

BookLab

by RENware Software Systems

BookLab System

(c) 2023 RENware Software Systems

BCAT catalogul cartilor

Cuprins:

- [BCAT catalogul cartilor](#)
 - [Catalogul cartilor - baza de date](#)
 - [Sablonul pentru o carte noua](#)
 - [Informatii despre aplicatie](#)
 - [Catalogul cartilor - interfata UI](#)
 - [Note](#)

Catalogul cartilor - baza de date

Catalogul cartilor (Books Catalog) este baza de date ce contine informatii despre toate cartile ce sunt gestionate de sistemul BookLab. Aceasta baza de date este in format JSON.

Structura bazei de date este urmatoarea:

```
{
  "data": [
    {
      "id": int,
      "code": "str",
      "short_desc": "str",
      "description": "str",
      "created_date": "str",
      "created_by": "str",
      "notes": "str"
    }
  ]
}
```

Structura JSON reprezentind cartile efective este o lista (array) cu inregistrarile cartilor, lista ce este valoarea cheii "data". Cu alte cuvinte inregistratea cu informatiile aferente unei carti are urmatoarea structura:

```
{
  "id": int,
  "code": "str",
  "short_desc": "str",
  "description": "str",
  "created_date": "YYYY-MM-DD",
  "created_by": "str",
  "notes": "str"
}
```



Note referitoare la chei

- `id` este cheia unica (PK) a cartii ca intreg generat automat de sistem
- `code` este o cheie (AK) ce permite cautarea unei carti dupa un cod alfanumeric ("humanized code"); aceasta informatie de obicei folosita ca si cod al cartii
- `short_desc` este o descriere sumara a cartii; aceasta descriere este de obicei folosita ca titlu si in zonele de "header", "footer" ale sectiunilor cartii
- `description` este o descriere larga a cartii, de obicei reprezentind o detaliere a titlului
- `notes` permite inregistrarea libera a orine informatii doreste utilizatorul referitor la acea carte
- cheile `created_*` sunt intretinute de sistem si permit un audit sumar al cartii: "cine a creat cartea si cind"

Catalogul cartilor este localizat in directorul `docs/data/` ¹ iar informatiile sunt stocate intr-un fisier cu numele `books_catalog.json` si sunt in format `JSON` standard. Ele pot fi accesate si manipulate prin intermediul oricarei biblioteci dedicata manipularii darelor si fisierelor format `JSON`.

Sablonul pentru o carte noua



... UPCOMING...

Acesta reprezinta sablonul (template) folosit in procesul de creare a unei carti noi (proces `NEWB`). Acest sablon contine urmatoarele elemente:

- *sectiunile (documentele) de start ale cartii* - pagina de `index.md` si documentul PDF final, ambele goale dar minim necesare in inceperea cartii
- *scripturile comenzi management carte* - reprezinta scripturile cu comenzile necesare pentru managementul si "operarea" acelei carti:
 - generarea temporara (de lucru) a cartii pentru previzualizare
 - generarea si asamblarea finala a cartii
 - pornirea serverului de previzualizare (server local HTTP)
 - monitorizarea fisierelor locale in timpul previzualizarii pentru a detecta eventuale modificari facute de utilizator
- *baza de date a cartii* este baza de date cu detaliile unei carti, baza de date ce este referentiata din [sectiunea catalogul cartilor](#). Continutul brut al [bazei de date cu catalogul cartilor](#)

...tbd...

Informatii despre aplicatie

In directorul `docs/data/` ¹ se gaseste un fisier ce contine informatii referitoare la aplicatia (sistemul aplicational) curent utilizata, mai exact la ceea ce in termeni clasici se numeste "aplicatia instalata".

Aceste informatii sunt stocate intr-un fisier cu numele `app_info.json` si sunt in format `JSON` standard. Ele pot fi accesate si manipulate prin intermediul oricarei biblioteci dedicata manipularii darelui si fisierelor format `JSON`.

Informatiile despre aplicatie sunt referitoare la versiunea acesteia si data la care a fost activata:

```
{
  "activated_on": "YYYY-MM-DD",
  "version": "M.m.p"
}
```

Catalogul cartilor - interfata UI

Fisierul template in format [markown format](#) poate fi accesat [aici](#).

Din punct de vedere al **proiectului BookLab si nu al produsului BookLab rezultat**, trebuie remarcat ca sablonul contine portiuni cu structuri Jinja e sunt "protejate" la prima compilare prin includerea lor in sectiuni Jinja tip `raw` - `endraw` astfel incit se vor regasi ca atare in fisierul HTML rezultat astfel incit sa fie evaluate corespunzator la executie (run time).

Datele utilizate in template si ce vor fi randate Jinja sunt transmise prin intermediul variabilei `bcat_data`.

Note

1. in sectiunea (de)servita de serverul HTTP static, ratiunea fiind ca sa fie accesibila la requesturi din acesta catre el insusi fara a "tranzita" serverul dinamic (a se vedea [sectiunea *sysInit* din documentul *Landscape*](#)) ←←
-

Last update: September 14, 2023