

# BookLab

by RENware Software Systems

**BookLab System**

*(c) 2023 RENware Software Systems*

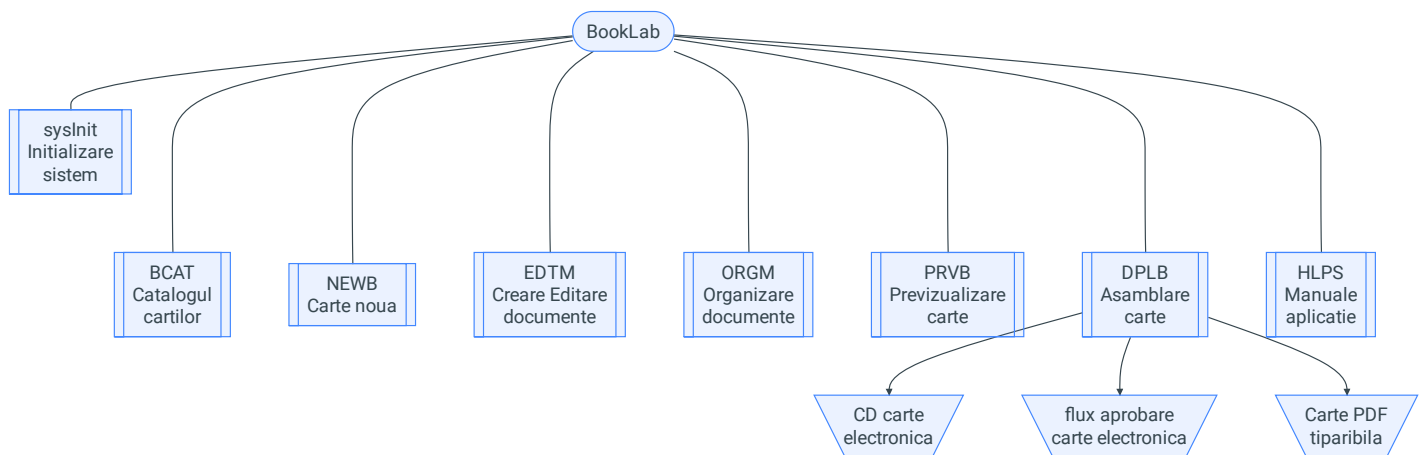
# Arhitectura sistemului

## Cuprins:

- Arhitectura sistemului
  - Introducere
  - sysInit Initializare sistem
  - BCAT catalogul cartilor
  - NEWB creare carte noua
  - EDTM preluare si editare continut
  - ORGM organizare structura carte
  - PRVB previzualizare si testare carte
  - DPLB asamblare publicare carte
  - HLPS help asistenta si manuale
  - Organizarea aplicatiei software
  - Note

## Introducere

Arhitectura sistemului pleaca de la structura de baza a acestuia, structura redată "pe scurt" mai jos. Diagrama puțin mai elaborată dar cu descrierea componentelor se găsește în [documentul 130.02-Overview.md](#), secțiunea "Structura si componenta BookLab" ce se recomandă a fi citită înainte de a începe studiul acestuia, pentru o mai bună înțelegere a lui.



Referitor la *locatia fizica a componentelor* se face remarcă ca toate componentele sistemului ce prezintă interfața vizuală cu utilizatorul au reprezentare atît în directorul `docs/` pentru partea de *front-end* (client side) cit și în rădăcina întregului sistem pentru partea de *back-end* (server-side). (Directorul `docs/` este generat prin compilarea aplicației în faza de development din directorul `doc_src/.`)



### Conceptele de server si client

- Se atrage atentia asupra tratarii corecte din punct de vedere conceptual a notiunilor de `server` si `client`.
- Chiar daca locatia de instalare a sistemului nu pare "un server cu multi cai putere", arhitectura conceptuala de client-server este pe deplin implementata cu respectarea tuturor principiilor aferente si o consistentizare a acestui lucru va permite o intelegere corecta a detaliilor tehnice ale sistemului de catre echipa tehnica de lucru.
- Locatiile fizice ale componentelor sunt prezentate NUMAI pentru *partea de SERVER*.
- Locatiile pentru *partea de CLIENT* sunt situate dupa o organizare clasica de site static incepind din directorul `docs/` iar codul este prezent in fisier `*.html` (in rare cazuri pot fi mai multe dar incep cu acelasi prefix).
- In general (daca nu specifica in clar altfel) `code-name` al componentelor este acelasi atat pentru SERVER cit si pentru CLIENT. Diferentierea intre acestea se intimpla deoarece raspund servere diferite pentru porturile aferente (80 client si 4111 server). Astfel pe ruta "normala" a unei componente, de ex ptr *NEWB*: `http://booklab.../newb` va raspunde intii componenta CLIENT (prezenta sub forma unui `index.html` in directorul `docs/newb/`).

## sysInit Initializare sistem

Pornirea sistemului comporta doua mari componente "on top", si anume:

- **serverul HTTP static** acesta va deservi partea de web compusa din "fisiere" statice. Acestea reprezinta fisierele HTML aferente interfetei UI dar si celelate fisiere "auxiliare" precum: CSS-uri, JS-uri, imagini, etc
- **serverul HTTP dinamic** ce opereaza in "background" (adica nu prezinta o interfata UI) si deserveste componentele care executa operatiile necesare (de ex create carte noua). Acest server este cunoscut si sub numele de `BookLab API`

De asemenea referitor la initializarea sistemului trebuie spus si ca **prima instalare a acestuia** se face cu o serie de programe / scripturi. Nu se intra in detalii deoarece procedura de instalare a sistemului este una cit se poate de clasica pentru instalările executate manual pe sisteme Linux iar pentru sisteme Windows acesta este tot una clasica, cu un simplu wizard "standardizat" si care executa scrierea in toate locatiile necesare (ex Registry).



### parametrii

- `code-name`: `sysInit`
- `SERVER locatie`: `sysInit/`
- `locatie scriptu PRIMA INSTALARE`: `setup/`
- `Document descriere detaliata`: [810.05a-sysInit\\_System\\_Process.md](#)

## BCAT catalogul cartilor

Acest modul este responsabil de catalogul (lista) cartilor si de sablonul de creare a unei carti. Astfel el contine urmatoarele elemente:

- **sablonul** implicit de creare a unei carti noi (si singurul in versiunea 1 a aplicatiei). Datele acestui sablon se afla in subdirectorul `bk_tmpl/`.
- **catalogul cartilor** ce este baza de date cu informatiile de baza despre carti. Acest catalog contine informatiile necesare pentru identificarea si referentierea unei carti in aplicatie. *Datele detaliate* aferente unei carti se gasesc in directorul aferent acesteia, director al carui nume se regaseste in acest catalog. Catalogul este stocat in format `JSON` si se gaseste in fisierul `books_catalog.json`.
- **interfata UI "Catalog carti"** ce este interfata CLIENT-UI<sup>1</sup> pentru managementul catalogului cartilor oferind operatii tip CRUD asupra bazei de date cu catalogul cartilor.

### parametrii

- *code-name*: `bcat`
- *SERVER locatie*: `books_metainfo/`
- *CLIENT-UI locatie*: `docs/bcat/index.html` (`doc_src/index.md` before 1-st compilation)
- *Document descriere detaliata*: `810.05a-bcat_System_Process.md`

## NEWB creare carte noua

-#TODO...



### parametrii

- *code-name*: newb
- *SERVER locatie*: newb /
- *CLIENT locatie*: --- #TODO nestabilit inca - update it
- *Document descriere detaliata*: [810.05a-newb\\_System\\_Process.md](#)

## EDTM preluare si editare continut

-#TODO...



### parametrii

- *code-name*: edm
- *SERVER locatie*: edm/
- *CLIENT locatie*: --- #TODO nestabilit inca - update it
- *Document descriere detaliata*: [810.05a-edm\\_System\\_Process.md](#)

## ORGM organizare structura carte

-#TODO...



### parametrii

- *code-name*: `orgm`
- *SERVER locatie*: `orgm/`
- *CLIENT locatie*: --- #TODO nestabilit inca - update it
- *Document descriere detaliata*: [810.05a-orgm\\_System\\_Process.md](#)

## PRVB previzualizare si testare carte

-#TODO...



### parametrii

- *code-name*: prvb
- *SERVER locatie*: prvb/
- *CLIENT locatie*: --- #TODO nestabilit inca - update it
- *Document descriere detaliata*: [810.05a-prvb\\_System\\_Process.md](#)



## DPLB asamblare publicare carte

-#TODO...



### parametrii

- *code-name*: `dplb`
- *SERVER locatie*: `dplb/`
- *CLIENT locatie*: --- #TODO nestabilit inca - update it
- *Document descriere detaliata*: [810.05a-dplb\\_System\\_Process.md](#)

## HLPS help asistenta si manuale

Manuale aplicatiei sunt de doua feluri (categorii) majore:

- **Manuale de utilizare**, acestea prezinta informatii pentru utilizarea curenta a aplicatiei. Catalogul acestora se poate accesa de la <SERVER NAME>/help/880.30-EUMA\_catalog.html.
- **Manuale de configurare**, acestea prezinta informatii pentru configurarea aplicatiei. Catalogul acestora se poate accesa de la <SERVER NAME>/help/880.30-ADMA\_catalog.html.

Pentru ambele tipuri de manuale se va prezenta un *catalog* ce contine lista tuturor manualelor disponibile in acea categorie. Lista contine numele manualului, o scurta descriere (daca numele nu este suficient de elocvent) si linkul catre manualul respectiv.

Manualele sunt localizate in sectiunea statica a aplicatiei (vezi si sectiunea "[Serverul HTTP static](#)") in directorul `docs/help/`.

## Organizarea aplicatiei software

Cod document metodologic: 810.05b Application Software Organization

In principiu aplicatia (sistemul) este organizata pe principiul *directoarelor de fisiere*. Aceasta permite o "rapida" regasire si recunoastere a informatiilor necesare chiar de la nivelul sistemului de operare (fara a necesita instrumente speciale ci doar comenzile existente in sistemul de operare). De asemenea o astfel de organizare este foarte apropiata de standardele clasice / originale ale aplicatiilor web.

Directoarele principale ale aplicatiei sunt enumerate la fiecare componenta din arhitectura si nu necesita explicatii suplimentare fata de cele scrise deja (vezi sectiunea [Arhitectura sistemului](#)). Detalii privind structura interna a fiecarui director aferent unei componente se regasesc in documentul de descriere detaliata aferent acelei componente. In situatii in care s-au considerat necesare diferite detalii pur specifice si "locale", in directorul componentei respective se gaseste un fisier `README_<cod-nume componenta>.md` ce prezinta aceste lucruri. Aceste documente se adreseaza in special dezvoltarilor aplicatiei (dar nu in mod obligatoriu) iar ca exemplu de ce poate contine un astfel de fisier este prezentarea modalitatii de pregatire a mediului local pentru dezvoltare.

## Note

- 
1. UI - desemneaza in general "Interfata Utilizator" (engl: User Interface) grafica, interactiva ce este afisata de catre browser (sau alta aplicatie GUI de acces) [↩](#)
- 

Last update: September 5, 2023