FIȘĂ DE PRODUS

SIVADOC 2020

Platformă cloud-native de management al documentelor, înregistrărilor, fluxurilor de lucru și arhivei electronice (ECM/DMS)

Arhitectură bazată pe microservicii containerizate, orchestrate Kubernetes și instalate declarativ (Helm, Kustomize, CRD, GitOps)

SIMAVI

Cuprins

1. Prezentare generală SIVADOC 3
   1.1 Componentele funcționale și microserviciile SIVADOC 3
2. Arhitectura cloud-native bazată pe microservicii 5
   2.1 Principii arhitecturale 5
   2.2 Diagrama logică a microserviciilor 5
   2.3 Catalogul microserviciilor SIVADOC 6
   2.4 Comunicarea între microservicii 7
   2.5 Izolarea resurselor de persistență 7
3. Containerizare, orchestrare și deployment declarativ 9
   3.1 Containerizare 9
   3.2 Orchestrare Kubernetes 9
   3.3 Deployment declarativ: Helm, Kustomize, CRD și GitOps 9
   3.4 Medii de lucru și continuitate 11
4. Versionare independentă și cicluri de actualizare separate 12
5. Componenta Captură date 14
6. Componenta Registratură 17
7. Componenta Fluxuri de lucru 20
8. Componenta Semnătură electronică 23
9. Componenta Arhivă electronică 24
10. Componenta de asistență AI în lucrul cu documentele (modul opțional) 27
11. Componenta Raportare 29
12. Componenta Evenimente / Notificări 31
13. Nucleul de conținut sivadoc-core: depozit, căutare, configurare 32
   13.1 Gestiunea documentelor 32
   13.2 Căutare 33
   13.3 SIVADOC Studio – definirea tipurilor de documente 33
14. Securitate, gestiunea identității și GDPR 34
   14.1 Autentificare și autorizare 34
   14.2 Audit 34
   14.3 Protecția datelor cu caracter personal 34
15. Integrare și interfețe de programare (API) 35
16. Observabilitate și operare 35
17. Multi-tenant 35
18. Utilizabilitate și accesibilitate 36

1. Prezentare generală SIVADOC

SIVADOC 2020 este platforma SIMAVI de Enterprise Content Management / Document Management System (ECM/DMS) pentru gestiunea documentelor, înregistrărilor, fluxurilor de lucru și arhivei electronice. Platforma acoperă întregul ciclu de viață al documentelor: Captură date, Registratură, Fluxuri de lucru, Semnătură electronică, Arhivă electronică, Asistență AI în lucrul cu documentele, Raportare și Evenimente/Notificări.

SIVADOC este o aplicație cloud-native, construită din nucleul de conținut sivadoc-core (depozit de documente, metadate, versiuni, căutare, interfață web, captură/OCR, semnătură electronică, evenimente și notificări) și din microservicii independente pentru registratură, fluxuri de lucru, arhivă și raportare: fiecare rulează ca serviciu separat, împachetat ca imagine de container, orchestrat de Kubernetes și instalat prin chart Helm propriu. Fiecare serviciu are cod sursă, configurație, dependențe și bază de date proprii, propriul număr de versiune și propriul ciclu de release, astfel încât poate fi actualizat, scalat sau restaurat fără a afecta restul platformei.

Accesul utilizatorilor se face exclusiv prin browser (Microsoft Edge, Mozilla Firefox, Google Chrome, Apple Safari), prin conexiuni securizate TLS. Platforma poate rula integral on-premise, în centrul de date sau în cloud-ul privat al beneficiarului, fără dependențe de servicii de cloud public, și poate funcționa în mod offline pe perioadă nedeterminată.

1.1 Componentele funcționale și microserviciile SIVADOC

Tabelul de mai jos arată, pentru fiecare componentă funcțională, serviciul sau modulul SIVADOC care o realizează și resursele de persistență ale acestuia.

Componentă funcțională	Microservicii SIVADOC	Persistență izolată	Chart Helm propriu
Captură date	sivadoc-core (modulele de captură și OCR)	PostgreSQL sivadoc_core; stocare binare; cozi de lucru OCR	`sivadoc-core`
Registratură	sivadoc-registry-service	PostgreSQL sivadoc_registry (registre, secvențe, înregistrări, organigramă)	`sivadoc-registry-service`
Fluxuri de lucru	sivadoc-workflow-engine	PostgreSQL sivadoc_workflow (definiții, instanțe, sarcini, audit procese); definiții de proces (kjar) versionate separat	`sivadoc-workflow-engine`
Semnătură electronică	sivadoc-core (modulul de semnătură)	configurații și jurnal de semnare în sivadoc_core; HSM / magazin certificate	`sivadoc-core`
Arhivă electronică	sivadoc-archive-service (+ arhiva electronică acreditată, cap. 9)	PostgreSQL sivadoc_archive; arhiva de retenție la administratorul de arhivă acreditat sau în bucket WORM propriu	`sivadoc-archive-service`
Asistență AI	modul opțional: sivadoc-ai, sivadoc-llm-inference (cap. 10)	Index vectorial ai_vector_index; PostgreSQL ai_db	`sivadoc-ai, sivadoc-llm-inference`
Raportare	sivadoc-report-catalog-service, sivadoc-report-query-service, sivadoc-report-export-service, sivadoc-report-scheduler-service, sivadoc-report-ingest-service	câte o bază PostgreSQL per serviciu (sivadoc_report_catalog, _query, _export, _scheduler, _ingest); depozit de raportare (read model) alimentat din evenimente	`câte un chart per serviciu`
Evenimente / Notificări	sivadoc-core (modulele de evenimente și notificări)	sivadoc_core; temporal_db pentru execuția durabilă (Temporal)	`sivadoc-core`

Componentă funcțională	Microservicii SIVADOC	Persistență izolată	Chart Helm propriu
Servicii de platformă	sivadoc-core (depozit de conținut, căutare, previzualizare, SIVADOC Studio, audit, interfață web)	PostgreSQL sivadoc_core; index Elasticsearch; stocare obiect criptată AES-256	sivadoc-core; umbrella sivadoc-platform

2. Arhitectura cloud-native bazată pe microservicii

2.1 Principii arhitecturale

Arhitectura SIVADOC respectă principiile aplicațiilor cloud-native:

- **Nucleu de conținut și un microserviciu per capabilitate de business extinsă. Nucleul sivadoc-core (depozit, căutare, interfață, captură/OCR, semnătură, evenimente/notificări) și fiecare capabilitate extinsă (registratură, fluxuri de lucru, arhivă, raportare – aceasta din urmă ca cinci servicii) sunt servicii distincte, cu responsabilitate unică, cod sursă, imagine, chart Helm, pipeline și bază de date proprii.**
- **Bază de date per serviciu.** Fiecare microserviciu deține exclusiv datele sale (bază de date relațională, index, cache sau bucket de stocare obiect). Niciun serviciu nu accesează direct baza de date a altui serviciu; schimbul de date se face numai prin API sau evenimente.
- **Comunicare prin API Gateway și magistrală de evenimente.** Comunicarea frontend–backend trece prin API Gateway (reverse proxy cu caching, management identitate, retry, telemetrie, load balancing). Comunicarea între servicii folosește API-uri REST (sincron) și Apache Kafka (asincron, publish/subscribe), cu prioritizarea mesajelor critice, retry automat și cozi dead-letter.
- **Etape distincte Build – Release – Run.** Fiecare versiune de serviciu este construită o singură dată, versionată imutabil și promovată identic între medii (test, producție, DR); diferențele de mediu există exclusiv în configurație.
- **Configurație externalizată și individuală.** Configurația fiecărui serviciu este gestionată separat (ConfigMap, Secret, fișiere values Helm per mediu), versionată în Git și auditabilă.
- **Izolare la nivel de proces și de rețea.** Fiecare serviciu rulează în propriul container, cu propriile biblioteci, pe porturi TCP dedicate, cu politici de rețea (NetworkPolicy) care permit doar fluxurile declarate. Un defect sau o modificare locală nu afectează funcționarea globală.
- **Servicii auxiliare adresabile prin URL.** Cache, depozite de date și mesageria sunt expuse prin URL-uri configurabile și pot fi înlocuite fără modificarea codului.
- **Sincronizare prin Materialized View și CQRS.** Serviciile care au nevoie de date ale altor servicii (de exemplu raportarea) își construiesc propriile vederi materializate din evenimente; pentru raportare și urmărirea stării se aplică separarea comenzilor de interogări (CQRS).
- **Stocare obiect prin API REST.** Conținutul documentelor este păstrat în stocare de tip obiect compatibilă S3, accesată prin operații HTTP standard (GET, PUT, POST, DELETE).

2.2 Diagrama logică a microserviciilor

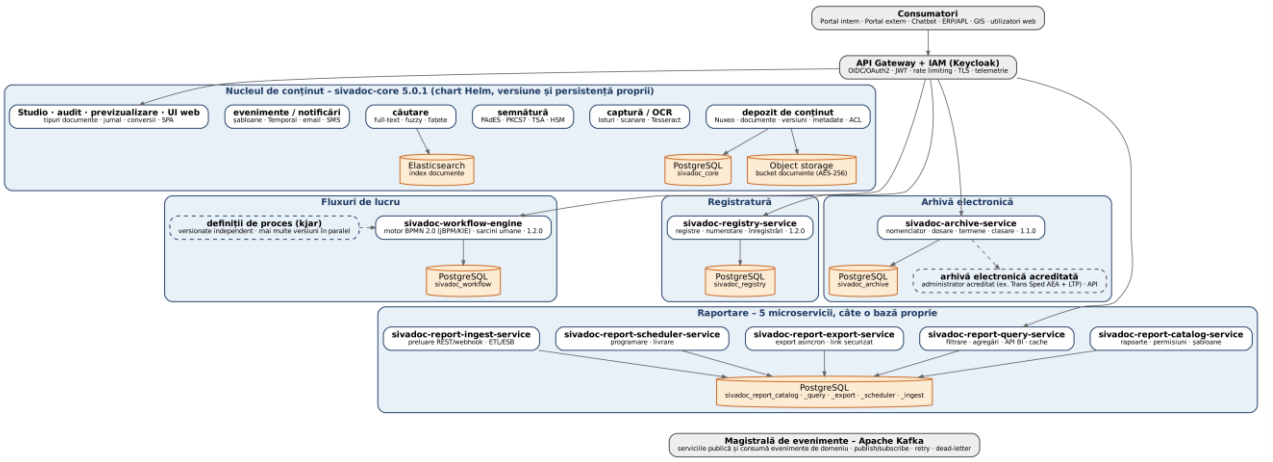


Figura 1 – Arhitectura logică SIVADOC: nucleul de conținut și microserviciile independente, fiecare cu persistență proprie

2.3 Catalogul microserviciilor SIVADOC

Serviciu	Responsabilitate	Tehnologie	Persistență proprie
sivadoc-core	Nucleul de conținut: depozit de documente (versiuni, metadate, ACL, check-in/check-out, colecții), căutare full-text, interfață web, captură/scanare/OCR, semnătură electronică, evenimente/notificări, SIVADOC Studio, audit, previzualizare	Java, Tesseract 5 / OpenCV (OCR), DSS (semnătură), LibreOffice / ImageMagick / FFmpeg (previzualizare), Temporal (evenimente)	PostgreSQL sivadoc_core; stocare obiect criptată AES-256; index Elasticsearch
sivadoc-registry-service	Registre configurabile per instituție, numerotare atomică pe registru și an, înregistrări cu emitent/destinatar/conținut, coduri de bare, organigramă, înlocuitori, căutarea înregistrărilor unui document	Java 17, Spring Boot, REST/OpenAPI, biblioteca sivadoc-service-chassis	PostgreSQL sivadoc_registry
sivadoc-workflow-engine	Execuția proceselor BPMN 2.0: instanțe, sarcini umane, temporizatoare, escaladări, reguli de decizie; definițiile de proces (kjar) sunt versionate și livrate independent, mai multe versiuni rulând în paralel	jBPM 7 (KIE Server), Java	PostgreSQL sivadoc_workflow
sivadoc-archive-service	Nomenclator arhivistic, termene de păstrare, dosare de arhivă, clasare, selecționare, blocare litigiu; transmiterea documentelor intrate în regim de retenție către arhiva electronică acreditată	Java 17, Spring Boot, biblioteca sivadoc-service-chassis	PostgreSQL sivadoc_archive
sivadoc-report-catalog-service	Rapoarte standard și ad-hoc, permisiuni RBAC per raport și categorie, vizualizări și șabloane salvate per utilizator/rol	Java 17, Spring Boot	PostgreSQL sivadoc_report_catalog
sivadoc-report-query-service	Filtrare, sortare, agregări, grupare, drill-down, API REST paginat pentru instrumente BI, cache cu TTL și invalidare	Java 17, Spring Boot; Knowage 9 (BI open-source) pentru tablouri de bord	PostgreSQL sivadoc_report_query (depozit de raportare)
sivadoc-report-export-service	Export CSV/XLSX/JSON/XML/PDF, execuție asincronă (coadă, retry, expirare), link securizat, mascarea datelor personale, jurnal de export	Java 17, Spring Batch	PostgreSQL sivadoc_report_export; cache Redis/Valkey
sivadoc-report-scheduler-service	Programare recurentă (cron), livrare pe e-mail, în arhivă sau prin API, istoricul rulărilor	Java 17, Spring Boot	PostgreSQL sivadoc_report_scheduler
sivadoc-report-ingest-service	Preluarea datelor prin REST/webhook și loturi ETL/ESB din arhivă, ERP, GIS, fluxuri, utilizatori și jurnale; validare; jurnal de preluare	Java 17, Spring Boot	PostgreSQL sivadoc_report_ingest
sivadoc-ai (opțional)	Asistent AI: RAG, sumarizare, comparare, extragere entități, agenți, task-uri asincrone – instalat numai când beneficiarul nu dispune de o platformă AI proprie (cap. 10)	Python, FastAPI, orchestrator agenți	Index vectorial ai_vector_index; PostgreSQL ai_db
sivadoc-llm-inference (opțional)	Servire modele LLM, speech-to-text și text-to-speech on-premise	Server de inferență pe GPU	Fără stare (modele pe volum read-only)

Chart-urile serviciilor sunt agregate în chart-ul umbrella sivadoc-platform (instalarea completă a platformei dintr-o singură versiune, cu dependințe fixate către chart-urile serviciilor) și folosesc chart-ul-bibliotecă sivadoc-

common (șabloane comune pentru Deployment, Service, HPA, PDB, NetworkPolicy). Serviciile Java sunt construite pe biblioteca proprie sivadoc-service-chassis (securitate Keycloak JWT, CORS, client HTTP, actuator), astfel încât fiecare serviciu conține doar modelul și logica propriului domeniu.

2.4 Comunicarea între microservicii

Comunicare sincronă. Fiecare microserviciu expune un API REST documentat OpenAPI 3, versionat în cale (de exemplu /api/registre/v1/inregistrari). Apelurile trec prin API Gateway, care aplică autentificarea (token JWT emis de Keycloak), limitarea ratei, timeout-uri, retry și circuit breaker.

Comunicare asincronă. Schimbările de stare relevante sunt publicate ca evenimente de domeniu pe Apache Kafka. Fiecare serviciu deține topic-urile în care publică; consumatorii își mențin propriile vederi materializate. Publicarea folosește modelul **transactional outbox** (evenimentul este scris în aceeași tranzacție cu datele serviciului), iar consumatorii sunt idempotenți.

Topic Kafka	Proprietar	Consumatori
`registratura.inregistrare.creata.v1`	sivadoc-registry-service	workflow-engine (pornire flux automat), raportare (ingest), nucleu (audit, evenimente)
`workflow.sarcina.alocata.v1`	sivadoc-workflow-engine	nucleu (notificări email + in-app, audit), raportare (timpi)
`workflow.proces.finalizat.v1`	sivadoc-workflow-engine	archive-service (clasare), raportare, Portal extern (actualizare stadiu cerere)
`arhiva.document.arhivat.v1`	sivadoc-archive-service	raportare, nucleu (evenimente – termene de păstrare, audit)
`captura.lot.exportat.v1`	sivadoc-core (captură)	nucleu (depozit), registry-service, raportare
`semnatura.document.semnat.v1`	sivadoc-core (semnătură)	nucleu (depozit, audit), archive-service
`continut.document.modificat.v1`	sivadoc-core (depozit)	nucleu (reindexare căutare), raportare, platforma AI (reindexare vectorială)
`evenimente.notificare.solicitata.v1`	sivadoc-core (evenimente)	nucleu (notificări)

Mesajele critice (de exemplu alocarea de sarcini) folosesc topic-uri cu prioritate și partiții dedicate; mesajele care eșuează după numărul configurat de reîncercări sunt mutate în topic-uri dead-letter și semnalate administratorului.

2.5 Izolarea resurselor de persistență

Fiecare serviciu cu stare are propria bază de date PostgreSQL, cu utilizator, parolă, schemă (migrări Liquibase) și politică de backup proprii; bazele pot fi găzduite pe un cluster PostgreSQL comun sau pe clustere dedicate per serviciu, declarate ca resurse Kubernetes de tip Cluster (operatorul CloudNativePG), după cerințele de izolare ale beneficiarului. Separarea este impusă de PostgreSQL și de rețea: un serviciu nu are credențiale pentru baza de date a altui serviciu (accesul încrucișat este refuzat), iar politicile de rețea blochează conexiunile nedeclarate.

Serviciu	Resursă de persistență	Date deținute exclusiv
sivadoc-core	PostgreSQL sivadoc_core; bucket object storage criptat AES-256; index Elasticsearch (reconstruibil din evenimente)	Documente, versiuni, metadate, ACL; tipuri de documente și formulare (Studio); loturi de captură; configurații și jurnal de semnare; șabloane de evenimente, jurnal de livrare și confirmări; jurnal de audit
sivadoc-registry-service	PostgreSQL sivadoc_registry (replicare sincronă în producție)	Registre, secvențe și contoare de numere (unicitate garantată tranzacțional), înregistrări, conexări, organigramă, perioade de indisponibilitate
sivadoc-workflow-engine	PostgreSQL sivadoc_workflow; artefacte kjar versionate în Git	Definiții de procese versionate, instanțe, sarcini umane, istoric și audit al execuției
sivadoc-archive-service	PostgreSQL sivadoc_archive; opțional bucket object storage cu object lock (WORM) pentru arhive de retenție gestionate intern	Nomenclatoare arhivistice versionate, termene de păstrare, dosare, registrul transferurilor către arhiva acreditată, procese verbale, marcaje de reținere
sivadoc-report-catalog / -query / -export / -scheduler / -ingest-service	Câte o bază PostgreSQL per serviciu (sivadoc_report_catalog, sivadoc_report_query, sivadoc_report_export, sivadoc_report_scheduler, sivadoc_report_ingest); Redis/Valkey pentru cache	Definiții de rapoarte și permisiuni; depozitul de raportare (read model construit din evenimente); cozi și rezultate de export; programări și istoricul rulărilor; jurnal de preluare
sivadoc-ai (opțional)	Index vectorial dedicat; PostgreSQL ai_db	Embeddings, jurnal de conversații și task-uri, feedback

Datele de care un serviciu are nevoie din alt domeniu (de exemplu numărul de înregistrare în arhivă sau starea fluxului în raportare) sunt replicate prin evenimente în vederi materializate proprii, nu citite din baza de date a altui serviciu.

3. Containerizare, orchestrare și deployment declarativ

3.1 Containerizare

- Fiecare microserviciu este livrat ca imagine OCI (compatibilă Docker), construită prin build multi-stage, pe imagine de bază minimală, rulând cu utilizator non-root și sistem de fișiere read-only.
- Imaginile sunt scanate de vulnerabilități, însoțite de SBOM și semnate criptografic înainte de publicarea în registry-ul OCI privat (Harbor) al platformei.
- Fiecare container expune sonde `liveness`, `readiness` și `startup`, metrici Prometheus și jurnale structurate (JSON) pe ieșirea standard.
- Componentele terțe (Tesseract, Knowage, Elasticsearch, Temporal, Kafka) rulează din imagini proprii, versionate separat.

3.2 Orchestrare Kubernetes

- Platforma rulează pe un cluster Kubernetes compatibil CNCF (Kubernetes, OpenShift sau Rancher RKE2), instalat în centrul de date sau în cloud-ul privat al beneficiarului.
- Serviciile fără stare rulează ca `Deployment` cu minim 2 replici, `HorizontalPodAutoscaler` (CPU, memorie sau lungimea cozii), `PodDisruptionBudget` și reguli anti-affinity între noduri, pentru o arhitectură fără punct unic de defectare.
- Serviciile cu stare (baze de date, Kafka, Elasticsearch) sunt gestionate prin operatori Kubernetes și resurse declarative (CRD), cu replicare și failover automat.
- Fiecare serviciu are `NetworkPolicy` proprie, `ServiceAccount` dedicat cu drepturi minime (RBAC) și `ResourceQuota` / `LimitRange` pentru izolarea resurselor.
- Lucrătorii OCR ai nucleului scalează orizontal în funcție de volumul de lucru din coadă; serviciul opțional de inferență LLM folosește noduri GPU dedicate.

3.3 Deployment declarativ: Helm, Kustomize, CRD și GitOps

Întreaga platformă este descrisă declarativ, în Git. Nu există pași manuali de instalare: starea fiecărui mediu este reconciliată automat din depozitul GitOps.

- **Helm – fiecare serviciu are propriul chart Helm, versionat SemVer și publicat ca artefact OCI; chart-ul umbrella sivadoc-platform agregă chart-urile serviciilor pentru instalarea completă, iar chart-ul-bibliotecă sivadoc-common furnizează șabloanele comune; helm lint și helm template rulează în pipeline pentru toate chart-urile. Chart-ul conține Deployment/StatefulSet, Service, Ingress/HTTPRoute, HPA, PDB, NetworkPolicy, ServiceMonitor și resursele CRD ale bazei de date și topic-urilor proprii.**
- **Kustomize** – diferențele între medii (producție, dezvoltare/testare, Disaster Recovery) sunt overlay-uri Kustomize și fișiere `values-<mediu>.yaml`, peste aceleași chart-uri și aceleași versiuni de imagini.
- **CRD** – bazele de date (`Cluster` CloudNativePG), topic-urile și utilizatorii Kafka (`KafkaTopic`, `KafkaUser` Strimzi), clusterul Elasticsearch (ECK), monitorizarea (`ServiceMonitor`, `PrometheusRule`) și aplicațiile GitOps (`Application`, `ApplicationSet` Argo CD) sunt declarate ca resurse Kubernetes.
- **GitOps (Argo CD)** – câte o resursă `Application` per microserviciu și per mediu sincronizează declarativ chart-ul și versiunea fixată în Git; orice abatere este detectată și corectată, iar revenirea la versiunea anterioară se face prin revert în Git.

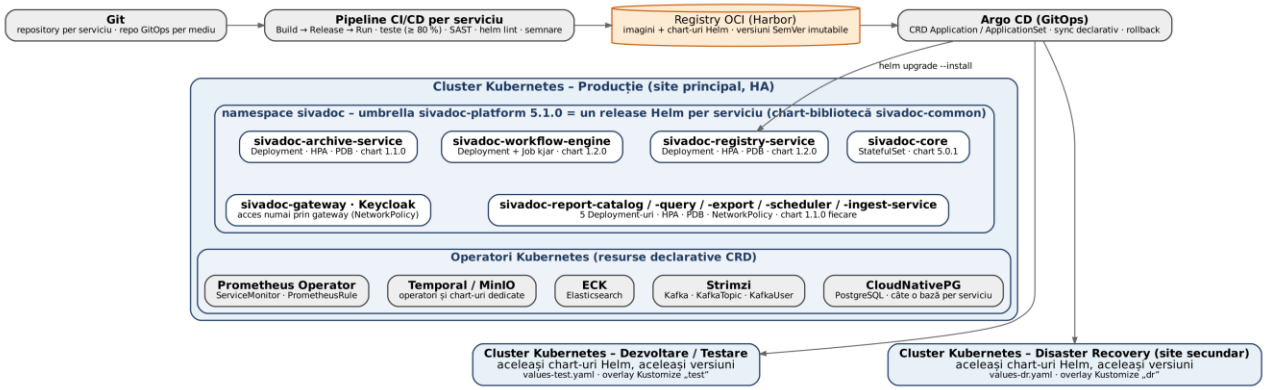


Figura 2 – Deployment SIVADOC: umbrella sivadoc-platform, un release Helm per serviciu, operatori CRD, GitOps pe trei medii

Structura chart-ului Helm al microserviciului sivadoc-registry-service

```
sivadoc-registry-service/
├── Chart.yaml                  # name, version (chart), appVersion (serviciu)
├── values.yaml                 # valori implicite
├── values-test.yaml            # mediu dezvoltare/testare
├── values-prod.yaml            # mediu producție
├── values-dr.yaml              # mediu Disaster Recovery
├── templates/
│   ├── deployment.yaml         # 2..N replici, probes, securityContext non-root
│   ├── service.yaml
│   ├── httproute.yaml          # expunere prin API Gateway
│   ├── hpa.yaml                # HorizontalPodAutoscaler
│   ├── pdb.yaml                # PodDisruptionBudget
│   ├── networkpolicy.yaml      # doar gateway, Kafka și propria bază de date
│   ├── configmap.yaml
│   ├── externalsecret.yaml     # credențiale din seif de secrete
│   ├── database-cluster.yaml   # CRD CloudNativePG: Cluster sivadoc-registry-db
│   ├── kafka-topics.yaml       # CRD Strimzi: KafkaTopic deținute de serviciu
│   ├── servicemonitor.yaml     # CRD Prometheus Operator
│   └── db-migration-job.yaml   # migrare schemă (Liquibase) la upgrade
└── tests/                      # helm test – verificare post-instalare
```

Extras Chart.yaml și values-prod.yaml

```
# Chart.yaml
apiVersion: v2
name: sivadoc-registry-service
description: SIVADOC – microserviciul Registratură
type: application
version: 1.2.0          # versiunea chart-ului
appVersion: "1.2.0"     # versiunea serviciului (imagine)

# values-prod.yaml
image:
  repository: harbor.local/sivadoc/sivadoc-registry-service
  tag: "1.2.0"
replicaCount: 3
autoscaling: { enabled: true, minReplicas: 3, maxReplicas: 10, targetCPUUtilizationPercentage: 70 }
resources:
  requests: { cpu: 500m, memory: 1Gi }
  limits:   { cpu: "2",  memory: 2Gi }
database:
  cluster: sivadoc-registry-db      # cluster PostgreSQL propriu
  instances: 3
  storage: 100Gi
  backup: { schedule: "0 0 * * * *", retention: 30d }   # backup orar (RPO ≤ 1h)
kafka:
  topics:
    - name: registratura.inregistrare.creata.v1
      partitions: 6
      replicas: 3
```

Extras resursă declarativă CRD – baza de date proprie a Registraturii

```
apiVersion: postgresql.cnpg.io/v1
kind: Cluster
metadata:
  name: sivadoc-registry-db
```

```
  namespace: sivadoc
  labels: { app.kubernetes.io/part-of: sivadoc, app.kubernetes.io/component: registry }
spec:
  instances: 3
  minSyncReplicas: 1
  maxSyncReplicas: 1
  storage: { size: 100Gi }
  bootstrap:
    initdb: { database: sivadoc_registry, owner: sivadoc_registry }
  backup:
    barmanObjectStore:
      destinationPath: s3://sivadoc-backup/sivadoc-registry-db
  affinity: { enablePodAntiAffinity: true, topologyKey: kubernetes.io/hostname }
```

Extras GitOps – aplicație Argo CD per microserviciu și per mediu

```
apiVersion: argoproj.io/v1alpha1
kind: Application
metadata:
  name: sivadoc-registry-service-prod
  namespace: argocd
spec:
  project: sivadoc
  source:
    repoURL: oci://harbor.local/charts
    chart: sivadoc-registry-service
    targetRevision: 1.2.0              # versiune fixată, independentă de alte servicii
    helm:
      valueFiles: [values-prod.yaml]
  destination: { server: https://k8s-prod.local, namespace: sivadoc }
  syncPolicy:
    automated: { prune: true, selfHeal: true }
    syncOptions: [CreateNamespace=false, ServerSideApply=true]
```

Extras Kustomize – overlay pentru mediul Disaster Recovery

```
# overlays/dr/kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
resources:
  - ../../base
patches:
  - target: { kind: Application, name: ".*-prod" }
    patch: |-
      - op: replace
        path: /spec/destination/server
        value: https://k8s-dr.local
      - op: replace
        path: /spec/source/helm/valueFiles/0
        value: values-dr.yaml
```

3.4 Medii de lucru și continuitate

- **Producție** – cluster Kubernetes în site-ul principal, cu minim 3 noduri de lucru și replicare sincronă a bazelor de date.
- **Dezvoltare / Testare** – cluster separat în site-ul principal, cu aceleași chart-uri și versiuni promovate din pipeline.
- **Disaster Recovery** – cluster în site-ul secundar, alimentat prin replicare asincronă a bazelor de date (CloudNativePG replica cluster), replicarea bucket-urilor de stocare obiect și MirrorMaker pentru Kafka; aceleași aplicații Argo CD cu overlay `dr`. Arhitectura permite obiective de tipul RPO ≤ 1 oră pentru datele dinamice și RTO de 1–4 ore.

4. Versionare independentă și cicluri de actualizare separate

Fiecare serviciu SIVADOC are propriul depozit Git, propriul pipeline CI/CD, propriul număr de versiune și propriul calendar de release. Actualizarea unui serviciu (de exemplu Registratura) nu presupune reconstruirea, retestarea completă sau reinstalarea altor servicii (de exemplu Arhiva sau Raportarea).

- **Versionare semantică (SemVer)** pentru fiecare imagine și fiecare chart: `MAJOR` pentru modificări incompatibile de API, `MINOR` pentru funcționalități noi compatibile, `PATCH` pentru corecții.
- **API-uri versionate** (`/v1`, `/v2`) și **evenimente versionate** (sufix `.v1` în numele topic-ului, schemă înregistrată în registrul de scheme). O versiune nouă de API coexistă cu cea veche pe durata perioadei de tranziție, astfel încât consumatorii se actualizează independent.
- **Teste de contract** (consumer-driven) rulează în pipeline-ul fiecărui serviciu și blochează un release care ar rupe compatibilitatea cu consumatorii declarați.
- **Migrări de schemă proprii (Liquibase), rulate de jobul de upgrade al chart-ului, numai pe baza de date a serviciului respectiv.**
- **Note de release și jurnal de modificări** per serviciu, publicate odată cu versiunea; registrul OCI păstrează toate versiunile imutabil, permițând revenirea rapidă.
- **Strategii de lansare** per serviciu: rolling update implicit, canary sau blue-green pentru serviciile critice, cu revenire automată dacă indicatorii de sănătate se degradează.



Figura 3 – Pipeline CI/CD per microserviciu: etape distincte Build, Release, Run

Extras pipeline CI/CD (sivadoc-registry-service)

```
# .gitlab-ci.yml – repository sivadoc-registry-service
stages: [build, test, release, deploy-test, deploy-prod]

build:
  stage: build
  script:
    - mvn -B verify                    # compilare + teste unitare (acoperire >= 80%)
    - mvn -B sonar:sonar               # analiză statică / SAST
    - buildah bud -t $IMAGE:$CI_COMMIT_SHA .

release:
  stage: release
  rules: [{ if: $CI_COMMIT_TAG =~ /^v\d+\.\d+\.\d+$/ }]
  script:
    - buildah push $IMAGE:$CI_COMMIT_SHA $IMAGE:${CI_COMMIT_TAG#v}
    - syft $IMAGE:${CI_COMMIT_TAG#v} -o spdx-json > sbom.json
    - cosign sign $IMAGE:${CI_COMMIT_TAG#v}
    - helm package chart/ --version $CHART_VERSION --app-version ${CI_COMMIT_TAG#v}
    - helm push sivadoc-registry-service-$CHART_VERSION.tgz oci://harbor.local/charts

deploy-test:
  stage: deploy-test
  script:
    - ./gitops-bump.sh test sivadoc-registry-service $CHART_VERSION   # commit în repo GitOps
    - ./contract-tests.sh

deploy-prod:
  stage: deploy-prod
  when: manual                         # aprobare release
  script:
    - ./gitops-bump.sh prod sivadoc-registry-service $CHART_VERSION
```

Exemplu de matrice a versiunilor

Tabelul prezintă versiunile curente ale serviciilor și chart-urilor (release 12.10.2026) și ilustrează faptul că serviciile evoluează independent: fiecare are propria versiune de serviciu și de chart și propriul ritm de release.

Serviciu	Versiune serviciu	Versiune chart	Ciclu de release	Compatibil cu API
sivadoc-core	5.0.1	5.0.1	semestrial	continut v1
sivadoc-registry-service	1.2.0	1.2.0	lunar + patch la nevoie	registre v1
sivadoc-workflow-engine	1.2.0	1.2.0	trimestrial; definițiile de proces (kjar) – independent	workflow v1
sivadoc-archive-service	1.1.0	1.1.0	lunar	arhiva v1
sivadoc-report-catalog-service	1.1.0	1.1.0	lunar	raportare v1
sivadoc-report-query-service	1.1.0	1.1.0	lunar	raportare v1
sivadoc-report-export-service	1.1.0	1.1.0	lunar	raportare v1
sivadoc-report-scheduler-service	1.1.0	1.1.0	lunar	raportare v1
sivadoc-report-ingest-service	1.1.0	1.1.0	lunar	raportare v1
sivadoc-platform (chart umbrella)	—	5.1.0	la fiecare release al unui serviciu	—
sivadoc-common (chart-bibliotecă)	—	1.1.0	la nevoie	—

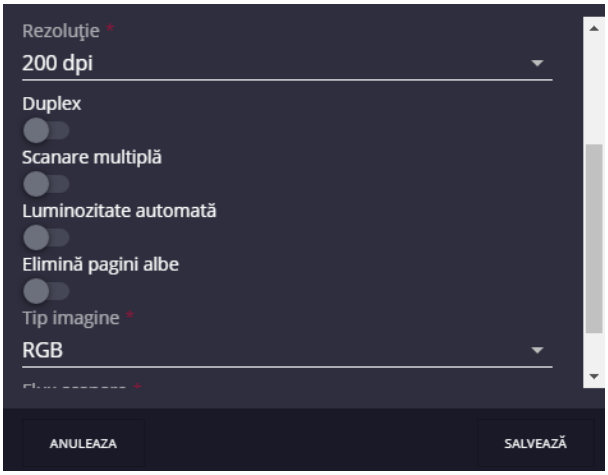
5. Componenta Captură date

Componenta Captură date populează arhiva operațională cu documente scanate sau importate, OCR-izate și indexate. Este realizată de modulele de captură (loturi, profile, indexare, export) și OCR (recunoaștere text și coduri de bare) ale nucleului sivadoc-core; lucrătorii OCR scalează independent, după lungimea cozii.

Fișa modulului	
Module	sivadoc-core: modulul de captură (loturi, profile, indexare, export) și lucrătorii OCR (fără stare, scalare după lungimea cozii)
Persistență	PostgreSQL sivadoc_core: loturi, profile de scanare, sesiuni salvate, jurnalul exporturilor
API / evenimente	REST /api/captura/v1; publică captura.lot.exportat.v1; folosește tipurile de documente definite în SIVADOC Studio
Deployment	Chart Helm sivadoc-core; agent de scanare local pentru stațiile cu scanere (TWAIN/WIA/ISIS)

Scanare și import

- Scanare directă din aplicație, cu alegerea scanerului disponibil pe stația utilizatorului și a profilului de scanare asociat.
- Profile de scanare configurabile: rezoluție, alb-negru/tonuri de gri/color, dimensiunea paginii, simplex sau duplex într-o singură trecere, eliminare pagini albe, luminozitate automată, în limita facilităților echipamentului.
- Suport pentru standardele de interfațare cu scanere TWAIN, WIA și ISIS; profilele create sau modificate se propagă automat la stațiile care le folosesc.
- Import de imagini și documente (JPG, TIFF, PDF și orice alt tip de fișier, inclusiv log, arhive, text, audio, video) din sistemul de fișiere, inclusiv prin drag and drop.
- Salvare în formate standard neproprietare (PDF, PDF/A, TIFF, JPEG, PNG) și compresie cu algoritmi neproprietari (JPEG, CCITT G4, ZIP/Deflate).
- Număr nelimitat de pagini și documente procesate, fără costuri de licență legate de volum.



Configurarea unui profil de scanare

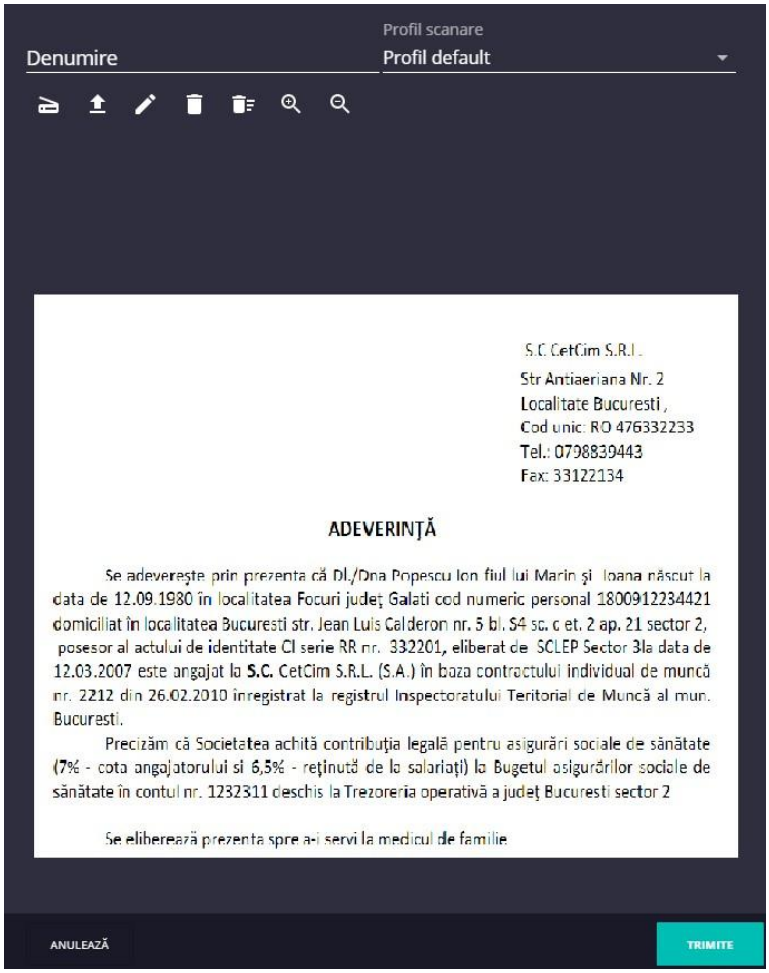
Prelucrarea loturilor

- Loturi cu număr practic nelimitat de documente, de tipuri diferite și cu seturi diferite de metadate în același lot.
- Recunoaștere și separare automată a tipurilor de documente dintr-un lot, pe baza unor loturi-exemplu de același tip folosite la antrenarea clasificatorului.

- Reordonare de pagini în cadrul unui document sau între documente prin drag and drop; unire și împărțire de documente; Undo/Redo pentru acțiunile asupra imaginilor.
- Îmbunătățirea imaginilor pentru creșterea calității OCR: îndreptare, eliminare margini negre, ajustare contrast, luminozitate și spectru de culori.
- Salvarea sesiunii de lucru și reluarea ulterioară.

Indexare și preluare metadate

- Completarea metadatelor în același ecran în care este afișat documentul.
- Metadate la nivel de lot, propagate automat către toate documentele din lot.
- Preluare automată de valori prin OCR zonal (șabloane OCR configurabile din administrare) și prin citire de coduri de bare (BCR); documentele cu același cod de bare sunt reunite automat și asociate înregistrării din registru.
- Liste de valori definite intern sau preluate prin conexiuni la baze de date, auto-complete cu valori folosite anterior, valori implicite.
- Validări configurabile: metadată obligatorie, metadată verificată prin script la completare, listă de valori; utilizatorii autorizați pot îmbogăți modelul existent sau defini scheme noi (prin SIVADOC Studio), iar schemele modificate se actualizează automat în componenta de captură.



Indexarea unui document cu preluarea valorilor prin OCR

Export în arhivă

- Configurarea modului de export: depozit și dosar de destinație, regula de denumire a documentelor pe baza metadatelor indexate, registrul în care se face înregistrarea.
- Semnarea electronică (simplă sau extinsă) a documentelor din lot înainte de export, opțional cu marcă temporală de la un furnizor certificat, prin modulul de semnătură.

- Toate configurările de scanare, indexare și export pot fi exportate într-un singur fișier și importate la alți utilizatori.
- Rapoarte predefinite de activitate pentru loturile transferate și jurnal al documentelor exportate.

6. Componenta Registratură

Registratura înregistrează documentele de intrare, ieșire și interne, generează numere unice de înregistrare și distribuie documentele către destinatari. Este realizată de microserviciul sivadoc-registry-service, care deține exclusiv registrele, secvențele de numere și înregistrările, în propria bază de date.

Fișa microserviciului	
Microserviciu	sivadoc-registry-service (Java 17, Spring Boot, sivadoc-service-chassis), minim 3 replici în producție
Persistență	Bază de date PostgreSQL proprie sivadoc_registry (replicare sincronă în producție). Contoarele de numere sunt gestionate tranzacțional în această bază, ceea ce garantează unicitatea numerelor indiferent de numărul de replici ale serviciului.
API / evenimente	REST /api/registre (registre, secvențe, înregistrări, rapoarte de registru, organigramă; înregistrările unui document: GET /api/registre/document/{documentRef}); publică registratura.inregistrare.creata.v1, registratura.document.distribuit.v1
Dependențe	sivadoc-core (conținutul documentelor, prin API), sivadoc-workflow-engine (pornirea fluxurilor, prin evenimente)
Deployment	Chart Helm sivadoc-registry-service, aplicație Argo CD proprie per mediu (cap. 3.3)

Registre și numerotare

- Crearea de registre direct din interfață, fără cunoștințe tehnice: nume, tip (intrare, ieșire, intern, arhivă, alt tip), an, index de start, schemă de metadate; număr nelimitat de registre.
- Secvențe de numerotare definite într-o interfață dedicată și atașate registrelor: prefix și sufix, intervale de numere, mod de incrementare, mai multe secvențe pe registru.
- Generare automată a numărului de înregistrare unic sau selecția unui număr preexistent, cu resetare automată la începutul fiecărui an calendaristic.
- Maparea seriilor de numere pe schemele de metadate definite în platformă și drepturi de securitate la nivel de registru.

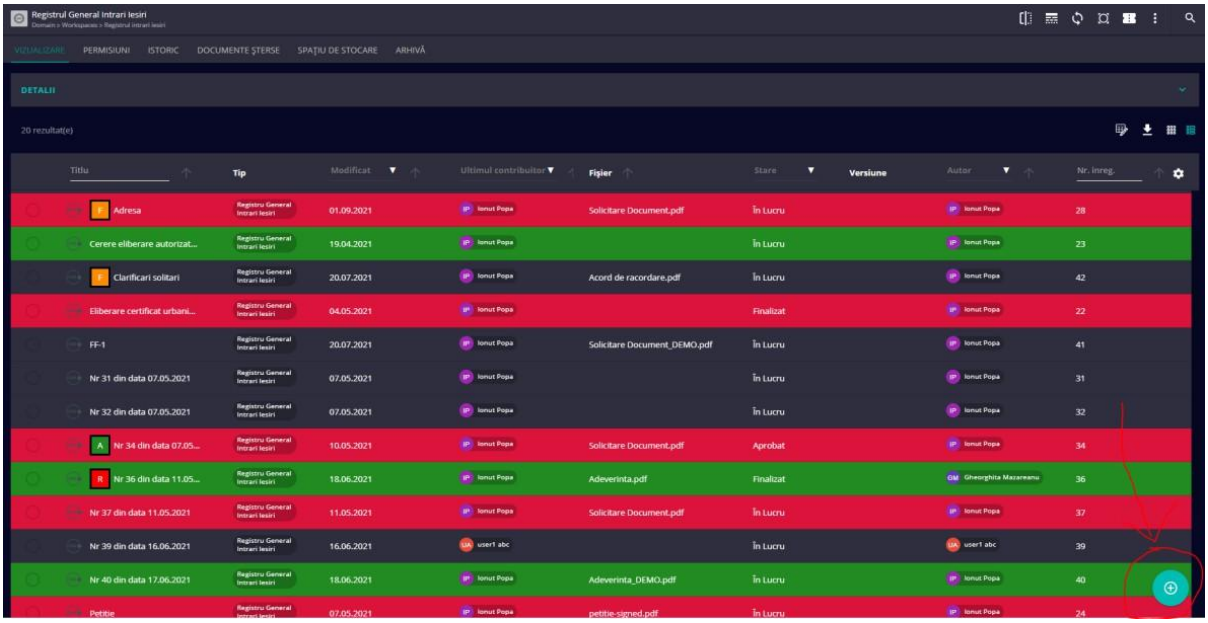


Definirea unei secvențe de registratură

Înregistrare și distribuție

- Înregistrarea documentelor de intrare, ieșire și interne, indiferent de canal: ghișeu, poștă, email, fax, portal extern. Se pot defini adrese de email la care documentele primite sunt înregistrate automat, cu preluarea expeditorului, subiectului și atașamentelor.
- Clasificarea documentelor ca intrare, ieșire sau interne și adăugarea lor pe oricare registru, cu oricare dintre schemele de metadate disponibile.
- Metadate suplimentare, inclusiv identificatori generați de sisteme externe (de exemplu numărul cererii din Portalul extern), după care documentele pot fi căutate și filtrate.

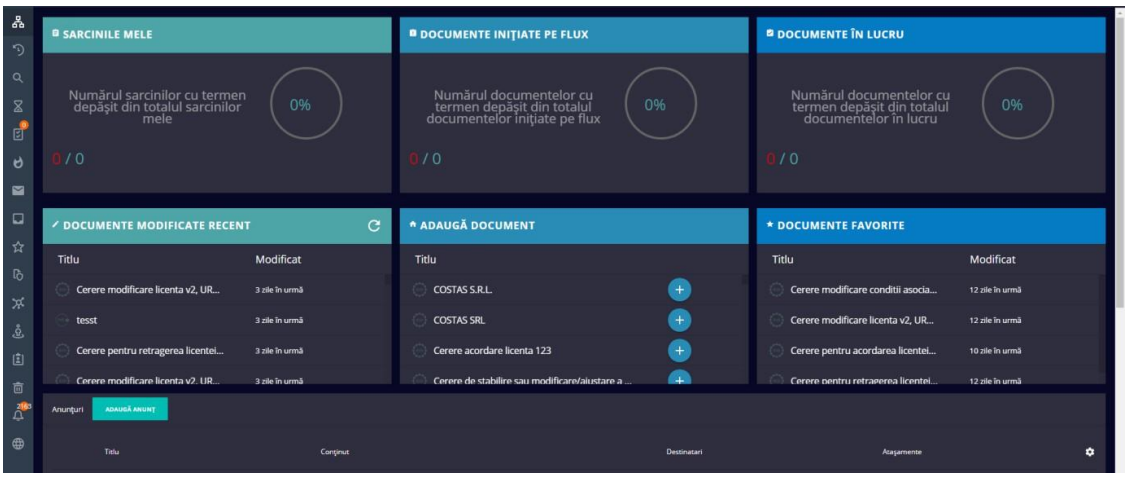
- Afișarea și tipărirea numărului de înregistrare ca cod de bare și injectarea codului de bare pe documentul electronic; generarea bonului/recipisei la depunere.
- Trimiterea documentului către unul sau mai mulți destinatari și pornirea manuală sau automată a unui flux de lucru, cu urmărirea pe baza istoricului, rezoluțiilor și notificărilor.
- Termen de rezolvare propus automat în funcție de tipul documentului și modificabil; semnalarea urgențelor în lista documentelor înregistrate.
- Conexarea înregistrărilor (legături bidirecționale), referințe către același document fizic și prevenirea duplicatelor prin sumă de control.



Lista documentelor înregistrate, cu evidențierea termenelor

Căutare și rapoarte de registru

- Căutare simplă după un cuvânt cheie (de exemplu numărul de înregistrare) și căutare avansată după criterii multiple și filtre, conform drepturilor de acces.
- Raportul registrului cu filtre, din care utilizatorul poate deschide metadatele, vizualiza documentul în viewer-ul integrat și descărca documentul; exportul raportului în PDF și alte formate.
- Tablou de bord cu imaginea de ansamblu a documentelor și fluxurilor în lucru, cu termene depășite sau apropiate.



Tabloul de bord al utilizatorului

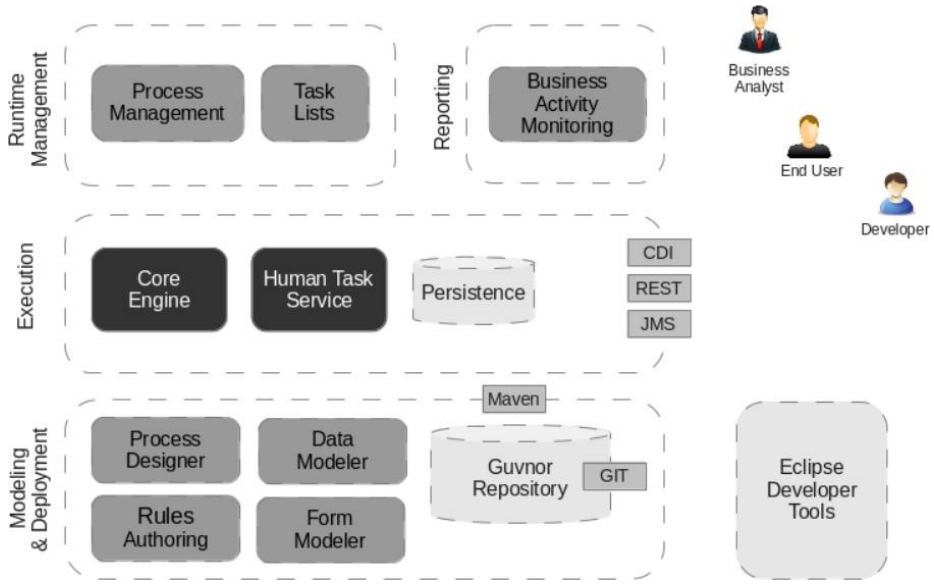
Structură organizatorică și înlocuitori

- Modelarea organigramei instituției; administratorii pot adăuga sau modifica departamente, aloca utilizatori și stabili un manager pentru fiecare departament.
- Declararea perioadei de indisponibilitate și a înlocuitorului; înlocuitorul primește notificare pe email cu perioada; utilizatorul indisponibil este marcat color în organigramă; funcția se dezactivează automat la expirarea perioadei sau manual, mai devreme.

7. Componenta Fluxuri de lucru

Fluxurile de lucru sunt realizate de microserviciul sivadoc-workflow-engine, care execută procesele BPMN 2.0 și sarcinile umane (jBPM / KIE Server), și de mediul web de modelare jBPM Business Central, din care definițiile de procese, formularele și regulile sunt livrate ca artefacte kjar, versionate independent de motor. Motorul rulează simultan mai multe versiuni ale aceluiași proces, iar instrumentele de modelare se actualizează fără întreruperea proceselor aflate în execuție.

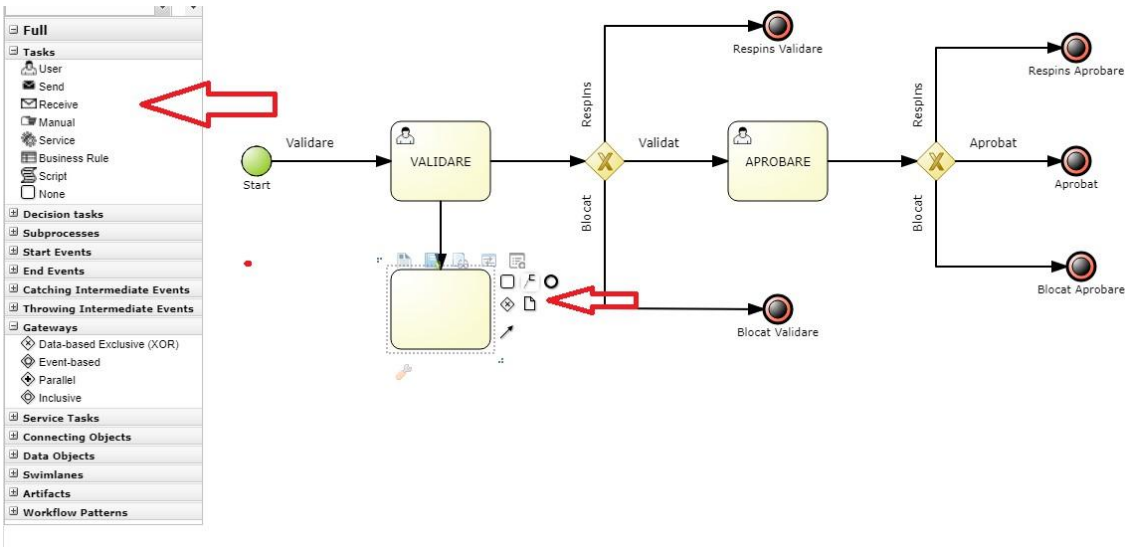
Fișa microserviciului	
Microserviciu	sivadoc-workflow-engine (motor BPMN 2.0, jBPM 7 – KIE Server, serviciu de sarcini umane); modelare în jBPM Business Central (process designer, data modeler, form modeler, rules authoring)
Persistență	Bază de date PostgreSQL proprie sivadoc_workflow (instanțe de procese, sarcini, audit execuție); definiții BPMN, formulare și reguli ca artefacte kjar versionate în Git
API / evenimente	REST `/api/workflow/v1` (pornire proces, sarcini, semnale, interogări), JMS/Kafka; publică `workflow.sarcina.alocata.v1`, `workflow.proces.finalizat.v1`; consumă `registratura.inregistrare.creata.v1`
Deployment	Chart Helm sivadoc-workflow-engine; definițiile de proces (kjar) se livrează printr-un Job de deploy separat, cu versiuni independente de motor



Componentele modulului de fluxuri: modelare, execuție, administrare

Modelarea fluxurilor

- Designer grafic web, fără cunoștințe de programare, care produce definiții BPMN 2.0; număr nelimitat de fluxuri și de pași.
- Încărcarea de definiții BPMN 2.0 create extern, după activarea depozitului de documente pentru fluxul respectiv.
- Pași de tip sarcină umană, sarcină sistem, decizie automată, acțiuni predefinite (tipărire, email, SMS, publicare, script) și porți (secvențial, paralel, exclusiv).
- Crearea de fluxuri noi și modificarea celor existente de către administratorii IT ai UAT-urilor, inclusiv fluxuri cu legături către alte instituții (apeluri de servicii prin ESB).
- Versionare, export, creare și ștergere de șabloane de fluxuri; simularea proceselor pentru estimarea timpilor, iterațiilor și costurilor.
- Configurarea formatului cererilor (formulare) de către utilizatori-cheie non-tehnici, pentru adaptarea rapidă la schimbări.



Editarea grafică a unui flux BPMN 2.0

Execuția fluxurilor și sarcinile

- Pornirea unui flux din orice context: detaliul documentului, rezultatele căutării, perspective, dosar; trimiterea manuală sau automată a documentelor pe flux predefinit sau ad-hoc.
- Prefiltrarea fluxurilor disponibile după tipul documentului și atribuirea automată a documentelor la fluxuri.
- Alocarea sarcinilor unui utilizator sau grup, sau după alocarea predefinită; distribuire automată în grup după încărcare, round robin sau valori de index; escaladare și funcția de superior.
- Notificări pentru sarcini în interfață și pe email, cu link direct către sarcină.
- Meniuri „Fluxurile mele”, „Sarcinile mele” și „Toate fluxurile”, cu filtrare, sortare după participanți și stare (activ, finalizat, termen depășit) și evidențierea fluxurilor necitite.
- Reprezentarea grafică a fluxului cu marcarea etapei curente; utilizatorul poate revendica, începe, finaliza sau elibera sarcina, poate adăuga comentarii, descărca și vizualiza conținutul și edita metadatele.
- Actualizarea stării de aprobare a documentului din sarcină; istoric dedicat pe document al tuturor fluxurilor parcurse.
- Rutare manuală la porțile de decizie sau automată după metadatele documentului.
- Fluxuri ad-hoc definite la rulare: activități (avizare, aprobare, semnare), participanți, distribuție secvențială sau paralelă, prioritate, escaladări.

Reguli de business pentru cererile din portal

- Reguli de optimizare a verificării și completării documentelor de emis și propunerea de texte predefinite în documentele emise.
- Reguli de interpretare a datelor GIS: verificarea suprapunerilor parțiale sau totale între straturi (de exemplu limite cadastrale – PUZ) și aplicarea ierarhiei PUD → PUZ → PUG pentru identificarea reglementărilor aplicabile, prin apeluri către Platforma GIS.
- Definirea tipurilor de taxe per tip de solicitare: taxe fixe (inclusiv de urgență) sau variabile după criterii (suprafață, urban/rural etc.), cu afișarea desfășurătorului taxelor.
- Generarea documentului-cerere din datele completate, precompletarea datelor solicitantului din profil, selecția beneficiarilor din profilul profesional.
- Semnarea electronică a cererii cu certificatul calificat definit în profil sau cu semnătură cu utilizare unică pusă la dispoziție de APL; atașarea de documente suplimentare; preluarea sistem-la-sistem a documentelor emise de alte instituții acolo unde există integrare.
- Preluarea datelor cadastrale din eTerra (ANCPI) pe baza numărului cadastral, afișate în GIS; plata online prin procesator sau prin ordin de plată/numerar cu încărcarea dovezii, sincronizată cu ERP; ajutor contextual pe secțiunile cererii.

Monitorizare și administrare

- Consolă de administrare pentru examinarea și monitorizarea instanțelor de procese (pornire, suspendare, reluare, migrare de versiune).
- Grafice de timpi de execuție per proces și de timp mediu per utilizator, pentru identificarea blocajelor; monitorizare BAM.



Statistici privind timpii de execuție a unui flux

8. Componenta Semnătură electronică

Semnarea, contrasemnarea și validarea semnăturilor sunt realizate de modulul de semnătură al nucleului sivadoc-core, folosit de toate celelalte componente (captură, fluxuri, arhivă, Portal extern) prin API.

Fișa modulului	
Modul	modulul de semnătură din sivadoc-core (biblioteca europeană DSS pentru formate AdES, PKCS#11 pentru token/HSM)
Persistență	sivadoc_core: configurații de furnizori și certificate, jurnalul semnărilor și validărilor
API / evenimente	REST `/api/semnatura/v1` (semnare, semnare lot, contrasemnare, validare, extragere informații); publică `semnatura.document.semnat.v1`
Deployment	În chart-ul sivadoc-core; acces la HSM prin rețea dedicată declarată în NetworkPolicy

- Integrarea mai multor furnizori de servicii de încredere, inclusiv servicii de semnătură electronică calificată ale furnizorilor acreditați, cu selectarea și configurarea furnizorului.
- Certificate software, pe token sau pe HSM; definirea certificatului în configurare doar prin proprietatea CN, celelalte informații fiind preluate automat din magazinul de certificate.
- Semnarea în lot cu certificatul calificat al administratorului de arhivă și marcă temporală, imediat înainte de stocarea în arhivă; PIN-ul certificatului este cerut o singură dată pentru întregul lot.
- Comunicare cu serverul de marcă temporală prin HTTPS cu autentificare mutuală, cu certificat client calificat.
- Semnarea fișierelor mai mari de 2 GB (semnare pe hash, transfer pe bucăți).
- Algoritm de hash SHA-256 sau superior, compatibil cu certificatul și acceptat de furnizor.
- Semnarea PDF/A cu păstrarea conformității PDF/A; celelalte tipuri de fișiere sunt încapsulate PKCS#7; se pot descărca atât fișierul PKCS#7, cât și originalul.
- Contrasemnarea fișierelor deja semnate, cu validarea certificatelor anterioare: verificare OCSP, verificarea emitentului în Trusted Root store și a corespondenței CN cu valoarea configurată.
- Extragerea informațiilor despre semnături și mărci temporale și afișarea detaliilor semnăturilor direct în interfață.
- Semnarea direct pe un document, pe pașii fluxurilor de aprobare sau dintr-o mapă care centralizează documentele de semnat, grupate pe tipuri de fluxuri, documente sau autori.

9. Componenta Arhivă electronică

Arhiva electronică este realizată de microserviciul sivadoc-archive-service, care deține nomenclatoarele arhivistice, termenele de păstrare, dosarele de arhivă, clasarea și selecționarea, în propria bază de date. Conținutul operațional este gestionat de depozitul nucleului sivadoc-core, iar vizualizarea de modulul de previzualizare. Pentru documentele intrate în regim de retenție, serviciul lucrează fie cu arhiva electronică acreditată a unui administrator de arhivă (secțiunea „Integrarea cu arhive electronice acreditate”), fie cu un spațiu de stocare WORM propriu.

Fișa microserviciului	
Microserviciu	sivadoc-archive-service (Java 17, Spring Boot, sivadoc-service-chassis); colaborează prin API cu nucleul sivadoc-core (depozit, previzualizare, semnătură) și cu arhiva electronică acreditată
Persistență	Bază de date PostgreSQL proprie sivadoc_archive; pentru arhivele de retenție gestionate intern, bucket de stocare obiect cu object lock (WORM), criptat AES-256
API / evenimente	REST `/api/arhiva/v1` (nomenclatoare, arhive, registrul arhivei, retenție, selecționare, joburi); publică `arhiva.document.arhivat.v1`, `arhiva.termen.expirat.v1`; consumă `workflow.proces.finalizat.v1`
Deployment	Chart Helm sivadoc-archive-service; joburile programate rulează ca CronJob Kubernetes din același chart

Nomenclator arhivistic și retenție

- Definirea nomenclatorului arhivistic într-o interfață grafică arborescentă, pornind de la structura organizatorică; versionarea nomenclatorului și stabilirea datei de intrare în vigoare a unei versiuni aprobate.
- Termen de păstrare pentru fiecare categorie din nomenclator; documentele nou intrate primesc termenul conform nomenclatorului activ la data emiterii; calculul termenului pe baza codului de nomenclator furnizat ca metadată.
- Păstrarea nemodificată a documentelor pe termen limitat sau permanent, în funcție de categoria din nomenclator.
- Registrul arhivei, cu număr unic de înregistrare, dimensiune, data arhivării, termen de păstrare, data ștergerii, nivel de clasificare și clasă de nomenclator; export CSV și PDF; proces-verbal de intrare generat automat.
- Cod unic de înregistrare pentru fiecare document intrat în arhivă și fișă de arhivare per document (număr unic, dimensiune, format, date de arhivare, metadate de caracterizare).

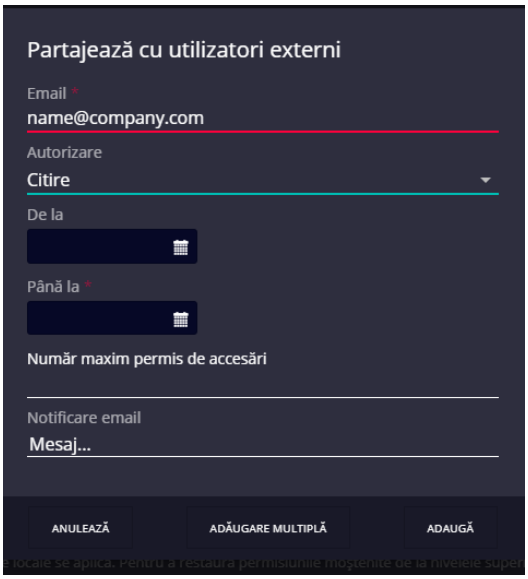
Arhive operaționale și de retenție

- Număr nelimitat de arhive, documente, scheme de metadate și metadate, în limita stocării; tipuri de metadate dată, numeric, șir de lungime fixă sau nedefinită, boolean.
- Definirea arhivelor operaționale sau de retenție direct din interfață, cu configurarea tipului de arhivă și a conexiunilor la baza de date și la stocare; acces la mai multe arhive cu o singură autentificare și roluri diferite per arhivă.
- Arhive de retenție conforme Legii 135/2007: conținutul este semnat obligatoriu cu certificat calificat și nu poate fi șters sau modificat nici de administrator (garanție tehnică prin object lock WORM); introducerea documentelor doar de către administratorul de arhivă, prin instrumente automate.
- Documente administrative de justificare a politicilor de securitate și a acțiunilor administrative, solicitate obligatoriu de sistem, cu metadate proprii (număr de la emitent, locul de păstrare a originalului) și asociate politicilor de securitate, pentru trasabilitatea schimbărilor de permisiuni.
- Extragerea automată și periodică a documentelor cu termen expirat, propunerea ștergerii către titularul arhivei și ștergerea doar după expirarea termenului și pe baza aprobării primarului/președintelui Consiliului Județean; marcare pentru comisia de selecționare.

- Marcaj de reținere (litigiu) care împiedică ștergerea la expirarea termenului; modificarea termenului de păstrare.

Lucrul cu documentele

- Adăugarea, ștergerea și editarea metadatelor documentelor din arhiva operațională direct din interfață; documente din șabloane prestabilite și fluxuri automatizate pentru creare și revizuire; populare prin componenta de captură.
- Viewer integrat în browser, fără programe auxiliare, pentru JPG, TIF, GIF, BMP, PDF, PDF/A, DOC, DOCX, RTF, TXT, HTML și pentru WAV, MP3, MP4; transfer doar al paginii consultate, miniaturi, zoom, rotire, fit-to-width, tipărire directă, vizualizarea conținutului arhivelor ZIP.
- Editor de text și editor tabelar integrate în platformă, fără aplicații externe.
- Navigare prin structuri arborescente generate din una sau mai multe metadate, configurabile dinamic, fără limită de număr sau adâncime; dosare virtuale definite prin reguli pe metadate, cu copertă dinamică ale cărei metadate se actualizează în timp real din documentele conexe.
- Favorite, link de acces per document transmisibil pe email, trimiterea documentului pe email din interfață; transfer de fișiere până la 5 GB în ambele sensuri, cu încărcare pe bucăți reluabilă.
- Colecții de documente partajate cu utilizatori externi aleși ad-hoc, pe perioadă limitată, cu verificare dublă prin email și număr de mobil, într-o zonă restricționată, fără ca documentele să părăsească locația de stocare; utilizatorii externi pot încărca documente în colecție.



Partajarea unei colecții cu utilizatori externi, cu perioadă și număr de accesări limitate

Procese asincrone și securitate

- Import bulk cu metadate dintr-un fișier CSV.
- Procese asincrone, programabile, care nu afectează timpul de răspuns: export documente (cu blocare la tipărire pentru anumite documente), actualizare metadate, aplicare politică de securitate, sincronizare cu serverul de directoare, actualizare statistici, export registru, export audit, copie off-line de siguranță cu documente și metadate.
- Jurnal de erori pentru procese, cu numărul unic al documentului și mesajul de eroare.
- Criptarea conținutului cu AES-256, cu chei configurabile per tip de document sau per depozit; aplicarea automată a regulilor de securitate la ingestie și ulterior, în funcție de metadate.

Integrarea cu arhive electronice acreditate

Pentru documentele care intră în regim de retenție în condițiile Legii nr. 135/2007, acolo unde cadrul legal sau beneficiarul impun păstrarea de către un administrator de arhivă electronică acreditat, sivadoc-archive-service transmite documentele prin API REST (JSON, HTTPS, token JWT), individual sau în loturi, către arhiva electronică

acreditată a unui prestator de servicii de încredere (de exemplu Trans Sped AEA + LTP), împreună cu codul de nomenclator, termenul de păstrare și identificatorul documentului din depozit. SIVADOC păstrează nomenclatorul arhivistic, dosarele și clasarea, iar documentele arhivate rămân consultabile din SIVADOC prin API (metadate, conținut semnat, raport de validare), cu drepturile verificate de arhiva acreditată.

10. Componenta de asistență AI în lucrul cu documentele (modul opțional)

Modulul opțional de asistență AI este realizat de microserviciul sivadoc-ai (orchestrare, RAG, agenți, task-uri) și de sivadoc-llm-inference (servirea modelelor de limbaj, recunoaștere și sinteză vocală), ambele rulând integral on-premise, pe noduri GPU din infrastructura beneficiarului. Modulul se instalează numai când beneficiarul nu dispune de o platformă de agenți AI proprie (vezi finalul capitolului).

Fișa modulului opțional	
Microservicii	sivadoc-ai (Python, FastAPI, orchestrator de agenți, motor de căutare hibrid); sivadoc-llm-inference (server de inferență pe GPU, modele open-weights, ASR/TTS)
Persistență	Index vectorial propriu `ai_vector_index`; PostgreSQL `ai-db` (conversații, task-uri, feedback, jurnal surse)
API / evenimente	REST și WebSocket `/api/ai/v1` (chat cu streaming, task-uri, export), documentate OpenAPI; consumă `continut.document.modificat.v1` pentru reindexare
Deployment	Chart-uri Helm `sivadoc-ai`, `sivadoc-llm-inference`; modelele sunt montate pe volum read-only și versionate separat de cod

- **Canale:** pagină web dedicată, modul de chat contextual integrabil în orice pagină SIVADOC sau în alte aplicații web, cu aspect configurabil (poziție, dimensiune, culori, pictograme, denumirea agentului); comunicare scrisă și vocală în română și engleză; canale suplimentare (telefonie, email) și conectare de canale noi prin REST API și WebSocket.
- **Lucrul cu documente:** încărcare sau selecție de documente din DMS, sumarizare, propunerea de întrebări, răspunsuri sub formă de text, rapoarte sau documente cu imagini, grafice și tabele, cu citarea explicită a surselor; mai multe task-uri simultane fără întreruperea conversației; task-uri recurente salvate și notificare pe email la finalizarea task-urilor lungi.
- **Surse și procesare:** documente, imagini, audio, baze de date, API-uri, RSS, data lake și baze de cunoștințe; speech recognition și OCR; raport de indexare (documente procesate, strategii, timpi, sarcini OCR/ASR); combinarea mai multor surse într-un singur task.
- **Permisiuni:** ACL per utilizator preluate din sursele integrate și respectate la generarea răspunsurilor; utilizatorul primește informații doar din documentele la care are acces; lista fișierelor folosite este jurnalizată.
- **Capabilități LLM:** extragere de informații, urmărirea referințelor interne, compararea documentelor și a versiunilor, analiza unor seturi mari de documente (de exemplu clauze contractuale), generarea de documente din șabloane sau exemple, interogări complexe în baze de date, interpretarea datelor (cauze, anomalii, tendințe, predicții), grafice bar și pie.
- **Căutare și RAG:** indexare multi-sursă, căutare după cuvinte cheie, semantică vectorială și după metadate; clasificare și extragere de entități (NER) la indexare; arhitectură RAG cu citarea surselor.
- **Orchestrare și configurare vizuală:** editor de tip diagramă pentru dialoguri, agenți și conectarea la surse, fără programare; execuție asincronă și paralelă.
- **Securitate:** roluri și grupuri, politici granulare, moștenirea drepturilor pe ierarhia organizațională, SSO/LDAP, jurnal de audit; funcționare integrală on-premise.
- **Integrare:** API dedicat pentru declanșarea task-urilor din alte aplicații, interfață încorporabilă, export PDF, Word, CSV.
- **Îmbunătățire continuă și raportare:** analiza conversațiilor (subiecte frecvente, lacune, feedback negativ) și recomandări; rapoarte de activitate și de performanță a agenților, comparabile pe intervale, exportabile PDF/XLSX/CSV.

- **Testare automată:** scenarii de test pentru răspunsurile LLM, rulate periodic și la fiecare actualizare, cu alertare la regresii.

Când beneficiarul dispune de o platformă de agenți AI proprie (de exemplu AISIV), funcțiile de asistență în lucrul cu documentele pot fi furnizate de aceasta prin API, cu respectarea drepturilor de acces din SIVADOC; în acest caz sivadoc-ai și sivadoc-llm-inference nu se instalează, iar SIVADOC expune platformei AI evenimentele de conținut și API-urile de depozit și căutare.

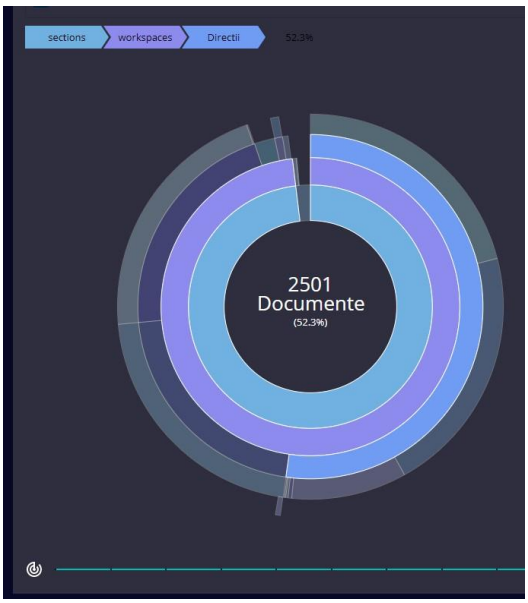
11. Componenta Raportare

Raportarea este realizată de cinci microservicii independente, fiecare cu bază de date proprie: sivadoc-report-catalog-service (catalogul rapoartelor standard și ad-hoc, permisiuni, vizualizări salvate), sivadoc-report-query-service (filtrare, sortare, agregări, drill-down, API paginat pentru BI, cache), sivadoc-report-export-service (export asincron CSV/XLSX/JSON/XML/PDF, link securizat, mascare), sivadoc-report-scheduler-service (programare recurentă și livrare) și sivadoc-report-ingest-service (preluarea datelor din evenimente, REST/webhook și loturi ETL/ESB). Depozitul de raportare este un model de citire construit din evenimentele celorlalte servicii (CQRS), astfel încât interogările analitice nu încarcă bazele de date operaționale; tablourile de bord interactive folosesc instrumentul BI open-source Knowage 9, alimentat din API-ul serviciului de interogare.

Fișa microserviciului	
Microservicii	sivadoc-report-catalog-service, sivadoc-report-query-service, sivadoc-report-export-service, sivadoc-report-scheduler-service, sivadoc-report-ingest-service (Java 17, Spring Boot / Spring Batch, sivadoc-service-chassis); Knowage 9 (BI open-source) pentru tablouri de bord
Persistență	Câte o bază PostgreSQL proprie per serviciu (sivadoc_report_catalog; sivadoc_report_query – depozitul de raportare cu vederi materializate; sivadoc_report_export; sivadoc_report_scheduler; sivadoc_report_ingest); Redis/Valkey pentru cache cu TTL și invalidare configurabilă
Surse de date	Evenimente Kafka din registratură, fluxuri, arhivă, captură, conținut; conectori ETL/ESB pentru ERP și GIS; REST/webhook pentru integrare near-real-time
API	REST paginat `/api/raportare/v1` și GraphQL; conectori ODBC/JDBC pentru instrumente BI (de exemplu Grafana, Tableau)
Deployment	Câte un chart Helm per serviciu (sivadoc-report-*-service); lucrătorii de export scalează după lungimea cozii

Rapoarte standard

- Rapoarte de utilizare: utilizatori activi și spațiul ocupat, pe fiecare arhivă de documente.
- Evoluția numărului și dimensiunii documentelor: luna anterioară, luna curentă, trimestrele anului curent, pe tipuri de documente.
- Volumul documentelor arhivate pe perioadă și arhivă; tipuri de documente și metadate; documente cu termen de păstrare expirat; activitatea utilizatorilor; istoricul modificărilor de metadate; documente în așteptare (aprobare, semnare, selecționare); șabloane și fluxuri parcurse; incidente și erori; rapoarte de audit și acces la documente.
- Statistici de depozit (cele mai descărcate documente, top autori, documente create și modificate pe săptămână) și statistici ale căutărilor.



Dashboard: distribuția documentelor pe depozite

Rapoarte personalizate, export și livrare

- Rapoarte ad-hoc create de utilizatorii autorizați: set de date sursă (arhivă, dosar, tip document), filtre dinamice pe metadate, coloane, sortare, grupare, format de export.
- Export PDF, CSV, XLSX, JSON, XML; salvare automată într-un dosar din arhivă sau trimitere pe email; programare recurentă (zilnic, săptămânal, lunar) cu livrare pe email, în folder sau către alte sisteme prin API.
- Afișare cu filtrare în timp real, căutare, sortare, agregări (sumă, medie, count, min/max), grafice (bare, linii, donut) și drill-down; salvarea vizualizărilor și șabloanelor per utilizator sau rol.

Performanță și operare

- Filtrare interactivă sub 3 secunde pentru seturi de până la 100.000 de rânduri, prin read model indexat și cache.
- Exporturi mari (până la circa 1 milion de rânduri) procesate asincron în coadă de joburi, cu retry, expirare, gestionarea erorilor, notificare la finalizare și link securizat de descărcare.
- Dashboard-uri near-real-time cu latență de actualizare sub 60 de secunde pentru sursele alimentate prin streaming.
- Monitorizarea sănătății joburilor și raportarea erorilor.

Securitate și audit

- Permisiuni per raport sau categorie (vizualizare, generare, modificare, distribuire, creare/ștergere rapoarte personalizate), RBAC și ABAC.
- Mascare/anonimizare configurabilă la export pentru datele cu caracter personal; transmitere doar prin canale securizate.
- Jurnalizarea fiecărui raport generat, exportat sau transmis: utilizator, dată/oră, parametri, format, destinație.

12. Componenta Evenimente / Notificări

Evenimentele de business și notificările sunt realizate de modulele de evenimente (șabloane, calculul termenelor, escaladare, orchestrare durabilă cu Temporal) și de notificări (livrare multi-canal cu retry și dead-letter) ale nucleului sivadoc-core.

Fișa modulelor	
Module	evenimente (motor de workflow distribuit Temporal pentru execuție durabilă și temporizatoare); notificări (Java, Spring Boot) – ambele în sivadoc-core
Persistență	sivadoc_core (șabloane, evenimente, stări, jurnal de livrare, confirmări de citire, preferințe); temporal_db (istoricul execuțiilor Temporal)
API / evenimente	REST `/api/evenimente/v1` și `/api/notificari/v1`, webhooks securizate în ambele sensuri; consumă evenimentele de domeniu ale celorlalte servicii; publică `evenimente.notificare.solicitata.v1`
Deployment	În chart-ul sivadoc-core; clusterul Temporal are chart propriu

- Șabloane de evenimente recurente sau critice (expirare contract, expirare termen de arhivare, revizuire periodică, reînnoire licență), gestionate printr-un compozitor de formulare, cu stări draft/activ/în revizuire, versiuni, suport multilingv și export/import prin API.
- Pentru fiecare șablon: denumire, tip de document, metadata care dă termenul, reguli de calcul (relative și absolute, intervale, condiții compuse, excluderea zilelor nelucrătoare), acțiune automată (notificare, marcarea documentului cu un statut, lansarea unui flux); testare în interfață prin previzualizare și dry-run cu destinatari de test.
- Notificări personalizate prin email, SMS, in-app, push și webhook către autor, roluri (mapate pe LDAP/AD/IdP) sau persoane din metadate; canalele se configurează per șablon și per destinatar, cu preferințe per utilizator; SMTP/SMTPS, gateway-uri SMS (inclusiv SMS Center-e instituționale), FCM/APNs.
- Conținut configurabil cu variabile dinamice, mesaj personalizat, subiect și semnătură organizațională; previzualizare, localizare, secțiuni condiționale, teste de livrare.
- Tablou de bord cu evenimente viitoare, în desfășurare și istoricul notificărilor: filtrare, sortare, căutare, drill-down, export CSV/XLSX, metrici (număr trimiteri, rată de succes, timpi).
- Listă de evenimente per document (denumire, document, dată, stare, utilizatori notificați) cu acțiuni rapide (reamintire, escaladare, marcare rezolvat) și link către document sau flux.
- Escaladare automată pe niveluri (după X zile către rolul A, după Y ore către rolul B), cu retry/backoff și oprire la rezolvare; marcarea manuală ca rezolvat, cu justificare opțională, care poate declanșa acțiuni ulterioare.
- Jurnal complet al notificărilor (destinatari, stare livrare, coduri gateway, reîncercări, inițiator, parametri) și confirmări de citire cu marcaj temporal.
- Rate limiting, batching și throttling pentru mesaje masive; livrare asincronă prin coadă, retry configurabil, dead-letter queue și alertarea administratorului.
- Ascultarea evenimentelor din DMS și din sisteme externe prin polling sau API, recepție prin webhook-uri securizate și atașarea de funcții personalizate (serverless) în fluxurile de evenimente.
- Observabilitate: metrici, dashboard operațional, jurnale centralizate, alerte la degradări (de exemplu creșterea erorilor SMS); TLS, criptare la rest, mascarea datelor personale, opt-out administrabil, conformitate GDPR.

13. Nucleul de conținut sivadoc-core: depozit, căutare, configurare

Nucleul de conținut sivadoc-core oferă capabilitățile comune folosite de componentele funcționale. Nucleul este livrat ca un singur serviciu, cu chart Helm, versiune (5.0.1) și persistență proprii (PostgreSQL sivadoc_core, stocare obiect, index Elasticsearch); modulele sale sunt:

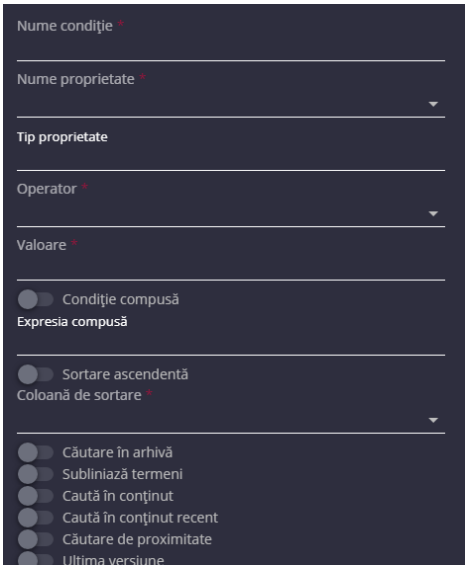
Modul	Rol	Persistență
Depozit de conținut	Depozitul de documente: documente, versiuni, metadate, ACL, check-in/check-out, colecții, coș de gunoi	PostgreSQL sivadoc_core; stocare obiect criptată AES-256
Căutare	Indexare full-text și căutare; reindexare din evenimente	Index Elasticsearch (reconstruibil din evenimente)
Previzualizare	Conversii de format, miniaturi, randare paginată	Fără stare
SIVADOC Studio	Definirea tipurilor de documente, metadatelor, formularelor și validărilor	PostgreSQL sivadoc_core
Audit	Jurnal centralizat de audit, append-only	PostgreSQL sivadoc_core (tabele append-only)
Interfață web	Interfața web a platformei	Fără stare

13.1 Gestiunea documentelor

- Stocarea documentelor și metadatelor asociate fluxurilor de business și a nomenclatoarelor; mai multe depozite de documente, organizate arborescent în dosare și subdosare, conform schemelor de indexare.
- Crearea automată de structuri de dosare în funcție de valorile metadatelor; metadate suplimentare valabile doar pe o anumită instanță, pe lângă cele standard ale tipului.
- Versionarea documentelor, metadatelor și fluxurilor, cu istoric complet, comparare și restaurare a versiunilor anterioare; versionare și audit granular al conținutului.
- Blocare/deblocare prin check-in/check-out și co-editare; ștergere doar dacă documentul nu este blocat; coș de gunoi cu restaurare, ștergere definitivă doar cu drept explicit; undo pentru mutare, ștergere, utilizare ca referință și editare de metadate.
- Etichete (tag-uri) cu motor de sugestii, căutare după toate sau oricare dintre etichete, import/export de etichete; colecții logice de documente fără mutarea lor; legături între documente.
- Abonare la notificări pe dosar sau document (creare, modificare, comentarii, publicare, ștergere, trimitere pe email, verificarea validității semnăturilor la arhivare); alerte configurabile pe dosar și document.
- Partajare externă prin email sau prin link cu perioadă de valabilitate și număr limitat de accesări.
- Orice tip de fișier, cu viewer integrat pentru text, imagini, audio și video; încărcare asincronă și pe bucăți (chunks) pentru fișiere foarte mari, reluabilă de la ultima bucată.
- Deschiderea, salvarea și completarea metadatelor direct din aplicații de tip Office (Word, Excel) și editarea documentelor de pe server; sincronizare bidirecțională cu un folder monitorizat (modulul Drive).
- Import și export în masă (sute/mii de documente) cu fișier-descriptor XML/Excel/CSV; export al unei colecții consultabile direct din browser de o entitate terță, fără instalare de programe.
- Replicarea conținutului expus (documente și metadate) între medii și instanțe ale platformei.
- Politici de retenție, arhivare și ștergere a documentelor, metadatelor și fluxurilor de business; reguli de migrare automată a conținutului între niveluri de stocare (inclusiv bibliotecă de benzi pentru retenție îndelungată).

13.2 Căutare

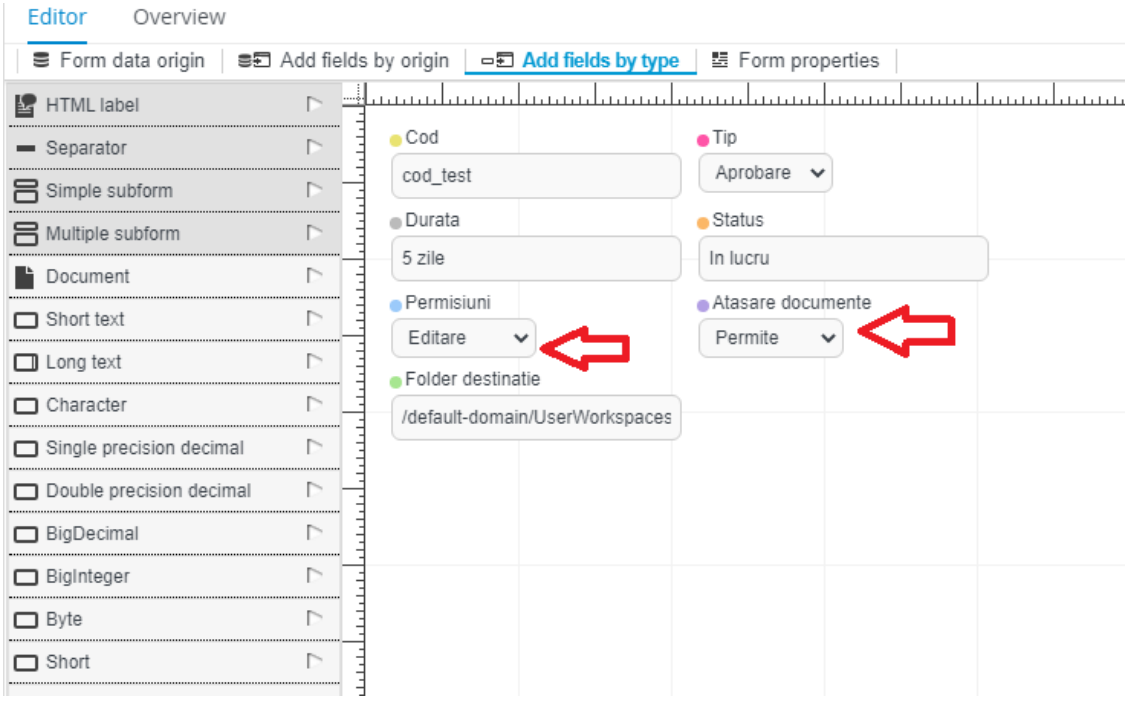
- Căutare simplă după metadate și în conținutul indexat, căutare avansată și complexă după criterii prestabilite, rafinare progresivă, căutări salvate ca șabloane.
- Căutare de tip fuzzy (toleranță la greșeli de scriere și diacritice) pentru utilizatorii interni și externi; căutare directă după numărul de înregistrare.
- Fațete și filtre pe metadate, sortare după criterii multiple; export al rezultatelor în Excel, PDF și alte formate.



Definirea unui filtru de căutare

13.3 SIVADOC Studio – definirea tipurilor de documente

- Definirea grafică a tipurilor de obiecte, a proprietăților lor și a relațiilor de constrângere dintre ele; număr nelimitat de tipuri de documente și metadate, de la atribute simple la formule sau valori preluate din sisteme externe.
- Formulare dinamice responsive, cu tab-uri, afișare condiționată, completare automată, formule de calcul, câmpuri de tip dată, text, radio, casetă de bifare, semnătură, imagine, link, butoane pentru linii repetabile.
- Reguli și validări pe câmpuri, preluarea datelor din surse externe și moduri de afișare per dispozitiv (desktop, tabletă, telefon).
- Nomenclatoare (vocabulare) administrabile: departamente, posturi, tipuri de operații, programe, proiecte, plăți, documente.



SIVADOC Studio – editorul de formulare

14. Securitate, gestiunea identității și GDPR

14.1 Autentificare și autorizare

- Autentificare centralizată prin Keycloak (OIDC/OAuth2, JWT), cu SSO și autentificare multi-factor; integrare în timp real cu LDAP/Active Directory, fără import de utilizatori; integrare SSO cu CAS și ForgeRock.
- Acces central bazat pe roluri, cu politici de securitate definite și administrate central; drepturi granulare pe funcționalități (meniuri), dosare și conținut (fluxuri, metadate, documente), prin liste de control al accesului.
- Administrator global și administratori de tenant; roluri definite în raport cu organigrama; delegarea responsabilităților între utilizatori.
- Parolele nu sunt stocate în clar și nici cu funcții hash nesigure; utilizatorii aplicației nu sunt utilizatori ai bazelor de date.
- Fiecare microserviciu validează tokenul la fiecare apel (zero trust între servicii); comunicarea între servicii este criptată mTLS; secretele sunt gestionate în seif și injectate la rulare.

14.2 Audit

Modulul de audit al nucleului colectează din toate serviciile, prin evenimente, operațiile efectuate (utilizator, operație, obiect, dată, oră, adresă IP) într-un jurnal append-only, filtrabil după utilizator și operație și exportabil asincron.

14.3 Protecția datelor cu caracter personal

- Identificarea automată a datelor cu caracter personal (CNP, email) în documentele încărcate, marcarea documentelor printr-un index și evidențierea vizuală a denumirii acestora.
- Metadate dedicate pentru marcarea datelor personale, căutare și filtrare după acestea, dashboard centralizat GDPR.
- Cenzurarea (blurarea) informațiilor sensibile pentru utilizatorii fără drept de acces; export automat criptat al datelor personale și metadatelor asociate.
- Criptarea documentelor din depozite cu AES-256, cu chei definite din interfața de administrare și chei diferite per tip de document sau depozit.

15. Integrare și interfețe de programare (API)

Fiecare microserviciu expune propriul API REST, documentat OpenAPI 3 și publicat în portalul pentru dezvoltatori al API Gateway. Integrarea cu alte sisteme (portaluri externe și interne, ERP, GIS, chatbot) se face prin API Gateway, prin componenta ESB și prin magistrala de evenimente.

Interfață	Utilizare
REST / OpenAPI 3 (JSON)	Interfața principală a fiecărui microserviciu: operații GET, POST, PUT, DELETE pe resursele proprii (documente, înregistrări, fluxuri, arhive, rapoarte, evenimente)
CMIS 1.1 (Browser binding, JSON)	Expunerea conținutului arhivat către sisteme externe după standardul CMIS
SOAP / WSDL	Integrare cu sisteme care nu suportă REST
Evenimente Kafka și webhooks	Integrare asincronă și near-real-time cu Portalul extern (actualizarea stadiului cererilor), ERP și Chatbot
S3 API	Acces la stocarea obiect pentru volume mari și pentru arhivarea pe termen lung
ODBC/JDBC, GraphQL	Conectori pentru instrumente BI externe (raportare)
Import/export fișiere (XML, CSV, Excel)	Integrare cu aplicații care nu expun servicii, migrare de date

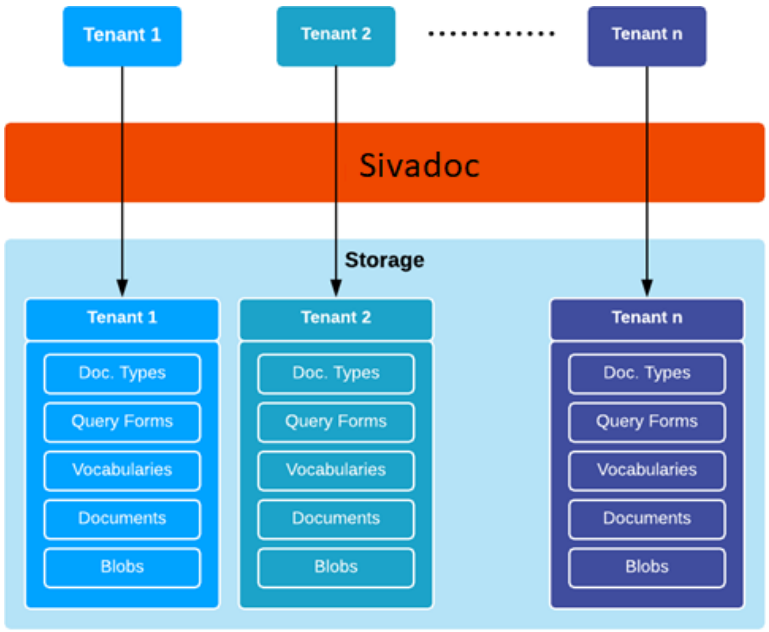
- Integrare cu Portalul extern: cererile depuse sunt înregistrate în sivadoc-registry-service, procesate în sivadoc-workflow-engine, iar schimbările de stare sunt publicate ca evenimente consumate de portal; în cazul revenirilor, noua versiune primește număr nou de intrare, legat de versiunile anterioare.
- Integrare cu furnizorul de servicii de arhivare electronică pentru transmiterea metadatelor cerute de lege; comunicare cu stocare obiect, NAS, SAN, HSM și biblioteci de benzi.
- Integrare cu Keycloak pentru fluxurile de aprobare a accesului și înrolare a utilizatorilor: fluxul este pornit de o acțiune din Keycloak, parcurs în motorul de fluxuri, iar rezoluția este transmisă înapoi.
- Expunerea de servicii web pentru acces la documentele arhivate și preluarea documentelor de arhivat se configurează fără scriere de cod; serviciile concrete se detaliază în etapele de analiză și proiectare.

16. Observabilitate și operare

- **Metrici:** fiecare microserviciu expune metrici Prometheus (rată de cereri, erori, latență, lungimea cozilor, metrici de business), colectate prin `ServiceMonitor`; dashboard-uri Grafana per serviciu.
- **Jurnale:** jurnale structurate colectate cu Fluent Bit și centralizate în Loki, corelate prin identificatorul cererii.
- **Trasare distribuită:** instrumentare OpenTelemetry, cu urmărirea unei cereri de la API Gateway prin toate microserviciile implicate.
- **Alertare:** reguli `PrometheusRule` per serviciu și Alertmanager, cu notificarea echipei de suport; integrare cu componenta de monitorizare și cu sistemul de Help desk.
- **Backup și restaurare per serviciu:** fiecare bază de date are politica proprie de backup (incremental continuu, backup complet zilnic) și poate fi restaurată independent la un moment dat (point-in-time recovery).

17. Multi-tenant

SIVADOC folosește conceptul de tenant pentru a deservi mai multe instituții dintr-o singură instalare, cu izolare completă: utilizatori, grupuri, nomenclatoare, filtre de căutare și documente electronice sunt vizibile numai în cadrul propriului tenant. Administratorii globali configurează platforma, iar administratorii de tenant gestionează utilizatorii, grupurile și nomenclatoarele propriei instituții.



Izolarea datelor pe tenanți

18. Utilizabilitate și accesibilitate

- Interfață web uniformă între module, cu componente statice cu aceeași dispunere, teme configurabile și tablou de bord personalizabil (carduri, ordine, secțiuni în funcție de drepturi).
- Documentație contextuală, mesaje de eroare care indică atât cauza, cât și modul de corectare.
- Interfață în limba română și engleză, cu formate de dată și numerice localizate.
- Interfață responsive pentru desktop, tabletă și telefon; navigare cu tastatura și contrast conform WCAG 2.1 nivel AA.