

Table of Contents

I SII Protectia Mediului

1 Sistem Informational Integrat Specializat pentru Protectia Mediului

- 1.1 Ce este SISPM
- 1.2 Aria de acoperire
 - 1.2.1 Componente generale
 - 1.2.2 Componente si module specifice protectiei mediului

I.I High Level Design

2 Arhitectura sistemului

- 2.1 Introducere
- 2.2 sysInit Initializare sistem
- 2.3 BCAT catalogul cartilor
- 2.4 BTMPL sablon carte noua
- 2.5 NEWB creare carte noua
- 2.6 EDTM preluare si editare continut
- 2.7 ORGM organizare structura carte
- 2.8 PRVB previzualizare si testare carte
- 2.9 DPLB asamblare publicare carte
- 2.10 HLPS help asistenta si manuale
- 2.11 Organizarea aplicatiei software
- 2.12 Note

I.II Detailed Level Design

3 BCAT catalogul cartilor

- 3.1 Catalogul cartilor - baza de date
- 3.2 Informatii despre aplicatie
- 3.3 Catalogul cartilor - interfata UI
- 3.4 Note

I.III Lista imaginilor

II About

4 Under construction page

I. SII Protectia Mediului

**SIISPM System**

(c) 2023 RENware Software Systems



1 Sistem Informational Integrat Specializat pentru Protectia Mediului

Cuprins:

- [Sistem Informational Integrat Specializat pentru Protectia Mediului](#)
 - [Ce este SIISPM](#)
 - [Aria de acoperire](#)
 - [Componente generale](#)
 - [Componente si module specifice protectiei mediului](#)

1.1 Ce este SIISPM

Acest sistem este o solutie gindita si elaborata pentru domeniul de "**Protectie a Mediului**".

Solutia este bazata pe platforma [NexGenAI minim versiunea 2.3.3 \(http://nexgenai.app/v2.3.3\)](http://nexgenai.app/v2.3.3) si foloseste cu precadere componentele acesteia.

1.2 Aria de acoperire



Terminologie - componente, module, pachete

Prin componente, module, pachete , aplicatii se va intelege la acest nivel simpla componenta logica software, nereperzentind nici un fel de "indicatie" privind modul de impachetare efectiv al elementelor software (cod)

Solutia acopera urmatoarele domenii de activitate:

1.2.1 Componente generale

- **(c1)** Portal web de servicii electronice (inclusiv Portal web general)
- **(c2)** Componenta pentru Managementul documentelor
- **(c3)** Componenta pentru aplicatii specifice GIS

1.2.2 Componente si module specifice protectiei mediului

- **(c4)** Componenta aplicativ Substante chimice
- **(c5)** Componenta aplicativ pentru Registrul National EMAS
- **(c6)** Componenta aplicativ avize de mediu, acorduri de mediu, autorizatii de mediu si autorizatii integrate de mediu
- **(c7)** Componenta aplicativ Protectia Atmosferei
- **(c8)** Componenta aplicativ Laboratoare
- **(c9)** Componenta aplicativ Monitorizare factori de mediu (radioactivitate)
- **(c10)** Componenta aplicativ pentru Registrul SEVESO
- **(c11)** Componenta aplicativ Emisii Industriale
- **(c12)** Componenta aplicativ EPRT (date aferente emisiilor si transferurilor de poluanti)
- **(c13)** Componenta aplicativ Sol si subsol
- **(c14)** Componenta aplicativ Ambalaje
- **(c15)** Componenta aplicativ Deseuri cu specializare pe produsele de tip Uleiuri
- **(c16)** Componenta aplicativ Colectare selectiva
- **(c17)** Componenta aplicativ Vehicule scoase din uz
- **(c18)** Componenta aplicativ Transport Deseuri
- **(c19)** Componenta aplicativ Deseuri baterii si acumulatori
- **(c20)** Componenta aplicativ Deseuri echipamente electrice si electronice
- **(c21)** Componenta aplicativ Situri Natura 2000
- **(c22)** Componenta aplicativ privind derogarile si capturile si uciderile accidentale ale speciilor strict protejate
- **(c23)** Componenta aplicativ Autorizarea activitatilor de recoltare/capturare si/sau comercializare a speciilor de flora si fauna salbatice
- **(c24)** Componenta aplicativ URS (alungarile, tranchilizarile si relocarile ori extragerile prin eutanasiere sau împuscare a exemplarelor de urs)
- **(c25)** Componenta aplicativ Gestiunea incidentelor provocate de speciile de interes cinegetic asupra culturilor agricole, silvice si animalelor domestice
- **(c26)** Componenta aplicativ ZOO
- **(c27)** Componenta aplicativ OMG si MMG (Organisme si Microorganisme modificate genetic)
- **(c28)** Componenta aplicativ Eticheta UE ecologica
- **(c29)** Componenta aplicativ GES (Registrul National al emisiilor de gaze)
- **(c30)** Componenta aplicativ Registrul national al operatorilor economici care nu se supun autorizarii de mediu
- **(c31)** Componenta aplicativ de gestionare a ecosistemelor si a serviciilor acestora în vederea sprijinirii luarii deciziilor de mediu

I.I High Level Design

**BookLab System**

(c) 2023 RENware Software Systems

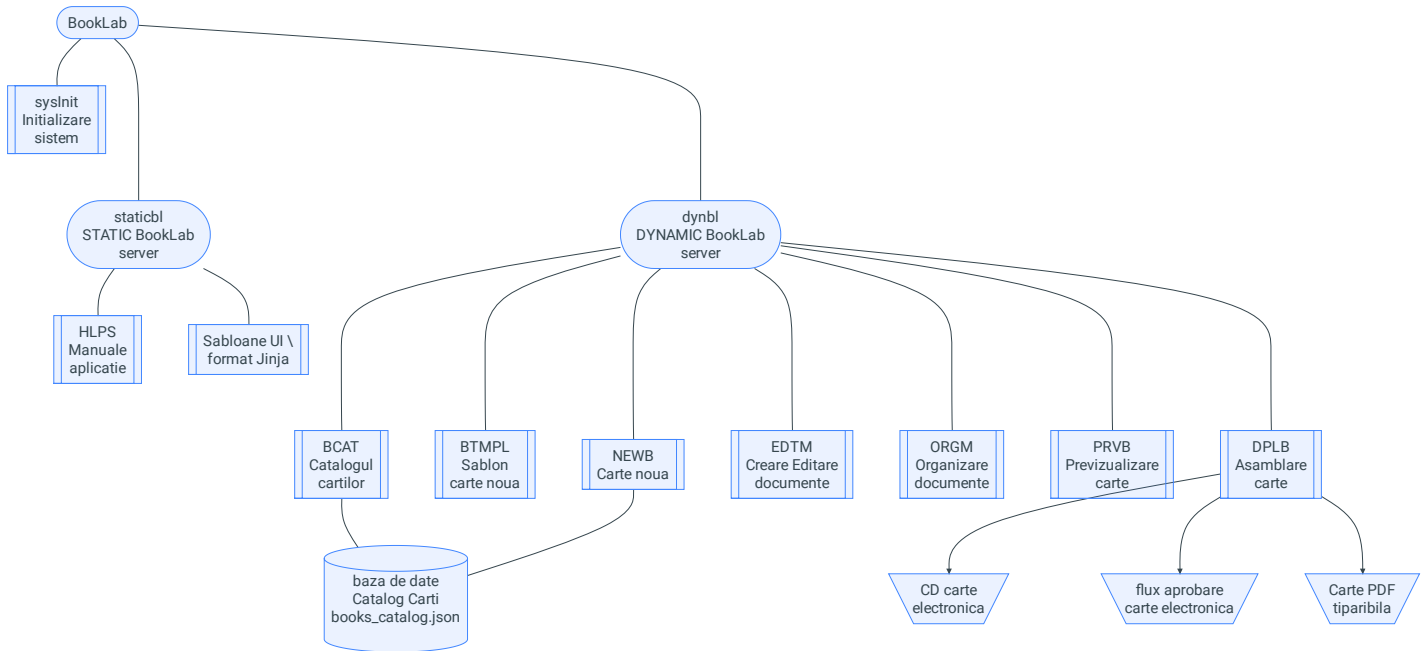
2 Arhitectura sistemului

Cuprins:

- [Arhitectura sistemului](#)
 - [Introducere](#)
 - [sysInit Initializare sistem](#)
 - [BCAT catalogul cartilor](#)
 - [BTMPL sablon carte noua](#)
 - [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](#)
-

2.1 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 developement din directorul `doc_src/.`)

⚠ Conceptele de server si client

- Se atrage atenția asupra tratării corecte din punct de vedere conceptual a noțiunilor de `server` și `client`. Arhitectura conceptuală de client-server este pe deplin implementată cu respectarea tuturor principiilor aferente și o consintținare a acestui lucru va permite o înțelegere corectă a detaliilor tehnice ale sistemului de către echipa tehnică de lucru.
- În general în toată documentația produsului prin termenul *CLIENT* se va înțelege și partea de *CLIENT FRONTEND* iar prin cea de *SERVER* și partea de *SERVER BACKEND*.
- Locațiile pentru *partea de server static* (componenta `staticbl`) sunt situate după o organizare clasică de site static începînd din directorul `docs/` iar codul "de start" este prezent în fisierul `index.html` (presupunînd că la utilizarea unui HTTP server anume la dorința clientului, acesta este configurat să funcționeze așa în lipsa specificării unui fisier în rutele URL).
- În general (dacă nu specifică în clar altfel) `code-name` al componentelor este același atît pentru SERVER cit și pentru CLIENT. Diferențierea între acestea se întîmplă deoarece răspund servere diferite pentru porturile aferente (80 client și 4111 server). Astfel pe ruta "normală" a unei componente, de ex ptr *NEWB*: `http://booklab.../newb` va răspunde întîi componenta CLIENT (prezintă sub forma unui `index.html` în directorul `docs/newb/`).

2.2 sysInit Initializare sistem

Pornirea sistemului comportă două mari componente "on top", și anume:

- `staticbl` - **STATIC BookLab serverul HTTP static** acesta va deservi partea de web compusă din "fișiere" statice. Acestea reprezintă fișierele HTML aferente interfeței UI dar și celelalte fișiere "auxiliare" precum: CSS-uri, JS-uri, imagini, etc

- **dynb1srv** - **DYNAMIC BookLab** serverul CGI dinamic ce permite rularea de scripturi Python server-side 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 instalarile 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`
- *locatie DYNAMIC BookLab:* `sysInit/`
- *locatie script PRIMA INSTALARE:* `setup/`
- *Document descriere detaliata:* [810.05a-sysInit_System_Process.md](#)

2.3 BCAT catalogul cartilor

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

- **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-UI¹ pentru managementul catalogului cartilor oferind operatii tip CRUD asupra bazei de date cu catalogul cartilor.

parametrii

- *code-name:* `bcat`
- *locatie DYNAMIC BookLab:* `docs/data/` cu baza de date catalogul cartilor `books_catalog.json`
- *locatie STATIC BookLab:* `docs/bcat/index.html` (`doc_src/index.md` before 1-st compilation)
- *Document descriere detaliata:* [810.05a-bcat_System_Process.md](#)

2.4 BTMPL sablon carte noua



-#TODO...

Aceasta componenta refera **sablonul** implicit de creare a unei carti noi (si singurul in versiunea 1 a aplicatiei).
Fisierele asociate acestui sablon se afla in subdirectorul `book_tmpl/`.

parametrii

- *code-name:* `book_tmpl`
- *locatie DYNAMIC BookLab:* `docs/book_tmpl/`
- *locatie STATIC BookLab:* ...nestabilit inca...
- *Document descriere detaliata:* [810.05a-btmpl_System_Process.md](#)

2.5 NEWB creare carte noua



-#TODO...


parametrii

- *code-name:* `newb`
- *locatie DYNAMIC BookLab:* `newb/`
- *locatie STATIC BookLab:* ...nestabilit inca...
- *Document descriere detaliata:* [810.05a-newb_System_Process.md](#)

2.6 EDTM preluare si editare continut



-#TODO...


parametrii

- code-name: edm
- locatie DYNAMIC BookLab: edm/
- locatie STATIC BookLab: ...nestabilit inca...
- Document descriere detaliata: [810.05a-edm_System_Process.md](#)

2.7 ORGM organizare structura carte



-#TODO...

parametrii

- code-name: orgm
- locatie DYNAMIC BookLab: orgm/
- locatie STATIC BookLab: ...nestabilit inca...
- Document descriere detaliata: [810.05a-orgm_System_Process.md](#)

2.8 PRVB previzualizare si testare carte



-#TODO...

parametrii

- *code-name:* prvb
- *port for running HTTP preview server:* 8111 (DIFFERENT than 80 which is reserved for main application)
- *locatie DYNAMIC BookLab:* prvb/
- *locatie STATIC BookLab:* ...nestabilit inca...
- *Document descriere detaliata:* [810.05a-prvb_System_Process.md](#)

2.9 DPLB asamblare publicare carte



-#TODO...

parametrii

- *code-name:* dplb
- *locatie DYNAMIC BookLab:* dplb/
- *locatie STATIC BookLab:* ...nestabilit inca...
- *Document descriere detaliata:* [810.05a-dplb_System_Process.md](#)

2.10 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/` .

2.11 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.

2.12 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) [↩](#)

I.II Detailed Level Design

**BookLab System**

(c) 2023 RENware Software Systems

3 BCAT catalogul cartilor

Cuprins:

- [BCAT catalogul cartilor](#)
 - [Catalogul cartilor - baza de date](#)
 - [Informatii despre aplicatie](#)
 - [Catalogul cartilor - interfata UI](#)
 - [Note](#)

3.1 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 darelui si fisierelor format `JSON`.

3.2 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"
}
```

3.3 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`.

3.4 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*](#)) ↩ ↩

I.III Lista imaginilor

II. About

4 Under construction page



UPCOMING...