

2. Követelmény, projekt, funkcionalitás

10 – itee_team

Konzulens:

Budai Péter

Csapattagok:

Elekes Tamás Csaba	E30C8Z	elekestamas22@gmail.com
Seres Márk Dániel	EUQ8V5	seres.dani@gmail.com
Rédey Bálint Attila	DAVRIZ	botvinnik09@gmail.com
Nagy András	VWBG06	nagyandrasgall@gmail.com
Fuksz Domonkos	GIT0NQ	fukszdomonkos@gmail.com

2014. március 1.

2. Követelmény, projekt, funkcionalitás

2.1. Bevezetés

2.1.1. Cél

A dokumentum célja az elkészülendő szoftver bemutatása fejlesztői szemszögből, ami a szoftver fejlesztői dokumentációja, amely segítségével megismerhető a szoftver felépítése és fejlesztésének lépései.

2.1.2. Szakterület

A kialakítandó szoftver egy játék, amely kizárólag szórakoztatási célokat szolgál. A fejlesztők célja egy olyan szoftver létrehozása, amely megfelel a feladatkiírásban foglalt követelményeknek.

2.1.3. Definíciók, rövidítések

HSZK Hallgatói Számítógép Központ

szoftver a program amit írunk

2.1.4. Hivatkozások

- http://en.wikipedia.org/wiki/Tower_defense
- [http://en.wikipedia.org/wiki/The_Lord_of_the_Rings_\(film_series\)](http://en.wikipedia.org/wiki/The_Lord_of_the_Rings_(film_series))

2.1.5. Összefoglalás

A dokumentum további részei az analízis modell, ami a szoftver főbb osztályainak, alkotóelemeinek tárgyalása, a szkeleton, amiben a belső működést tárgyaljuk.

A prototípus részben a részletes terveket, use-case-eket lehet majd megtalálni.

A szoftver grafikus tulajdonságait, interfészét, a végső fázisban dokumentáljuk.

2.2. Áttekintés

2.2.1. Általános áttekintés

A szoftver kialakítandó architektúrája objektum-orientált, modell-nézet-vezérlő mintát követ.

A modell fontosabb alrendszerei:

- pálya,
- tornyok,
- akadályok,
- ellenségek.

Az ellenségek, illetve a tornyok és akadályok közötti kapcsolat lényege, hogy a tornyok tudják, mikor ér a hatókörükbe egy ellenség. A tornyoknak tudniuk kell értesíteni az ellenségeket, amikor tüzet nyitnak, ami alapján az ellenségnek változik az életereje.

A pálya és az ellenségek közötti kapcsolat lényege, hogy az ellenségek ismerjék a pályán lévő utat, tudják azt követni.

2.2.2. Funkciók

Az eredeti feladatkiírás szövege

A gonosz emberek, tündék, törpök és hobbitok szövetséget kötnek, hogy elpusztítsák az Egy Gyűrűt a Végzet Hegyénél. Szerencsére csak Mordor földjén keresztül tudnak eljutni a hegyhez, így jóságos Szarumánnak lehetősége van védelmi tornyokat építeni, hogy segítsen megvédeni Szauron hatalmát. A játék célja annak megakadályozása, hogy a Gyűrű szövetségének tagjai közül bárki is eljusson a Végzet Hegyéhez. Egy ellenség akkor pusztul el, ha összességében megfelelő mértékű sebzést kap a tornyokból származó lövedékektől. A tornyok építéséhez Szarumánnak a varázserejét kell használnia. Szarumán akkor tud tornyot építeni, ha megfelelő mennyiségű varázsereje van hozzá. A varázsereje minden egyes elpusztított ember, tünde, törp vagy hobbit után bizonyos mértékben növekszik.

A Gyűrű szövetségének tagjai különböző utakon juthatnak el a Végzet Hegyéhez. Az utakról nem térhetnek le. Szarumán az utakra nem tud tornyot építeni, csak az utak mellé. Az utakra azonban tehet akadályokat, amik az akadály területén lassítják az ellenség haladását. A tornyoknak van egy adott hatótávolsága és tüzelési gyakorisága. Szarumán a varázserejét arra is használhatja, hogy a tornyokat és akadályokat különböző varázskövekkel ruházza fel. A varázsköveknek több fajtája is létezik, és különböző hatásúak lehetnek. Egyes kövek növelhetik a tornyok hatótávolságát vagy tüzelési gyakoriságát, más kövek egy-egy típusú ellenfél esetén megnövelik a lövedékek sebzési erejét.

A játék során az ellenségek folyamatosan jönnek. A játék elején ritkábban, később gyakrabban és nagyobb csoportokban, azonban számuk véges, előbb-utóbb elfogynak. A játék akkor ér véget, ha egy ellenség eljut a Végzet Hegyéhez, vagy ha már sikerült az összes ellenséget kiirtani. Az első esetben Szauron és Szarumán megsemmisül, utóbbi esetben fényes győzelmet aratnak, és örökké uralni fogják a világot.

Feladat specifikációja

A feladat egy grafikus felületű logikai játékszoftver készítése Java nyelven, ami megfelel a fenti leírás szövegének, továbbá működik a HSzK számítógépein.

A szoftver elindítása után a kezdőképernyő fogadja a felhasználót, ahol új játékot indíthat a megfelelő gombbal. Az új játék indítása esetén a felhasználónak ki kell választania egy pályát. Ha ez megtörtént, elindul a játék.

A pályák téglalap alakúak és különböző méretűek lehetnek. A pályák egy $n \times m$ méretű négyzetrácsból állnak, ahol m a sorok n pedig az oszlopok számát jelenti.

A pálya négyzetrácsos szerkezetének a celláira a játékos védelmi egységeket helyezhet varázserőért cserébe. A cellák két típusúak lehetnek: út, és háttér. A háttérre alkotó cellákra tornyokat lehet helyezni, az út celláira pedig akadályokat.

A tornyoknak három jellemzője van: tüzelési sebesség, sebzés és hatótáv. Amint egy ellenség egy torony hatókörébe ér, az tüzet nyit rá, azonban egy torony egyszerre csak egy célpontra tüzelhet. Lehetséges a tornyokat fejleszteni is: két hely van minden toronyban, amibe a torony további képességeit, azaz a hatótávolságot és a tüzelési gyorsaságot fokozó kristályokat helyezhetünk. Végül pedig minden torony esetén lehetőség van egy darab, az egyes fajok elleni sebzést növelő kristályt helyezni.

Az út celláira elhelyezhető akadályok lassítják a felettük elhaladó ellenségeket, viszont elhelyezésük után használnak minden ellenség lassításakor, míg végül meg nem szűnnek. Az akadályok jellemzői, hogy mennyi ellenség haladhat át rajtuk és a lassítás mértéke. Az akadályokat is lehetséges fejleszteni: lehet növelni a lassítás mértékét, illetve meg lehet javítani az akadályt.

A tornyok és akadályok elhelyezése úgy történik, hogy a játékos a kívánt cellára kattint, majd a felkínált lehetőségek közül kiválasztja az elhelyezendő tornyot, illetve akadályt.

A védelmi egységeket el is adhatja a játékos, ekkor az arra fordított varázserőnek, tehát a vásárlás és az esetleges fejlesztések árának a fele visszatérül. Az eladásához először rá kell kattintani az eladni kívánt védelmi egység képére, majd a megjelenő lehetőségek közül ki kell választani az eladást. Az akadályok esetén a visszajáró varázserő az elhasználtság mértékétől is függ: a ráfordított költségek felét, tehát ami egy torony

esetén visszajárna, még meg kell szorozni az ellenségek számának, amik még áthaladhatnak az akadályon, és az ellenségek számának, amik összesen áthaladhatnak az akadályon, hányadosával.

A tornyok és akadályok fejlesztéséhez a felhasználó kiválasztja a fejleszteni kívánt egységet a rajta való kattintással, és megjelennek az azzal kapcsolatos műveletek, amelyek között szerepel az egység különböző módú fejlesztése is. Minden fejlesztés varázserőbe kerül.

Egy ellenségnek két jellemzője van: a sebessége és az életereje. A játékban négy különböző ellenfél van: emberek, tündék, törpök és hobbitok. Minden fajnak különböző jellemzőik vannak: az emberek sebessége és életereje közepes, a törpök sebessége lassú, életerejük magas, a tündék sebessége gyors, életerejük magas, a hobbitok sebessége lassú, életerejük pedig alacsony. Egy ellenség megölése után növekedik egy adott mennyiséggel a játékos varázsereje. Ez a megölt ellenség életerejétől függ.

Az ellenségek a belépési pontnál, vagy pontoknál érkeznek a játék kezdetétől a végéig, és a Végzet hegye felé tartanak. Az útvonalon lehetnek elágazások is, amikhez érve az ellenségek véletlenszerűen haladnak tovább a Végzet Hegye felé. A hegyet Szarumán varázlata védi, ami azonnal végez az odaérkezőkkel, azonban minden megölt ellenség után gyengül, míg végül meg nem szűnik. A hegyet védő varázslat ereje pályánként változik. A varázslat megszűnése után az első bejutó ellenséggel a játék véget ér.

A játékos akkor nyer, ha megöl minden ellenséget, és egy sem jut be a hegybe.

2.2.3. Felhasználók

A felhasználókról (játékosokról) a továbbiakban, az általánosság megszorítása nélkül, néhány tulajdonságot feltételezünk. A szoftver (játék) interaktív mivolta miatt a felhasználóról feltételezhető, hogy képes elsajátítani a meglévő leírások, útmutatók, illetve a menük és fülek alapján a szoftver használatát. Korbéli megkötés nincs, azonban részéről elvártak bizonyos alapszintű ismeretek a számítógép, illetve az ahhoz kapcsolódó inputok (billentyűzet, egér) kezeléséről. Továbbá feltételezett részéről a szoftver rendeltetésszerű használata.

2.2.4. Korlátozások

Az elkészült szoftvernek bizonyos, nem funkcionális, korlátoknak eleget kell tennie.

Legfontosabb ezek közül, hogy tudnia kell futni a HSZK-s gépeken. Ebből kifolyólag a különböző hardveres és szoftveres megkötések szempontjából ezen gépek paraméterei mérvadóak. A szoftvert Java 1.6 nyelven kell implementálni. A szoftvernek parancssorból futtathatónak és hordozhatónak kell lennie. A grafikai megjelenést tekintve egy adott, a képernyőn lévő, elemről a felhasználó számára egyértelműen eldönthetőnek kell lennie, hogy mi a szerepe, célja a szoftverben.

2.2.5. Feltételezések, kapcsolatok

A dokumentáció olvasójáról feltételezzük, hogy ismeri az alap koncepcióját a torony védés játékoknak, illetve ismeri A Gyűrűk Ura trilógia világát.

2.3. Követelmények

2.3.1. Funkcionális követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Use-case	Komment
F2.3.1.1	A szoftver elindítása után megjelenő főmenüből lehessen új játékot indítani	Funkcionális tesztelés	alapvető	feladat-leírás	Új játék	

F2.3.1.2	Lehessen pályát választani az új játék kezdése után	Funkcionális tesztelés	fontos	csapat	Pálya kiválasztása	
F2.3.1.3	A pályán lehessen tornyokat és akadályokat elhelyezni	Funkcionális tesztelés	alapvető	feladatleírás	Torony/Akadály elhelyezése	
F2.3.1.4	A pályán elhelyezett tornyokat és akadályokat lehessen fejleszteni	Funkcionális tesztelés	alapvető	feladatleírás	Torony/Akadály fejlesztése	
F2.3.1.5	A pályán elhelyezett tornyokat és akadályokat el lehessen adni	Funkcionális tesztelés	fontos	csapat	Torony/Akadály eladása	
F2.3.1.6	A szoftver ne fagyjon le játék közben, működjön helyesen	Funkcionális tesztelés	alapvető	csapat		

2.3.2. Erőforrásokkal kapcsolatos követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
E2.3.2.1	Java 1.6 futtató környezet	Futtatás előtt feltelepítettség ellenőrzése	Fontos	HSZK gépkövetelmény	
E2.3.2.2	Operációs rendszer: Windows XP vagy újabb vagy Linux vagy Mac OS X 10.7.3 vagy újabb	Felhasználó ellenőrzi	Fontos	Java 1.6 követelmény	

2.3.3. Átadással kapcsolatos követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
A2.3.3	A szoftvert 3 fázisban adjuk át: szkeleton, prototípus és grafikus változat. Mindhárom esetben forráskódot adunk át, amihez könnyen érthető telepítési útmutatót csatolunk.	Budai Péter laborvezető ellenőrzi.	Fontos		A mellékelt útmutató segítségével egy általános számítógépes ismeretekkel rendelkező felhasználó telepíteni tudja a szoftvert. Szükség esetén a telepítést vállaljuk.

2.3.4. Egyéb nem funkcionális követelmények

Azonosító	Leírás	Ellenőrzés	Prioritás	Forrás	Komment
N2.3.4.1	Bárki számára könnyen kezelhető legyen a szoftver.		Fontos	csapat	A szoftverrel mellé tartozó README fájl utasításai szerint, JAVA JDK 1.6 alatt fordíthatónak és futtathatónak kell lennie.
N2.3.4.2	Bemenetekre egyből kell regálni a rendszernek.	Tesztelés a minimum gép-igénynek a környékén.	Fontos	általános	
N2.3.4.3	Billentyűzet vagy egér segítségével kezelhető legyen		Alapvető	általános	
N2.3.4.4	Hordozhatóság	A szoftvert teszteljük több operációs rendszeren.	Fontos	feladat-leírás	Windows, Linux, OS X
N2.3.4.5	Terhelhetőség	Olyan tesztesetekkel amik sok objektumot használnak.	opcionális	csapat	
N2.3.4.6	Megfelelő dokumentáltság, többek közt karbantarthatóság szempontjából	Projektvezető (Budai Péter)	Alapvető	Projektvezető	

2.4. Lényeges use-case-ek

2.4.1. Use-case leírások

Use-case neve	Új játék
Rövid leírás	Új játék kezdése
Aktorok	Játékos
Forgatókönyv	A játékos a főmenüben kiválasztja az „Új játék kezdése” gombot. Ekkor megjelenik a pálya kiválasztása.

Use-case neve	Pálya kiválasztása
Rövid leírás	Pálya kiválasztása pályák közül
Aktorok	Játékos
Forgatókönyv	A játékos rákattint a kiválasztandó pálya nevére. Ezután megjelenik a kiválasztott pálya és elkezdődik a játék.

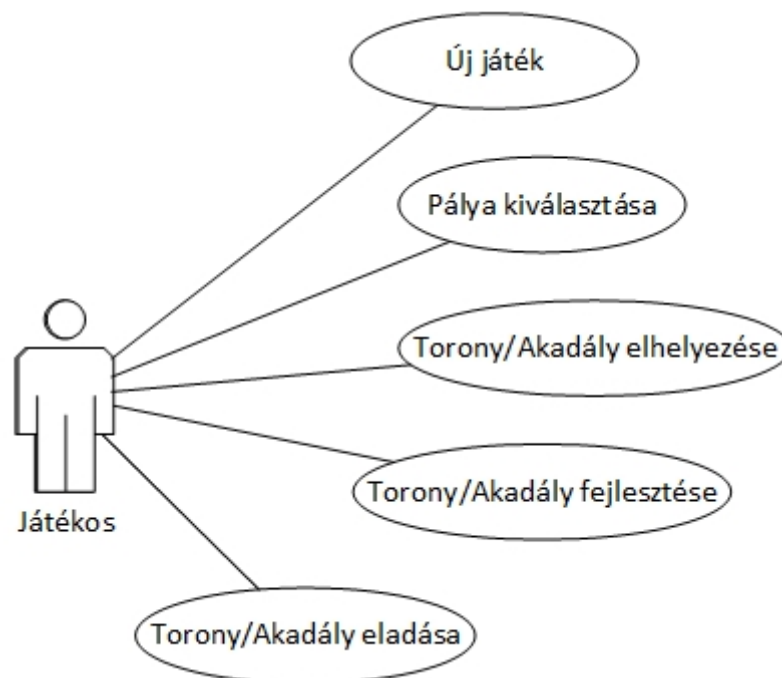
Use-case neve	Torony/Akadály elhelyezése
Rövid leírás	Torony vagy Akadály elhelyezése az arra kijelölt helyeken
Aktorok	Játékos

Forgatókönyv	A játékos az arra kijelölt helyre kattintva választhatja ki az elhelyezni kívánt tornyot vagy akadályt amennyiben van rá elég varázsereje.
--------------	--

Use-case neve	Torony/Akadály fejlesztése
Rövid leírás	Korábban elhelyezett torony/akadály hatékonyságának növelése
Aktorok	Játékos
Forgatókönyv	A játékos a már elhelyezett tornyot vagy akadályt fejlesztheti, hatékonyságát növelheti bizonyos mértékben, amennyiben van rá elég varázsereje.

Use-case neve	Torony/Akadály eladása
Rövid leírás	Korábban elhelyezett torony/akadály eladása
Aktorok	Játékos
Forgatókönyv	A játékos a már elhelyezett tornyot vagy akadályt eladhatja, és így visszakapja a ráfordított költségek egy részét.

2.4.2. Use-case diagram



2.5. Szótár

akadály (obstacle) az út celláira helyezett védelmi egység, a felette haladó ellenségek lassabban haladnak

belépési pont (entry point) az a hely, ahol az ellenségek belépnek a pályára

cella (cell) a pálya négyzetrácsos szerkezetét alkotó elemek

eladás (sell) egy védelmi egység megszüntetése, ami az arra költött varázsereő egy részének visszatérülésével jár

elhasználódás (attrition) a csapdák amortizációja, amit a rajtuk áthaladó ellenségek okoznak

elhelyezés (placing) egy védelmi egység a pálya egy bizonyos cellájára való telepítése varázserőért cserébe

ellenség (enemy) azok az egységek, amik a pálya belépési pontjainál jönnek be, és a Végzet Hegye felé haladnak. Fajtái: tünde, törp, hobbit, ember

ember(human) az ellenség egy fajtája, sebessége közepes, életerejé közepes

életerő (health) ellenségeket jellemző érték, amit a tornyokból kilőtt lövedékekkel lehet csökkenteni. Ha eléri a 0-t az ellenség meghal.

faj: ellenség típusok tünde, törp, hobbit, ember

fejlesztés (upgrade) egy védelmi egység hatékonyabbá tétele kristállyal

gépigény olyan számítógépes hardverek és nem hardveres erőforrások, amelyek minimálisan szükségesek a szoftver telepíthetőségéhez, rendes működéséhez

háttér (background) a pálya úton kívüli része, ide helyezhetjük a tornyokat

hobbit(hobbit) az ellenség egy fajtája, sebessége lassú, életerejé alacsony

játékos (player) a szoftver felhasználója

kristály (gem) tornyok/akadályok fejlesztésére alkalmas varázseszköz

kristály tartó(gem slot) a tornyok kristályokat tároló helyei

nyerni (win) az akkor fennálló helyzet, mikor elfogynak az ellenségek, és egy sem jutott be a Végzet Hegyébe

pálya (map) játéktér, legalább 1 út van rajta, minden más része háttér

readme olyan fájl, amely információkat tartalmaz a szoftver telepítéséről és használatba vételéről

sebez (damage) csökkenti az ellenség életerejét

szótár ez a dokumentum

torony (tower) a háttér celláira helyezett védelmi egység, automatikusan tüzel az ellenségre

törp(dwarf) az ellenség egy fajtája, sebessége lassú, életerejé magas

tünde(elf) az ellenség egy fajtája, sebessége gyors, életerejé magas

tüzel (fire) lövedéket kibocsát, mellyel ellenséget sebez meg

újratöltési idő(reload time) a torony két tüzelése között eltelt idő

út (way, road) a pálya azon része, ahol az ellenséges egységek haladnak, és az akadályokat helyezhetjük

varázserő (witchcraft) a játékban használt fizetőeszköz, cserébe lehet a védelmi egységeket elhelyezni, fejleszteni

védelmi egység torony vagy akadály

Végzet Hegye az ellenségek célpontja, az ide való eljutásuk megakadályozása a játékos célja

veszíteni (lose) az akkor fennálló helyzet, mikor egy ellenség bejut a Végzet Hegyébe

2.6. Projekt terv

2.6.1. Határidők

	Dátum	Feladat
1	febr. 14.	14 h - csapatok regisztrációja
2	febr. 24.	Követelmény, projekt, funkcionalitás - beadás
3	márc. 3.	Analízis modell kidolgozása 1. - beadás
4	márc. 10.	Analízis modell kidolgozása 2. - beadás
5	márc. 17.	Szkeleton tervezése - beadás
6	márc. 24.	Szkeleton - beadás
7	márc. 31.	Prototípus koncepciója - beadás
8	ápr. 7.	Részletes tervek - beadás
9	ápr. 22.	Prototípus - beadás
10	ápr. 28.	Grafikus felület specifikációja - beadás
11	máj. 12.	Grafikus változat - beadás
12	máj. 16.	Összefoglalás - beadás

2.6.2. Átadás

A szoftvert három fő fázisban adjuk át:

1. Skeleton. A szoftver váza, az átadott részletes specifikáció, és analízis modell alapján. Ez a verzió csak a modell működését szemlélteti, csak karakteres ki- és bemenetekkel.

Az objektumoknak csak az interfésze definiált.

Az egyes metódusok, kiírják a képernyőre nevüket és meghívják azon metódusokat, amelyeket a szolgáltatás végrehajtása érdekében meg kell hívniuk. Elágazás esetén feltesz egy kérdést, aminek a válasza alapján fut tovább a modell.

2. Prototípus. Elkészült szoftver minden tulajdonságával bír a prototípus, csak a grafikus megjelenítés hiányzik róla. A pillanatnyi állapotot alfanumerikus képernyőn lehet követni, illetve log fájlokba menti a futásának részleteit.

3. Grafikus változat. A prototípustól csak a megjelenésben különbözik, illetve annak belső megvalósításával bővített változata.

2.6.3. Szerepkörök

- Csapatvezető: Elekes Tamás
 - Értekezlet előkészületei, lebonyolítása
 - Project felügyelet
 - Kód tervezés
 - Implementálás
 - Audio programozás
- Fuksz Domonkos
 - Dokumentálás
 - Dokumentumok formai minősége
 - Tesztelés
 - Kódolás

- Nagy András
 - Értekezletek logolása
 - Implementálás
 - Tesztelés
- Seres Márk Dániel
 - Feladat beadás
 - Implementálás
 - Tesztelés
 - UML diagram készítés
- Rédey Bálint
 - Dokumentálás
 - Implementálás
 - Grafikus programozás
 - UML diagram készítés
 - Feladatok feltöltése

Ezen felül minden csapattagnak a konzultációkat látogatnia kell, illetve heti 1 teljes létszámú értekezleten részt kell vennie.

2.6.4. Felhasznált fejlesztőeszközök

Verziókezelés A csoportmunka szempontjából nagyon fontos, hogy mindenki naprakészen hozzáférjen a kód és a dokumentumok aktuális verzióihoz. Ehhez a Git verziókezelő rendszert választottuk, ami felhő alapú, könnyen kezelhető rendszer.

Java JDK A szoftvert Java nyelven kerül implementálásra, amihez az ingyenes Eclipse szoftvert használjuk. Az Eclipse nagyon jó környezetet ad Java JUnit tesztelés végzésére is.

UML Fontos a jó minőségű dokumentáció elkészítéséhez az ábrák megfelelő és szép elkészítése. Erre a célra több UML rajzoló szoftvert is használunk: StarUML és Microsoft Office Visio 2013.

Kommunikáció A csapatmunka során elengedhetetlen a megfelelő kommunikáció a tagok között. Egyszerű szöveges kommunikációra Facebook csoportot használunk, míg fontosabb dolgok tárgyalásakor a Skype konferenciabeszélgetést használjuk.

2.7. Napló

Kezdet	Időtartam	Résztvevők	Leírás
2014.02.09. 16:00	1 óra	Seres Fuksz Elekes Rédey Nagy	Alakuló értekezlet. Ismerkedés, adatok egyeztetése.
2014.02.19. 08:15	1 óra	Seres Fuksz Elekes Rédey Nagy	Konzultáció

Kezdet	Időtartam	Résztevők	Leírás
2014.02.19. 09:20	2,5 óra	Seres Fuksz Elekes Rédey Nagy	Értekezlet. Döntés: Szerepkörök tisztázása. Fuksz dokumentálja a 2.1.1, 2.1.2, 2.3.2, 2.4.1 pontokat. Seres dokumentálja a 2.2.1, 2.2.2, 2.4.1, 2.3.1 pontokat. Rédey dokumentálja a 2.2.3, 2.2.4, pontokat. Nagy dokumentálja a 2.3.3 pontot. Elekes dokumentálja a 2.3.4, 2.6 pontokat. 2.1.3-5, 2.2.5, 2.5 pontokat az előző pontokkal végezve dokumentáljuk.
2014.02.19. 13:00	1 óra	Elekes	2.3.4, 2.6 pontok dokumentálása
2014.02.19. 14:15	45 perc	Rédey	2.2.3, 2.2.4 pontok dokumentálása
2014.02.19. 18:00	2 óra	Nagy	A dokumentum miért readonly? hibakeresés
2014.02.21. 16:00	1 óra	Seres	2.2.1, 2.2.2 pont dokumentálása
2014.02.21. 16:00	20 perc	Nagy	A 2.3.3 dokumentálása
2014.02.22. 18:30	15 perc	Fuksz	2.4.1 pont kitöltése
2014.02.22. 19:30	10 perc	Seres	2.3.1 pont kitöltése
2014.02.23. 15:45	1 óra	Fuksz	2.1.1, 2.1.2, 2.3.2, 2.4.2 pont kitöltése
2014.02.23. 17:00	40 perc	Elekes	2.1.5, 2.2.5 pont, szótár
2014.02.23. 19:00	10 perc	Nagy	2.1.3, 2.1.4 pont kitöltése