játék által kijelölt padlót vagy padlókat. Ha a láda lekerül erről a mezőről, a lyukak visszállnak padlóvá. Ha munkás áll a kapcsolóra, akkor nem kapcsol.

**kiindulási mező:** Minden dolgozóhoz pontosan egy ilyen mező tartozik, ami a játéktér szélén található és három oldalról fallal határolt. Egy kiindulási mezőn csak az ahhoz tartozó dolgozó tartózkodhat, más dolgozó vagy láda nem. Egy dolgozó szempontjából másik dolgozó kiindulási mezője falnak vagy oszlopnak tekinthető, azaz őt arra rátolva a dolgozó életet veszít. A dolgozók ide kerülnek a játék elején, illetve életvesztést követően.

**láda:** Egy entitás, amely a játéktérben található, egy mező méretű és dolgozók által eltolható.

**láda beragad:** Egy láda beragadtnak tekinthető, ha azt már egyetlen irányba se lehet eltolni, mivel legalább két szomszédos irányban kiindulási mező**,** fal, oszlop vagy egy beragadt láda található.

**láda (előírt) helye:** A játék elején kijelölt mező, amire egy ládát rátolva a tolást végző játékos pontot kap.

**láda leesik:** Egy esemény, ami akkor következik be, ha egy láda egy lyukra kerül. Ennek keretében a láda eltűnik a játéktérből.

**lánc:** Ládák és/vagy dolgozók sorozata, amelyek egy bizonyos irányba szomszédos mezőkön helyezkednek el egymáshoz képest, közvetlen egymás után. Egy láncban szereplő ládát vagy játékost eltolva, az eltolja a vele szomszédos ládát vagy játékost, ugyanabban az irányban.

**lánc vége:** A lánc első, illetve utolsó eleme, lehet dolgozó vagy láda.

**lépés:** Egy esemény, ami alatt egy dolgozó eltol egy vagy több ládát és/vagy dolgozót egy mezővel.

**lyuk:** Egy padlótípus, amelyre, ha dolgozó lép, az életet veszít, ha pedig ládát tolnak rá, az leesik.

**mező:** A játékteret felépítő egység, több típusa van, melyek különböző tulajdonságokkal jellemezhetők.

**nyer:** Egy játékos nyer, ha a játék végén neki van a legtöbb pontja, vagy egyedül az ő dolgozója maradt életben.

**oszlop:** Egy mezőtípus, amely a játéktér belsejében található. Az oszlopon ládák nem tolhatók keresztül és egy dolgozót egy oszlopnak nekitolva, az életet veszít.

**padló:** Egy általános mezőtípus, amelyen dolgozó vagy láda tartózkodhat és ami nincs semmilyen hatással a rajta álló játékosra vagy ládára. Ez a mező alap típusa.

**pont:** Egy szám, ami az egyes játékosokhoz van rendelve. Egy játékos pontjainak száma eggyel nő, ha a játékos a helyére tol egy ládát az általa irányított dolgozóval. Ha egy helyén lévő ládát eltolnak, a ládát helyére toló játékos pontjainak száma eggyel csökken.

**pontlopás:** Egy esemény, ami akkor következik be, ha egy játékos a helyére tol egy vagy több ládát úgy, hogy közötte és a ládák között más dolgozók is voltak. Ekkor a tolást kezdeményező játékos kapja a pontokat.

**raktár:** ld. játéktér.

**raktárépület**: ld. játéktér

**Sokoban:** Egy népszerű játék, amelyen a projektben tárgyalt játék alapul.

**szívecske:** Egy szimbólum, amellyel ládák meg lehetnek jelölve a játék által. Az ilyen ládák lyukba tolása eggyel megnöveli az eltoló dolgozó életeinek a számát.

**szomszédos mező:** Egy mezőnek az oldalain túl lévő mezők. Egy mezőnek 2-4 szomszédja lehet, attól függően, hogy az a játéktér sarkában, szélén vagy belsejében tartózkodik.

**veszít:** Egy játékos veszít, ha dolgozójának életeinek száma eléri a nullát, vagy a játék végén nem neki van a legtöbb pontja. Ekkor az ő dolgozója eltűnik a raktárból.

## Projekt terv

### A végrehajtás lépései

A feladatot három lépcsőben valósítjuk meg. A folyamatos munkát részhatáridők is segítik, melyeket az alábbi táblázat foglal össze.

|  |  |  |
| --- | --- | --- |
| **Feladat** | **Leadás módja\*** | **Határidő** |
| Követelmény, projekt, funkcionálitás | beadás | febr. 19. |
| Analízis modell kidolgozása 1. | beadás | febr. 26. |
| Analízis modell kidolgozása 2. | beadás | márc. 5. |
| Szkeleton tervezése | beadás | márc. 12. |
| **Szkeleton** | beadás + feltöltés | márc. 19. |
| Prototípus koncepciója | beadás | márc. 26. |
| Részletes tervek | beadás | ápr. 9. |
| Prototípus készítése, tesztelése | - | ápr. 16. |
| **Prototípus** | beadás + feltöltés | ápr. 23. |
| Grafikus felület specifikációja | beadás | máj. 2. |
| Grafikus változat készítése | - | máj. 7. |
| **Grafikus változat** | beadás + feltöltés | máj. 14. |
| Összefoglalás | beadás + feltöltés | máj. 18. |

\*A leadási módok magyarázata:

* beadás: A dokumentum leadása nyomtatott formában a konzulens részére.
* feltöltés: A forráskód, illetve az esetleges tesztbemenetek és elvárt kimenetek feltöltése az erre kialakított Hercules feladatbeadó rendszerbe.

A *szkeleton* változat célja annak bizonyítása, hogy az objektum és dinamikus modellek a definiált feladat egy modelljét alkotják. A szkeleton egy program, amelyben már valamennyi, a végső rendszerben is szereplő business objektum szerepel. Az objektumoknak csak az interfésze definiált. Valamennyi metódus az indulás pillanatában az ernyőre szöveges változatban kiírja a saját nevét, majd meghívja azon metódusokat, amelyeket a szolgáltatás végrehajtása érdekében meg kell hívnia. Amennyiben a metódusból valamely feltétel fennállása esetén hívunk meg más metódusokat, akkor a feltételre vonatkozó kérdést interaktívan az ernyőn fel kell tenni és a kapott válasz alapján kell a továbbiakban eljárni. A szkeletonnak alkalmasnak kell lenni arra, hogy a különböző forgatókönyvek és szekvencia diagramok ellenőrizhetők legyenek. Csak karakteres ernyőkezelés fogadható el, mert ez biztosítja a rendszer egyszerűségét.

A *prototípus* program célja annak demonstrálása, hogy a program elkészült, helyesen működik, valamennyi feladatát teljesíti. A prototípus változat egy elkészült program kivéve a kifejlett grafikus interfészt. Ez a program is parancssorból futtatható és karakteres ernyőkezelést alkalmaz. A változat tervezési szempontból elkészült, az ütemezés, az aktív objektumok kezelése megoldott. A business objektumok - a megjelenítésre vonatkozó részeket kivéve - valamennyi metódusa a végleges algoritmusokat tartalmazza. A megjelenítés és működtetés egy alfanumerikus ernyőn vezérelhető és követhető, ugyanakkor a vezérlés fájlból is történhet és a megjelenítés fájlba is logolható, ezzel megteremtve a rendszer tesztelésének lehetőségét. Különös figyelmet kell fordítani a parancssori interfész logikájára, felépítésére, valamint arra, hogy az mennyiben tükrözi és teszi láthatóvá a program működését, a beavatkozások hatásait.

A teljes (*grafikus*) változat a prototípustól elvileg csak a kezelői felület minőségében különbözhet. Ennek változatnak az értékelésekor a hangsúlyt sokkal inkább a megvalósítás belső szerkezetére, semmint a külalakra kell helyezni.

### A csapat

A főbb feladatköröket igyekszünk úgy elosztani, hogy mindenki a számára legkézenfekvőbb témával tudjon foglalkozni. További fontos szempont a közel egységes terhelés, így szükség szerint egyes tagok az elsődleges feladatkörükön kívül eső témával is foglalkozhatnak.

|  |  |
| --- | --- |
| **Név** | **Elsődleges feladatkör** |
| Csanády Máté Bende | Dokumentáció, UML |
| Jani Balázs Gábor | Programozás, Monitoring |
| Lakatos Dániel (CSK) | Menedzsment, Dokumentáció, Programozás |
| Lenkefi Péter | Programozás, Grafika |
| Szakállas Gergely | Dokumentáció, UML, Programozás |

### 2.6.3 Kapcsolattartás

**Kommunikáció:** Az elsődleges kommunikációs csatornánkat a Slack nevű program jelenti. Itt lehetőség van különböző beszélgetéseket létrehozni az eltérő feladatokhoz, valamint csatlakoztatni a verzió- és dokumentumkezelő rendszerünket, ami lehetőséget ad egy átfogó kép kialakítására.

**Meeting:** Igyekszünk hetente legalább egyszer személyes meetinget tartani, ahol összefoglaljuk a történéseket, felvetjük és megvitatjuk az éppen aktuális problémákat. Ezen kívül részt veszünk a kijelölt konzultációkon, minél több csapattaggal.

**Dokumentumkezelés:** A dokumentumok kezelésére Google Docsot használunk. Ennek segítségével közösen szerkeszthetjük az állományt, követhetjük egymás változtatásait, valamint hozzászólhatunk, kiegészíthetjük egymás terveit, írásait.

**Verziókezelés:** Egy ilyen szintű projektnél már fontosnak tartjuk egy verziókezelő használatát, hogy mindenki átfogó képet kapjon a munka aktuális állapotáról, és a kódintegráció is zökkenőmentes legyen. A Git nevű szoftvert választottuk erre a feladatra, és hostként a Githubot használjuk.

### 2.6.4 Programok

**Dokumentáció:** A tervezési fázisban Google Docs dokumentumokat használunk, a dokumentációt később Microsoft Wordben véglegesítjük.

**Verziókezelés:** A Git nevű szoftvert választottuk erre a feladatra.

**Verziókezelő host**: A GitHub-on tároljuk a git repository-t.

**Fejlesztőkörnyezet:** A programozás az Eclipse szoftverben valósul meg.

**UML:** Az UML diagramokat a WhiteStarUML, valamint az Eclipse szoftverekkel készítjük.

**Grafika**: A grafikai elemek elkészítéséhez GIMP-et illetve Inkscape-et használunk.

## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2018.02.16. 16:00 | 0,5 óra | LENKEFI | Cél (2.1.1) és szakterület (2.1.2) specifikálása, felhasználók meghatározása (2.2.3) |
| 2018.02.16. 16:30 | 1 óra | LENKEFI | Funkciók meghatározása (2.2.2) |
| 2018.02.16. 18:00 | 1 óra | JANI | Funkciók kiegészítése (2.2.2) |
| 2018.02.16. 19:00 | 1,5 óra | SZAKÁLLAS | Szótár előzetes megírása a funkciók alapján (2.5) |
| 2018.02.16. 20:00 | 1 óra | LAKATOS | Projekt terv elkészítése (2.6) |
| 2018.02.17. 13:00 | 1 óra | LAKATOS | Funkciók átfogalmazása, pontosítása (2.2.2) |
| 2018.02.17. 14:00 | 0,5 óra | CSANÁDY | Bevezetés kiegészítése (2.1) |
| 2018.02.17. 15:00 | 1 óra | SZAKÁLLAS | Funkciók áttekintése, pontosítása (2.2.2), szótár frissítése (2.5) |
| 2018.02.17. 15:00 | 1 óra | LAKATOS | Funkcionális, átadással kapcsolatos és egyéb nem funkcionális követelmények meghatározása (2.3.1, 2.3.3, 2.3.4) |
| 2018.02.17. 18:30 | 2 óra | SZAKÁLLAS | Lényeges use-casek meghatározása (2.4) |
| 2018.02.17. 18:30 | 2,5 óra | CSANÁDY | Szótár frissítése (2.5), a dokumentum általános felülvizsgálata |
| 2018.02.17. 19:00 | 0,5 óra | LENKEFI | Funkciók és követelmények felülvizsgálata (2.2.2, 2.3) |
| 2018.02.17. 19:00 | 2 óra | JANI | A dokumentum általános felülvizsgálata, esztétikai és grammatikai hibák javítása |
| 2018.02.17. 21:00 | 1 óra | LENKEFI | Általános áttekintés hozzáadása (2.2.1) |
| 2018.02.17 23:00 | 0,5 óra | LAKATOS | Definíciók és rövidítések magyarázatának hozzáadása (2.1.3) |
| 2018.02.17 23:30 | 1 óra | LAKATOS | Dokumentum véglegesítése, esztétikai korrekciók |