

Tema 1

Managementul continutului unitatilor de disc

Sa se creeze o aplicatie in C# si WPF (Windows Presentation Foundation) care sa permita managementul continutului aflat pe unitatile logice din calculator. Aplicatia trebuie sa permita crearea de categorii noi (ex: Muzica cu subcategoriile: Rock, Clasica, Punk, etc, fiecare din aceste subcategorii putand fi la randul ei compusa din sub-subcategorii, pana la orice nivel). Aplicatia trebuie sa poata incadra un anumit fisier sau director ca facand parte dintr-o anumita categorie (ex: directorul c:\downloads\Java ar putea fi incadrat in domeniul “documentatie”; **se va implementa si cazul în care un fisier/director face parte din mai multe categorii**).

Prezentarea continutului si a categoriilor trebuie facuta sub forma arborescenta (folosind controale TreeView).

Incarcarea initiala a continutului discurilor se va face in „mod lenes” (lazy load), adica initial se incarca doar lista de directoare radacina; la prima expandare a unui nod ne-frunza se va face incarcarea nodurilor copil (deci doar nivelul urmator).

Legatura se va face doar intre categorii frunza si directoare/fisiere.

Un nod categorie care contine copii nu va putea fi asociat nici unui continut de pe disc (i.e. nodul « Muzica » avand copii « Rock » si « Clasica », fiind prea general nu poate fi asociat niciunui folder sau fisier).

Daca un nod categorie are asocieri deja facute, nu i se vor mai putea adauga noduri copil, pana cand nu i se sterg toate asocierile.

Daca un director este asociat unei categorii, atunci se considera ca intregul sau continut (subdirectoare, fisiere la orice nivel) este asociat acelei categorii.

Aplicatia trebuie sa permita:

- *Adaugarea, modificarea si stergerea* de noi categorii (aplicatia porneste cu o categorie radacina care nu poate fi stearsa, doar redenumita)
- *Redenumirea* categoriilor
- *Salvarea asocierilor* create in format XML sau JSON si restaurarea lor.
- *Cautarea dupa urmatoarele criterii:*
 - toate entitatile care fac parte dintr-o anumita categorie;
 - parti din nume de fisiere si directoare (wildcards) – aceasta cautare nu trebuie sa implice activitate cu unitatile de sistem pe care se afla fisierele, se va face folosind doar datele salvate in documentul XML sau JSON.

Scenarii:

1. In faza de creare a asocierilor:
 - a. se asociaza o categorie cu directoare sau fisiere, dar pentru categoria respectiva exista si alte directoare/fisiere asociate; in acest caz se vor adauga noile asocieri la cele vechi (fara a aparea dubluri) – cu alte cuvinte, un fisier/director nu poate fi asociat de mai multe ori cu aceeasi categorie
 - b. se doreste vizualizarea asocierilor facute pana la un moment dat – acestea trebuiesc preluate din documentul XML sau JSON si *doar* directoarele/fisierele respective vor fi trecute pe “checked”; aceste asocieri pot fi modificate, imbogatite si sterse, prin actionarea casutelor de checkbox corespunzatoare; trecerea intre cele 2 stari: creare a asocierilor si revedere

a asociierilor facute se va face prin apasarea unui buton sau activarea unui submeniu, etc.

Nota: daca se asociaza un director cu o anumita categorie, se va considera ca intregul continut al directorului este asociat acelei categorii; pentru usurinta se poate memora doar asocierea intre categorie si acel director, omitandu-se mentionarea explicita a continutului sau.

2. In faza de preluare a asociierilor din fisierul XML sau JSON salvat pe disc:
La selectarea unei categorii vor fi afisate toate directoarele/fisierele care au fost asociate ei; arborele cu directoare/fisiere va fi refacut pe baza informatiei din XML sau JSON

In meniul de Help va fi o intrare de tip About in care va aparea numele studentului si un control Hyperlink a carui apasare va duce la lansarea automata a clientului de mail implicit cu adresa de email a destinatarului setata corect.

Sugestii de documentare:

Lucrul cu controlul treeview:

- <http://www.wpf-tutorial.com/treeview-control/introduction/>
- <https://www.codeproject.com/Articles/124644/Basic-Understanding-of-Tree-View-in-WPF>

Lucrul cu XML:

- <http://csharp.net-tutorials.com/xml/introduction/>
- <http://www.c-sharpcorner.com/uploadfile/mahesh/readwritexmltutmeli2111282005041517am/readwritexmltutmeli21.aspx>
- https://www.w3schools.com/xml/xpath_intro.asp (pentru expresii XPath)

Lucrul cu JSON:

- <http://www.newtonsoft.com/json> (recomandat)
- [https://msdn.microsoft.com/en-us/library/bb410770\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/bb410770(v=vs.110).aspx)
- https://github.com/ServiceStackV3/mythz_blog/blob/master/pages/344.md

Termen de predare: sambata, 6 mai 2017.

Observatii:

1. Temele copiate, „lucrate impreuna cu un coleg” sau pentru care nu se pot explica in mod convingator functionalitatile implementate vor primi **nota 1**, fara posibilitatea de a fi prezentate din nou.
2. Se noteaza cu **1 pct** din nota finala a acestei teme lucrul cu **{Binding}**.
3. Se acorda **1 pct** din oficiu.

Model de interfata (orientativ):

