

Programozási technológia 2.

Felhasználói dokumentáció

Fekete Fanni

DYS9AA

Könyvtár

Tartalomjegyzék

Fejlesztési napló	3
1. Bevezetés.....	4
1.1 A program célja, alapvető feladatai	4
2. Általános leírás	4
2.1 Termék működése	4
3. Általános követelmények	4
4. Felületi követelmények	5
4.1 Felhasználói felület	5
4.2 Hardware felület.....	5
4.3 Szoftver felület.....	5
4.4 Kommunikációs felület	5
5. Működési követelmények	5
5.1 Bemenő adatok.....	5
5.2 Feldolgozás	5
5.3 Kimenő adatok.....	5
5.4 Hibakezelés	6
6. Use-case diagram	6
7. User-story diagram.....	6
8. Osztály diagram	8
9. Osztályok leírása.....	8
10. Egyed-kapcsolati diagram	9
11. Tervezési megkötések	9
11.1 Nem működési követelmény.....	10
12. Adatbázis követelménye	10
12.1 Egyéb követelmények.....	10

Fejlesztési napló

Dátum	Leírás	Megjegyzés
2016. 03. 01.	Követelmény elemzés (use-case diagramok és user-storyk)	
2016. 03. 16.	Statikus terv (osztály diagram)	
2016. 03. 31.	Részleges prototípus tesztelés, egyed-kapcsolati diagram	
2016. 04. 04.	Teljes megvalósítás, teszteléssel	

1. Bevezetés

Ez a dokumentum egy követelmény specifikációt tartalmaz egy könyvtár nyilvántartásának szoftveréről. A következő oldalakon pontos megfogalmazás lesz arra, hogy a szoftver hogyan és milyen funkciókkal működjön.

1.1 A program célja, alapvető feladatai

Az elkészítendő program segítségével egy könyvtár kölcsönzési adatait lehet nyilván tartani. A kölcsönző személyek adatait lehet megadni, valamint a könyvek adatait felvinni. A könyvek állapotát ki lehet listázni, illetve statisztikákat mutat róluk. A tagokhoz a rendszer automatikusan késedelmi díjat számol, és kezeli a kölcsönzések folyamatát.

2. Általános leírás

Könyvtár nyilvántartása. A program tartalmazni fogja az összes alapvető funkciót, ami egy könyvtár fenntartásához szükséges. Egyszerűen kezelhető lesz a program, az adatok tárolását egy MySQL adatbázis fogja tartalmazni.

2.1 Termék működése

- Új tag felvitele
- Új kölcsönzés
- Könyvek zárolása
- Könyvek, illetve tagok kilistázása

3. Általános követelmények

- A programban megtekinthetők a tagok adatai (név, cím, könyvtárjegy szám), valamint a könyvek adatai (szerző, cím, ISBN szám, kiadás éve, összes példányszám, szabad példányszám).
- A könyvek zárolhatóak (és a zárolás visszaállítható) ilyenkor a kint lévő példányokat csak visszahozni lehet, új kölcsönzés nem vehető fel.
- A könyveknél jelenítsük meg, összesen hányszor kölcsönözték ki.
- Lehetőségünk van új tag felvitelére (név és cím megadásával). A könyvtárjegy száma automatikusan generálódik. A tag módosítható, illetve törölhető is, amennyiben nincs kint lévő kölcsönzése (a könyvtárjegy száma nem módosítható).
- Tag, illetve könyv kijelölésével vihetünk fel új kölcsönzést. Csak akkor lehet kölcsönözni, ha még van szabad példány és nem zárolt, egyszerre egy tag csak négy könyvet kölcsönözhet, minden könyvből csak egy példányt és csak akkor, ha nincs aktív kölcsönzése. A kölcsönzés dátumát automatikusan rögzítjük.
- A tagot kiválasztva listázódnak kölcsönzései dátum szerint.
- A kikölcsönzött könyveket lehessen visszahozni. A visszahozatal dátumát automatikusan rögzítjük.
- 30 napon belüli 5 késés után a tag egy hetes büntetést kap, amely idő alatt nem kölcsönözhet könyvet illetve további egy hétig maximálisan 2 könyvet kölcsönözhet, a büntetések lejáratát egy háttérszál ellenőrizze periodikusan.

4. Felületi követelmények

4.1 Felhasználói felület

- A kész program végső változata billentyűzet és egér felhasználásával lesz irányítható, grafikus felhasználói felülettel fog rendelkezni, ami Java Swing segítségével lesz megvalósítva.

4.2 Hardware felület

- A futáshoz szükséges, hogy a felhasználó számítógépére telepítve legyen a Java Runtime Environment (JRE), a program hardverigénye megegyezik a Sun által meghatározott minimum-konfigurációval: Pentium 166 MHz vagy annál gyorsabb processzor, minimum 32 MB RAM és 125 MB hely a háttértárolón.

4.3 Szoftver felület

- A java platformfüggetlensége miatt bármilyen operációs rendszeren jó a program, ami támogatja a Java-t.

4.4 Kommunikációs felület

- Java Database Connectivity
- Java Swing

5. Működési követelmények

- Könyvek zárolása
- Könyvek kikölcsönzésének számának megjelenítése
- Új tag felvétele
- Új kölcsönzés
- Tag kiválasztásánál kölcsönzései kilistázása dátum szerint
- Visszahozatal dátumának rögzítése
- Büntetés

5.1 Bemenő adatok

- Új tag
- Új kölcsönzés
- Új könyv

5.2 Feldolgozás

- Adatbázisból keresés
- Adatbázisba beillesztés
- Adatbázisból törlés
- Folyamatos adatbázis ellenőrzés

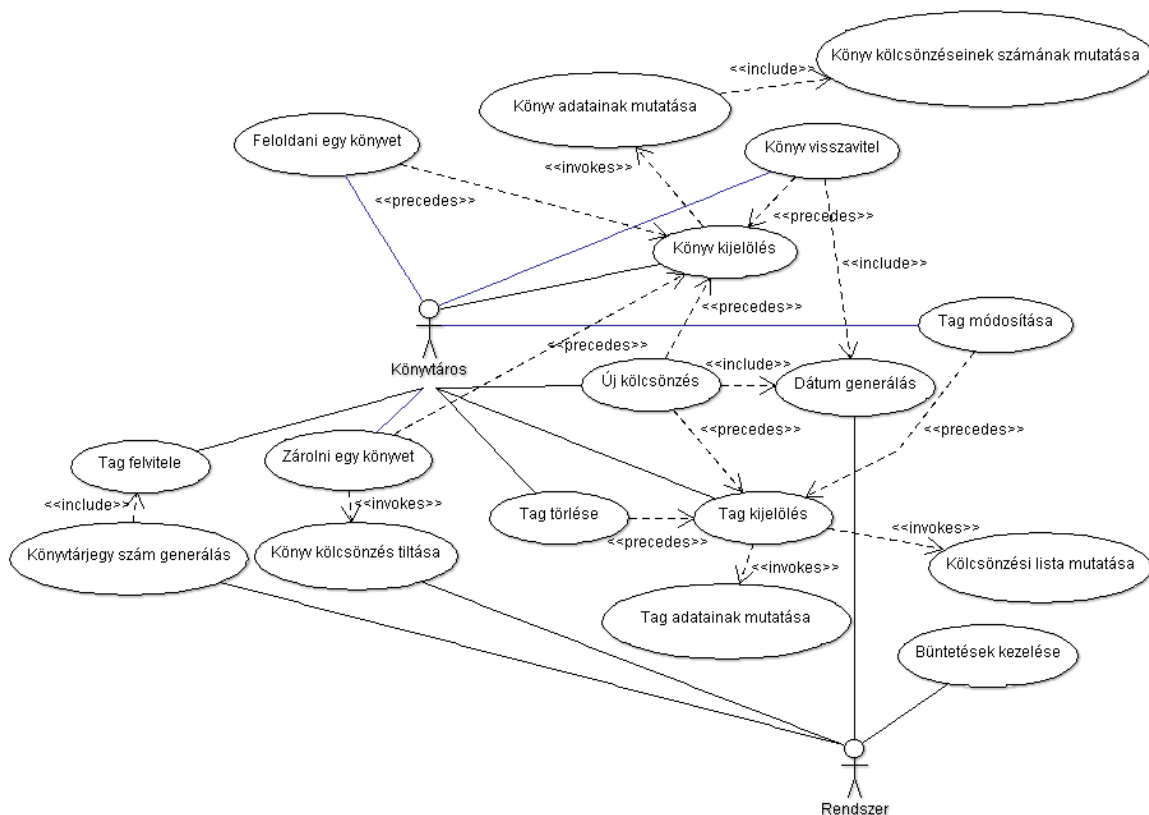
5.3 Kimenő adatok

- Adott könyv kikölcsönzésének száma, aktuális kikölcsönzöttsége
- Adott könyv adatai
- Tagok kölcsönzéseinek kilistázása
- Tagok adatai

5.4 Hibakezelés

- A program ellenőrzi a megadható bemeneteket és jelzi, ha egy adat hibás.
- Az adatbázis kapcsolatot a program ellenőrzi, hogy megszakadt e.

6. Use-case



7. User-story

AS A		Könyvtáros
I WANT TO		Tagok megtekintése
1	GIVEN	Tag kiválasztása, valamelyik adata alapján
	WHEN	Megtalálja a tagot az adatbázisban
	THEN	A program visszaadja a tag adatait
2	GIVEN	Tag kiválasztása, valamelyik adata alapján
	WHEN	Nem találja meg a tagot az adatbázisban
	THEN	A program hibaüzenettel tér vissza

AS A		Könyvtáros
I WANT TO		Könyv megtekintése
1	GIVEN	Könyv kiválasztása, valamelyik adata alapján
	WHEN	Megtalálja a könyvet az adatbázisban
	THEN	A program visszaadja a könyv adatait
2	GIVEN	Könyv kiválasztása, valamelyik adata alapján
	WHEN	Nem találja meg a könyvet az adatbázisban
	THEN	A program hibaüzenettel tér vissza

AS A		Könyvtáros
I WANT TO		Zárolni egy könyvet
1	GIVEN	Megtekintett könyv zárolásának megnyomása
	WHEN	Van bent példány
	THEN	Kölcsönzés nem vehető fel beállítása
2	GIVEN	Megtekintett könyv zárolásának megnyomása
	WHEN	Nincs bent példány
	THEN	A program hibaüzenettel tér vissza

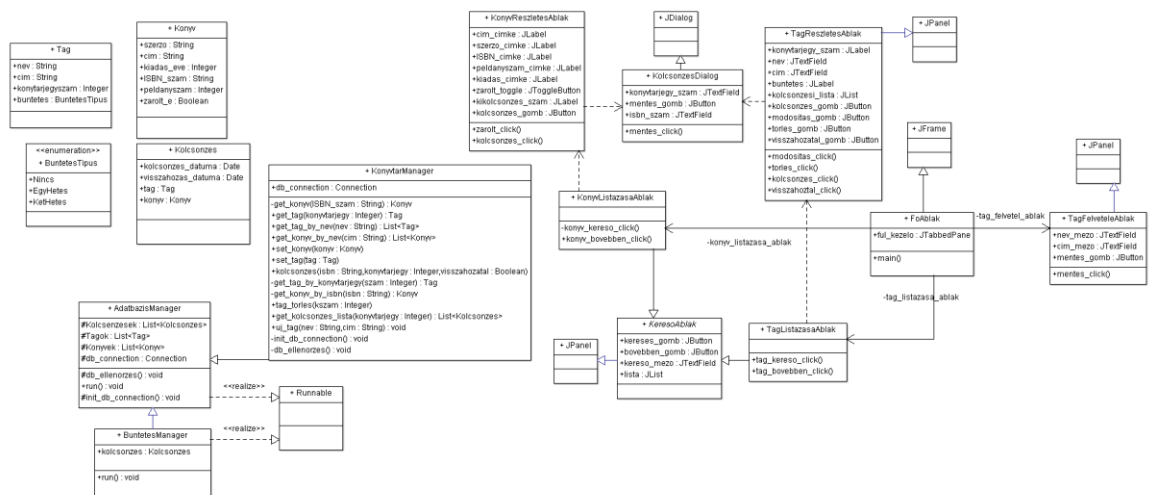
AS A		Könyvtáros
I WANT TO		Tag felvitele
1	GIVEN	Új tag opciót választottuk
	WHEN	A beadott adatok helyes formátumban vannak
	THEN	Hozzáadjuk a tagot
2	GIVEN	Új tag opciót választottuk
	WHEN	A beadott adatok formátuma hibás
	THEN	A program hibaüzenettel tér vissza

AS A		Könyvtáros
I WANT TO		Tag törlése
1	GIVEN	A kiválasztott tag törlését választottuk
	WHEN	Nincs kikölcsönzött könyve
	THEN	Töröljük a rekordot az adatbázisból
2	GIVEN	A kiválasztott tag törlését választottuk
	WHEN	Van kikölcsönzött könyve
	THEN	A program hibaüzenettel tér vissza

AS A		Könyvtáros
I WANT TO		Tag módosítása
1	GIVEN	A kiválasztott tag adatai módosítását választottuk
	WHEN	Ha van módosított adat a mezőkben
	THEN	Felvisszük a módosított adatokat
2	GIVEN	A kiválasztott tag adatai módosítását választottuk
	WHEN	Ha nincs módosított adat a mezőkben
	THEN	A program nem csinál semmit

AS A		Könyvtáros
I WANT TO		Új kölcsönzés
1	GIVEN	Tag vagy könyv kijelölésével új kölcsönzés választása
	WHEN	Ha van szabad példány, nem zárolt a könyv, nincs a tagnak 4nél több könyve kivéve, nincs kivéve ez a könyv már
	THEN	Felvisszük a kölcsönzés adatait, és rögzítjük az aktuális dátumot
2	GIVEN	Tag vagy könyv kijelölésével új kölcsönzés választása
	WHEN	Ha nincs szabad példány vagy zárolt a könyv vagy a tagnak 4nél több könyve van kivéve vagy ki van véve ez a könyv nála
	THEN	A program hibaüzenettel tér vissza

8. Osztály diagram

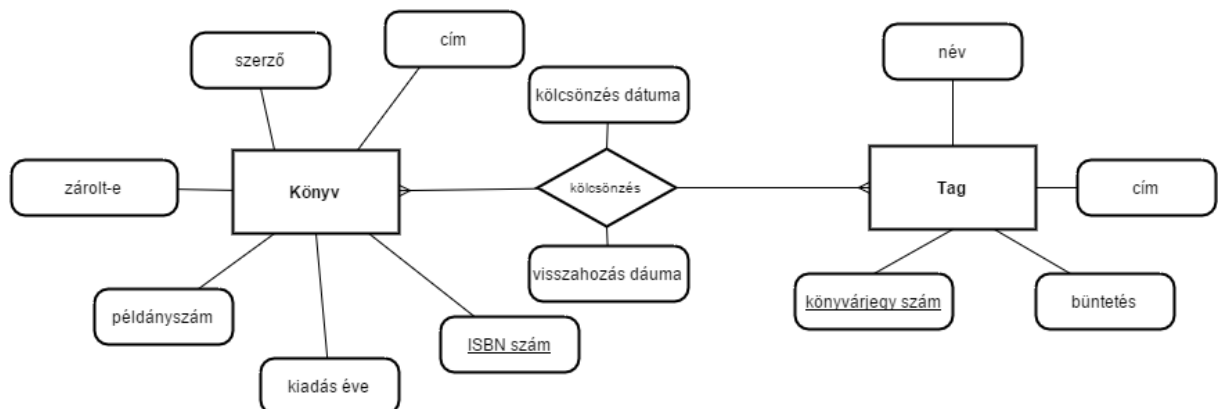


9. Osztályok leírása

- Tag: A tag tulajdonságait tartalmazó osztály.
- Konyv: A könyv tulajdonságait tartalmazó osztály.
- Kolcsonzes: Visszaadja, hogy melyik tag milyen könyvet kölcsönözött ki, illetve a kölcsönzés és a visszahozás dátuma is megjelenik.

- <<enumeration>> BuntetesTipus: A büntetés típusait tartalmazó osztály. Megkülönböztetjük az első heti, a második heti büntetést, és amikor nincs.
- <<static>> KonyvtarManager: Állandó kapcsolatot biztosít az adatbázissal, hogy a nyilvántartás információi mindig frissek maradjanak.
- AdatBazisManager: A tagok, könyvek és kölcsönzések adatbázisát tartalmazza, illetve állandó kapcsolatban áll a program többi részével.
- BuntetesManager: Meghatározott időközönként frissíti az adatbázist és azokat akik késnek a könyv visszahozásával a megfelelő mértékben bünteti.
- FoAblak: Főprogram, elindításakor 3 fül közül lehet választani: új tag felvétele, tagok listázása és könyvek listázása.
- TagFelveteleAblak: Meg kell adni az új tagnak a nevét és címét, a könyvtárjegyszámát a program automatikusan generálja.
- TagListazasaAblak: Kilistázza a tagokat, illetve bővebb információval szolgál az adott illetőről.
- KonyvListazasaAblak: Kilistázza a könyveket, illetve bővebb információval szolgál az adott könyvről.
- KeresoAblak: Absztrakt osztály.
- TagReszletesAblak: Miután a listából kiválasztottunk egy tagot minden információt megmutat róla. Kilistázza a kölcsönzéseit valamint új kölcsönzést hozzáadni, illetve az adott tagot lehet módosítani (csak név, cím), illetve törölni.
- KonyvReszletesAblak: Miután a listából kiválasztottuk a könyvet lehet új kölcsönzést hozzáadni vagy zárolni egy könyvet. Ilyenkor csak visszahozni lehet kikölcsönözni a zárolt példányt nem.
- KolcsonzesDialog: Új kölcsönzés esetén meg kell adni az egyedi könyvtárjegyszámot, illetve a könyv ISBN-számát.

10. Egyed-kapcsolati diagram



11. Tervezési megkötések

- A programot háromrétegű (model/view/controller) architektúrában kell felépíteni, amelyben a megjelenítés rétege elkülönül az adatkezelő modelltől és az adatokon végezhető műveleteket tartalmazó vezérlő rétegtől.

- A programhoz létre kell hozni a megfelelő adatbázist, amelyet a leírtak mentén kell megvalósítani. Az adatbázisban kellő számú mintaadatnak kell lennie a megfelelő teszteléshez (táblánként minimum 20 sor).
- Az adatok bevitelénél törekedni kell a felhasználóbarát, hibamentes megoldásokra. A program nem fogadhat el hibás bemenetet. Ahol lehetséges, biztosítsuk a kiválasztási lehetőséget, adatbevitelnél ellenőrizni kell az adatok helyességét (pl. az időpont ne lehessen múltbéli érték, a mennyiség ne lehessen negatív szám).
- Az implementációt Unit tesztekkel is tesztelni kell.
- A dokumentációnak tartalmaznia kell a feladat elemzését, felhasználói eseteit (UML felhasználói esetek diagrammal), a program szerkezetének leírását (UML osztálydiagrammal), valamint az adatbázis felépítésének leírását (egyedkapcsolati diagrammal).
- Az implementáció során törekedni kell a jól olvasható, átlátható, tiszta kód írására. (Clean Code: A Handbook of Agile Software Craftsmanship by Robert C. Martin; Chapter 1-10).

11.1 Nem működési követelmény

- Készítsünk egy egyszerű kliens Cache komponenst, amely ideiglenesen tárolóként szolgál az adatbázis és az implementált alkalmazás között, a betöltött adatokat tárolja és egy háttérszálon keresztül megadott időközönként letölti az adatbázisból az utolsó menet óta létrehozott illetve módosított rekordokat.
- Ezen változtatásokat jelenítsük meg a megfelelő helyen. (Feltételezhetjük, hogy egy adat szerkesztése közben soha nem módosul az adatbázisban.)
- A felhasználói felületen helyezzünk el egy gombot, amely a frissítést azonnal kiváltja.

12. Adatbázis követelménye

Az adatbázis az alábbi adatokat tárolja:

- tagok (név, cím, könyvtárjegy szám);
- könyvek (szerző, cím, ISBN szám, kiadás éve, példányszám);
- kölcsönzések (tag, könyv, kölcsönzés dátuma, visszahozás dátuma).

12.1 Egyéb követelmények

- NULL érték nem lehetséges.
- Azokat a számításokat és kereséseket végezzük a memóriában, amiket az adatbázis nem bír elvégezni.