

MEGOLDÁS

1. USE CASE DIAGRAM

Első lépésként keressük meg, milyen szereplőink lesznek a rendszerben.

Lesz egy *látogató*, illetve, mivel van bejelentkezési lehetőség is, így biztosan lesz egy *bejelentkezett felhasználó* is. Ezen kívül vegyük fel az *adatbázist*, mint szereplőt.

Ezután gyűjtjük ki a funkciókat a rendszer leírásából. Biztosan kell, mivel szerepel a szövegben:

- bejelentkezés
- „Emlékezz rám” funkció
- regisztráció
- mappa megnyitás (vagy mappa tartalom listázás)
- jegyzet módosítás
- jegyzet törlés
- új jegyzet felvétele
- új mappa felvétele
- új címke felvétele
- mappák listázása
- kijelentkezés
- részletes keresés
- egyszerű keresés

A fenti funkciókon kívül még a szövegből kikövetkeztethető, hogy lehetőség van a rendszer funkcióinak elolvasására (mivel a főoldalon bejelentkezés előtt ez látszik). Valamint, mivel *adatbázis* is szerepel a diagramon, így egy használati eset az „*adatok kezelése*”.

Miután fölvtünk minden fenti funkciónak egy-egy használati esetet, már csak a kapcsolatokat kell megállapítani közöttük.

A *látogató* kapcsolatban áll a *funkciók olvasása*, *bejelentkezés*, *regisztráció* használati esetekkel.

A bejelentkezett felhasználó kapcsolatban áll (használja) a:

- mappák listázása
- kijelentkezés

használati esetekkel. A három új elem létrehozását egyszerűbb kezelni, ha egy használati eset speciális eseteiként tekintünk rájuk. Így hozzunk létre egy „*Új elem létrehozása*” használati esetet, és kapcsoljuk specializáló kapcsolattal hozzá az *új jegyzet*, *új mappa* és *új címke* eseteket.

A keresésnél két megközelítést is használhatunk. Egyrészt, kezelhetjük itt is speciális esetként, ekkor fel kell venni egy *keresés* használati esetet, amelynek a leszármazottjai lesznek az *egyszerű* és a *részletes* keresés. De felfoghatjuk úgy is az oldalt, hogy alapértelmezetten *egyszerű* keresést használjuk, és onnan lehetőségünk van további mezőket megadni a *részletes* kereséshez. Ekkor a *részletes keresés* az *egyszerű* kereséssel <<extend>> kapcsolatban áll.

Az *emlékezz rám* funkció a bejelentkezést terjeszti ki, tehát vele áll <<extend>> kapcsolatban.

Mivel mappa tartalmát csak úgy tudjuk megnézni, ha előtte kilistázzuk az elérhető mappákat, így itt is egy kiterjesztő kapcsolat lesz. Hasonlóan a „címkékre szűrés” és a „jegyzet módosítás” esetek. A jegyzet módosítás esetet ugyan olyan céllal vettük fel, mint az „új elem felvételét”.

Végül minden olyan használati esetet, amelyik elment valamit az adatbázisba vagy kiolvas valamit az adatbázisból, kössünk össze az „adatbázis kezelés” használati eset <<include>> kapcsolattal.

2. CLASS DIAGRAM

A három egyed osztály (*Note*, *User*, *Notebook*) felvétele magától értetődik. A példán felvettem minden *getter-setter* függvényt, de ZH-n elég lesz 1-1 belőlük.

A *NoteDAO insertNote* függvénye a címet és a szöveget várja paraméterben, ezért két *String*et kap. Mivel ez csak egy adatbázis beszúrást végez, ezért nem tér vissza semmilyen értékkel.

A *getAllNote* függvény – ahogy a szövegben is szerepel – egy felhasználót vár paraméterül, és egy *Note* típusú listát ad vissza.

A *fillNoteBook* egy *NoteBook* objektumot kap paraméterben, aminek feltölti a listáját. Mivel „helyben” végzi a műveletet, ezért nincs szükség visszatérési értékre.

A *NoteControl* osztály egy *User* és egy *Note* adattagot tartalmaz. A *createNewNote* függvény nem vár paramétert, mivel üres jegyzetet ad vissza. Visszatérési értéke az új jegyzet, tehát egy *Note* típusú objektum.

A *deleteNote()* az adattagban tárolt jegyzetet törli, tehát nem vár paramétert és nem tér vissza semmivel.

Az *addTagToNote* egy szöveg típusú paramétert vár, ez lesz a jegyzet címkéje. Visszatérési érték itt sincs.

Végül már csak a kapcsolatok maradtak. Adattagként a *Note* objektum szerepel a *NoteControl* és a *Notebook* osztálynál, a *User* pedig szintén a *NoteControl*nál, így ott aggregáció típusú kapcsolatot kell jelölni (nem erős tartalmazás). A *NoteDAO* mindhárom egyed osztállyal kapcsolatban áll.

A *NoteDAO* jelen állapotában nincs kapcsolatban a *NoteControl* osztállyal, mert a jegyzeten végrehajtott műveletekkel kapcsolatban nem volt szó arról, hogy a módosítások adatbázisban is tárolódnak. De ez is egy elfogadható megoldás (indoklással), ekkor a *NoteDAO* is kapcsolatban áll a *NoteControl*lal.

3. TESZTELÉS

Teszteljárások

ID: TP_0001

TestCases: TC_0001, TC_0002, TC_0003

Description: bejelentkezés tesztelése

Step0: Nyissuk meg az oldalt és kattintsunk a bejelentkezésre

Step1: Az username szövegdobozba írjuk be a \$UNAME szöveget

Step2: Az password szövegdobozba írjuk be a \$PASS szöveget

Step3: Nyomjuk meg az \$LOGIN gombot

Step4: Ellenőrizzük az eredményt. Elvárt eredmény: \$MSG üzenet és továbbítás a \$SITE oldalra

ID: TP_0002

TestCases: TC_0004, TC_0005, TC_0006, TC_0007

Description: jegyzet felvételének tesztelése

Step0: Nyissuk meg az oldalt, jelentkezünk be és válasszuk ki az „Új jegyzet felvétele” menüpontot

Step1: Az title szövegdobozba írjuk be a \$TITLE szöveget

Step2: Az text szövegmezőbe írjuk be a \$TEXT szöveget

Step3: A legördülő menüből válasszuk ki a \$DIR mappát

Step4: Nyomjuk meg az \$ADD gombot

Step4: Ellenőrizzük az eredményt. Elvárt eredmény: \$MSG üzenet és továbbítás a \$SITE oldalra

Tesztesetek

ID: TC_0001

TP: TP_0001

Description: bejelentkezés tesztelése helyes felhasználónév-jelszó párra

Input(s): \$UNAME=testuser ; \$PASS=testpass

Action: nyomjuk meg az „Belépés” gombot

Expected output: a megjelenő üzenet: \$MSG=„Sikeres bejelentkezés!”, továbbítás a \$SITE=home.jsp oldalra

ID: TC_0002

TP: TP_0001

Description: bejelentkezés tesztelése helyes felhasználónév, rossz jelszó párra

Input(s): \$UNAME=testuser ; \$PASS=tpass

Action: nyomjuk meg az „Belépés” gombot

Expected output: a megjelenő üzenet: \$MSG=„Hibás felhasználónév vagy jelszó!”, továbbítás a \$SITE=index.jsp oldalra

ID: TC_0003

TP: TP_0001

Description: bejelentkezés tesztelése SQL injection esetén

Input(s): \$UNAME=DROP TABLE user; ; \$PASS=DELETE FROM user WHERE 1

Action: nyomjuk meg az „Belépés” gombot

Expected output: a megjelenő üzenet: \$MSG=„Hibás felhasználónév vagy jelszó!”, továbbítás a \$SITE=index.jsp oldalra (és az adatbázis megmarad)

ID: TC_0004

TP: TP_0002

Description: jegyzet felvételének tesztelése cím nélkül

Input(s): \$TITLE=<emptystring> ; \$TEXT="Első jegyzet"

Action: válasszuk ki a \$DIR="Jegyzeteim" mappát a legördülő menüből és nyomjuk meg az \$ADD=„Mentés” gombot

Expected output: a megjelenő üzenet: \$MSG="Kötelező címet megadni!", továbbítás a \$SITE=<aktuális> oldalra

ID: TC_0002

TP: TP_0002

Description: jegyzet felvételének tesztelése azonos címmel

Input(s): \$TITLE="Első jegyzet" ; \$TEXT="Első jegyzet"

Action: válasszuk ki a \$DIR="Jegyzeteim" mappát a legördülő menüből és nyomjuk meg az \$ADD=„Mentés” gombot

Expected output: a megjelenő üzenet: \$MSG="Létező jegyzetnév!", továbbítás a \$SITE=<aktuális> oldalra

ID: TC_0003

TP: TP_0002

Description: jegyzet felvételének tesztelése mappa megadás nélkül

Input(s): \$TITLE="Második jegyzet" ; \$TEXT="Újabb jegyzet"

Action: ne válasszunk ki mappát a legördülő menüből, nyomjuk meg az \$ADD=„Mentés” gombot

Expected output: a megjelenő üzenet: \$MSG="Nincs mappa megadva!", továbbítás a \$SITE=<aktuális> oldalra

ID: TC_0004

TP: TP_0002

Description: jegyzet felvételének tesztelése sikeres esetben

Input(s): \$TITLE="Második jegyzet" ; \$TEXT="Újabb jegyzet"

Action: válasszuk ki a \$DIR="Jegyzeteim" mappát a legördülő menüből és nyomjuk meg az \$ADD=„Mentés” gombot

Expected output: a megjelenő üzenet: \$MSG="Sikeres felvétel!", továbbítás a \$SITE=<mappalistázás> oldalra