



Istituto Nazionale di Fisica Nucleare

LIBRERIE E STRUMENTI PER LO SVILUPPO SW

Bologna – 1 Dicembre 2021

Claudio Bisegni – Emanuele Turella
Direzione Sistemi Informativi

A QUESTA ATTIVITÀ STANNO PARTECIPANDO

CLAUDIO
Bisegni



EMANUELE
Turella

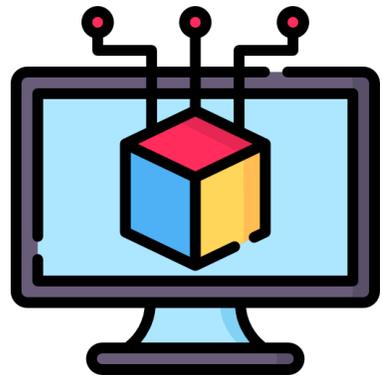


ANTONELLO
Paoletti

FRANCESCO
Serafini



LUCA
Sanelli



Back-end

BACKEND: BASE-SERVICE-LIB

- Costruita in cima a Spring Boot
- Autenticazione con JWT (OAuth2)
- Autorizzazione dei Rest Controller tramite guardia basata su entitlement
- Standardizzazione delle chiamate ad altri microservizi tramite user-token o application-token
- Gestione nativa del tracing+logging distribuito (ispirato a zipkin)
- Caching delle autorizzazioni utente
- Background job e Message queue service basato su Kafka
- Swagger/OpenAPI già attivo

La BaseService ha lo scopo di integrare Spring Boot nel contesto INFN

infn-microservices > BaseService



BaseService 

Project ID: 1169

 **119** Commit  **4** branch  **2** Tag  **1,3 MB** File

Micro Servizio base per lo sviluppo di backend all'interno

BACKEND: BASE-SERVICE-LIB

Al suo interno sono implementate tutte le funzioni ripetitive di cui ogni progetto ha bisogno per lavorare nell'ecosistema INFN

- Costruita in cima a Spring Boot
- Autenticazione con JWT (OAuth2)
- Autorizzazione dei Rest Controller tramite guardia basata su entitlement
- Standardizzazione delle chiamate ad altri microservizi tramite user-token o application-token
- Gestione nativa del tracing+logging distribuito (ispirato a zipkin)
- Caching delle autorizzazioni utente
- Background job service basato su Kafka
- Swagger/OpenAPI già configurato

```
it:
  infn:
    sisinfo:
      microservice:
        cache:
          enabled: false
          url-discovery-strategy: static
          url-discovery-map:
            godiva: ${GDV:https://godiva-test.dsi.infn.it}
        #queue
        queue-enabled: true
        queue-server-address: ${QUEUE_SERVER:localhost:9092}
        queue-client-id: application-management
        queue-group-id: application-management
        #queue job
        queue-job-processing: true
        queue-job-processing-concurrency: 1
        queue-job-processing-topic: application-management-job
      oidc:
        client:
          realm: test-realm
          auth-server: "https://${KK_HOST:localhost}:9090/auth"
          client-id: ${KK_CLIENT_ID:test-client}
          client-secret: ${KK_CLIENT_SECRET}
```

BACKEND: BASE-SERVICE-LIB

- Costruita in cima a Spring Boot
- Autenticazione con JWT (OAuth2)
- Autorizzazione dei Rest Controller tramite guardia basata su entitlement
- Standardizzazione delle chiamate ad altri microservizi tramite user-token o application-token
- Gestione nativa del tracing+logging distribuito (ispirato a zipkin)
- Caching delle autorizzazioni utente
- Background job service basato su Kafka
- Swagger/OpenAPI già configurato

Un middleware dedicato si occupa di verificare l'autorizzazione all'uso delle API, tramite l'inserimento di

```
@SecurityAccess(accessLevel = AccessLevelToken,  
                accessEntitlement = "app:app_create")  
@GetMapping(path = "/utils/verifyExist")  
@Operation(summary = "Check if the chosen name is already present",  
           responses = {  
               @ApiResponse(responseCode = "200",  
                             description = "Application ID or null"),  
           })  
public Boolean verifyExist(  
    @Parameter(description = "Is the app name", required = false)  
    @RequestParam("name") String name) {
```

BACKEND: BASE-SERVICE-LIB

- Costruita in cima a Spring Boot
- Autenticazione con JWT (OAuth2)
- Autorizzazione dei Rest Controller tramite guardia basata su entitlement
- Standardizzazione delle chiamate ad altri microservizi tramite user-token o application-token
- Gestione nativa del tracing+logging distribuito (ispirato a zipkin)
- Caching delle autorizzazioni utente
- Background job service basato su Kafka
- Swagger/OpenAPI già configurato

Le chiamate ad altri servizi possono avvenire sia tramite token utente che token applicativo. Anche questo aspetto è standardizzato in una funzione apposita

```
public interface ApiCommunication {  
    Enumeration for token type identification  
  
    public enum TokenType {  
        TokenTypeApplication,  
        TokenTypeUser  
    }  
  
    Call and http rest api using the service registry  
    Params:      method – is the http dialect used for call the api  
                tokenType – is the token that need to be used for authenticate the api  
                (Application or Logged user)  
                service – is the name of the service as registered in service registry  
                uri – is the api uri within the service  
                payload – is the payload to send to the api  
                retType – is the result type of the api(need to be mapped exactly with t  
                of the remote api)  
  
    Type parameters: <T> – the type of the class result of the oration  
    Returns:        The answer of the api  
  
    public <T> T callAPI(HttpMethod method,  
                        ApiCommunication.TokenType tokenType,  
                        String service,  
                        String uri,  
                        Object payload,  
                        ParameterizedTypeReference<T> retType,  
                        Object... uriVariables) throws Exception;
```

BACKEND: BASE-SERVICE-LIB

- Costruita in cima a Spring Boot
- Autenticazione con JWT (OAuth2)
- Autorizzazione dei Rest Controller tramite guardia basata su entitlement
- Standardizzazione delle chiamate ad altri microservizi tramite user-token o application-token
- **Gestione nativa del tracing+logging distribuito (ispirato a zipkin)**
- Caching delle autorizzazioni utente
- Background job service basato su Kafka
- Swagger/OpenAPI già configurato

Date che le interazioni fra i servizi potrebbero essere molte e complesse, anche asincrone, la libreria gestisce il tracing distribuito iniettando degli header nelle richieste http

Time	traceld	spanid	kubernetes.labels.app
▶ June 28th 2017, 14:05:00.983	f60dae70fc973940	f60dae70fc973940	graphql
▶ June 28th 2017, 14:05:00.981	f60dae70fc973940	c29cbb99dae2f20e	bookmark
▶ June 28th 2017, 14:05:00.972	f60dae70fc973940	f60dae70fc973940	graphql

BACKEND: BASE-SERVICE-LIB

Per velocizzare la scrittura di test, è stata introdotta la possibilità di simulare identità ed eventi sulle message-queue

- Mocked Identity Utility

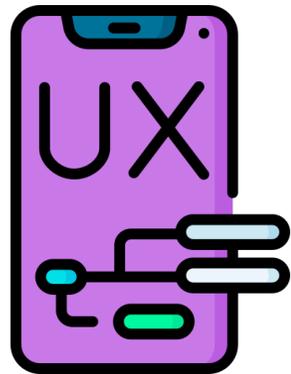
- Attivabile tramite profilo spring «mock-auth»
- Viene forgiato un token JWT all'avvio
- La UI può leggere la lista delle identità mocked tramite API

- Mocked Action Utility

- Attivabile tramite profilo spring «mock-action»
- Api rest per il fetch delle azioni disponibili
- Api rest per la sottomissione di un'azione via label

```
- infnUUID: MOCK_INFN_LNF_DIRECTOR
issuer: issuer
label: "Director LNF"
username: Inf_director
firstName: Alan
lastName: Parson
authentication:
  isMemberOf:
  schacUserStatus:
  eduPersonEntitlement:
    - urn:mace:inf.n.it:booking:entity@admin:/inf-n-lnf
    - urn:mace:inf.n.it:booking:workflow:approver@management:/inf-n-lnf
  edupersonassurance:
```

```
- label: Action One
type: job
data:
  id: id_q
  jobImplementation: it.inf.n.microservice.Class
  jobParameter:
    key_1: value_1
    key_2: value_2
```



Front-end

FRONTEND: INFN-APP-SHELL

- Libreria di componenti, guardie e servizi costruita su Angular
- Quickstart ridotto a una sola riga di shell
- Autenticazione con JWT (OAuth2) tramite guardia. Possibile la gestione di UI pubbliche
- Autorizzazione delle rotte tramite guardia role-based (isMemberOf oppure Entitlement)
- Standardizzazione l'aspetto dell'applicazione: temi, dark mode, alto contrasto, lingue, guide di on-boarding , utilities
- Gestisce nativamente il caricamento di config file dinamici (build-one / deploy-anywhere)
- Logging, server health, pannello assistenza

La AppShell è stata costruita con lo stesso scopo della BaseService

sysinfo_org > sw > Infn App Shell



Infn App Shell

Project ID: 3144

[Abbandona il progetto](#)

64 Commit

3 branch

8 Tag

2,5 MB File

FRONTEND: INFN-APP-SHELL

- Libreria di componenti, guardie e servizi costruita su Angular
- Quickstart ridotto a una sola riga di shell
- Autenticazione con JWT (OAuth2) tramite guardia. Possibile la gestione di UI pubbliche
- Autorizzazione delle rotte tramite guardia role-based (isMemberOf oppure Entitlement)
- Standardizzazione l'aspetto dell'applicazione: temi, dark mode, alto contrasto, lingue, guide di on-boarding , utilities
- Gestisce nativamente il caricamento di config file dinamici (build-one / deploy-anywhere)
- Logging, server health, pannello assistenza

```
<lib-infn-app-shell [appName]="appNameLg"  
  [appNameMd]="appNameMd" [appNameSm]="appNameSm"  
  [menuItems]="menuItems">  
  <router-outlet></router-outlet>  
</lib-infn-app-shell>
```

```
// shell & libs -----  
InfnAppShellModule.forRoot(environment),  
JoyrideModule.forRoot(),  
CoolStorageModule.forRoot(),  
OAuthModule.forRoot({  
  resourceServer: {  
    allowedUrls: ['http://localhost:8080/', 'https://'],  
    sendAccessToken: true  
  }  
}),  
FontAwesomeModule,
```

```
providers: [  
  {  
    provide: BASE_PATH,  
    useFactory: (config: AppEnvironmentService) => config.environment.API_BASE_PATH,  
    deps: [AppEnvironmentService],  
  },  
],
```

FRONTEND: INFN-APP-SHELL

- Libreria di componenti, guardie e servizi costruita su Angular
- Quickstart ridotto a una sola riga di shell
- Autenticazione con JWT (OAuth2) tramite guardia. Possibile la gestione di UI pubbliche
- Autorizzazione delle rotte tramite guardia role-based (isMemberOf oppure Entitlement)
- Standardizzazione l'aspetto dell'applicazione: temi, dark mode, alto contrasto, lingue, guide di on-boarding , utilities
- Gestisce nativamente il caricamento di config file dinamici (build-one / deploy-anywhere)
- Logging, server health, pannello assistenza

```
1 # modalità non interattiva
2 └─[$] ng new --collection @sysinfo/inf-n-app-shell \
3     --name ${PRJ_NAME} \ # nome del progetto (se omeo va
4     --root ${PRJ_DIR}   # directory del progetto (se omeo
5
6 # modalità interattiva
7 └─[$] ng new --collection @sysinfo/inf-n-app-shell
8 ? What is the name of your project?   Brand New Application
9 ? In which directory (if blank is used the project's name)?
10
11 # si può iniziare lo sviluppo
12 cd ${PRJ_DIR} && ng serve -o
13 code ${PRJ_DIR}
```

FRONTEND: INFN-APP-SHELL

- Libreria di componenti, guardie e servizi costruita su Angular
- Quickstart ridotto a una sola riga di shell
- Autenticazione con JWT (OAuth2) tramite guardia. Possibile la gestione di UI pubbliche
- Autorizzazione delle rotte tramite guardia role-based (isMemberOf oppure Entitlement)
- Standardizzazione l'aspetto dell'applicazione: temi, dark mode, alto contrasto, lingue, guide di on-boarding , utilities
- Gestisce nativamente il caricamento di config file dinamici (build-one / deploy-anywhere)
- Logging, server health, pannello assistenza

```
{  
  path: 'pub/app-guide',  
  component: AppComponent,  
},
```

```
{  
  path: 'priv/utente/inventari',  
  component: ListaInventariComponent,  
  canActivate: [MustBeAuth2Guard]  
},
```

```
{  
  path: 'priv/ufficio/inventari/:cdr',  
  component: ListaInventariUffComponent,  
  canActivate: [CheckRolesGuard, MustBeAuth2Guard],  
  data: {  
    roles: [  
      'i:inf:([a-zA-Z0-9]+):ebs:inf_resp_amm::resp_mans:',  
      'i:inf:([a-zA-Z0-9]+):ebs:ufficio_oc::resp_mans:'  
    ]  
  }  
},
```

Utilizza «authorization code flow»
più sicuro di «implicit flow» e
permette il renew in background

FRONTEND: INFN-APP-SHELL

- Standardizzazione l'aspetto dell'applicazione: temi, dark mode, alto contrasto, lingue, guide di on-boarding , utilities

Il layout dell'applicazione è un classico F-Shape con dei pannelli laterali aggiuntivi per le preferenze ed i diversi menù

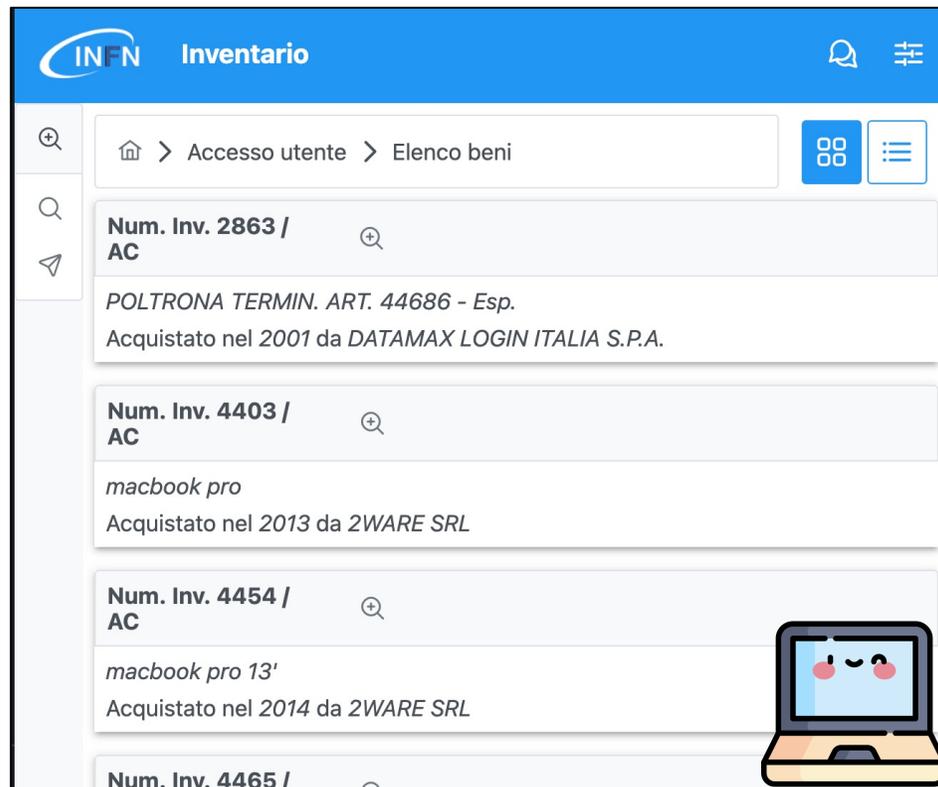
The screenshot displays the 'Inventario Self-Service' application interface. The main content area shows a list of assets with columns for inventory number, description, and acquisition details. A 'Preferenze' (Preferences) modal is open on the right, allowing users to adjust settings such as language (Italiano, English), theme (Scura, Normale), and visual theme (Default, Alto contrasto, Bootstrap). The application header includes the INFN logo and the title 'Inventario Self-Service'.

Num. Inv.	Descrizione	Acquistato da
2863 / AC	POLTRONA TERMIN. ART. 44686 - Esp.	DATAMAX LOGIN ITALIA S.P.A.
4454 / AC	macbook pro 13'	2WARE SRL
4620 / AC	APPLE iMAC 27" S/N SDGKXVHDDJ1GQ - IVA	C&C CONSULTING SPA

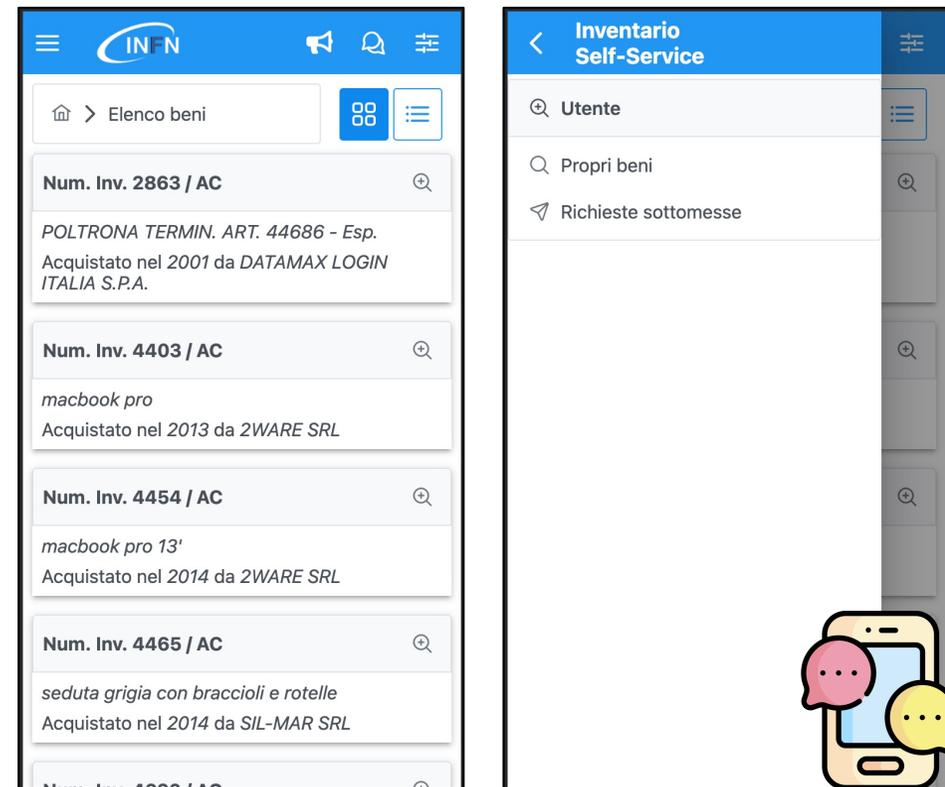
FRONTEND: INFN-APP-SHELL

- Standardizzazione l'aspetto dell'applicazione: temi, dark mode, alto contrasto, lingue, guide di on-boarding , utilities

Il Menù laterale viene compresso nei dispositivi con schermo medio e si apre al passaggio del mouse.

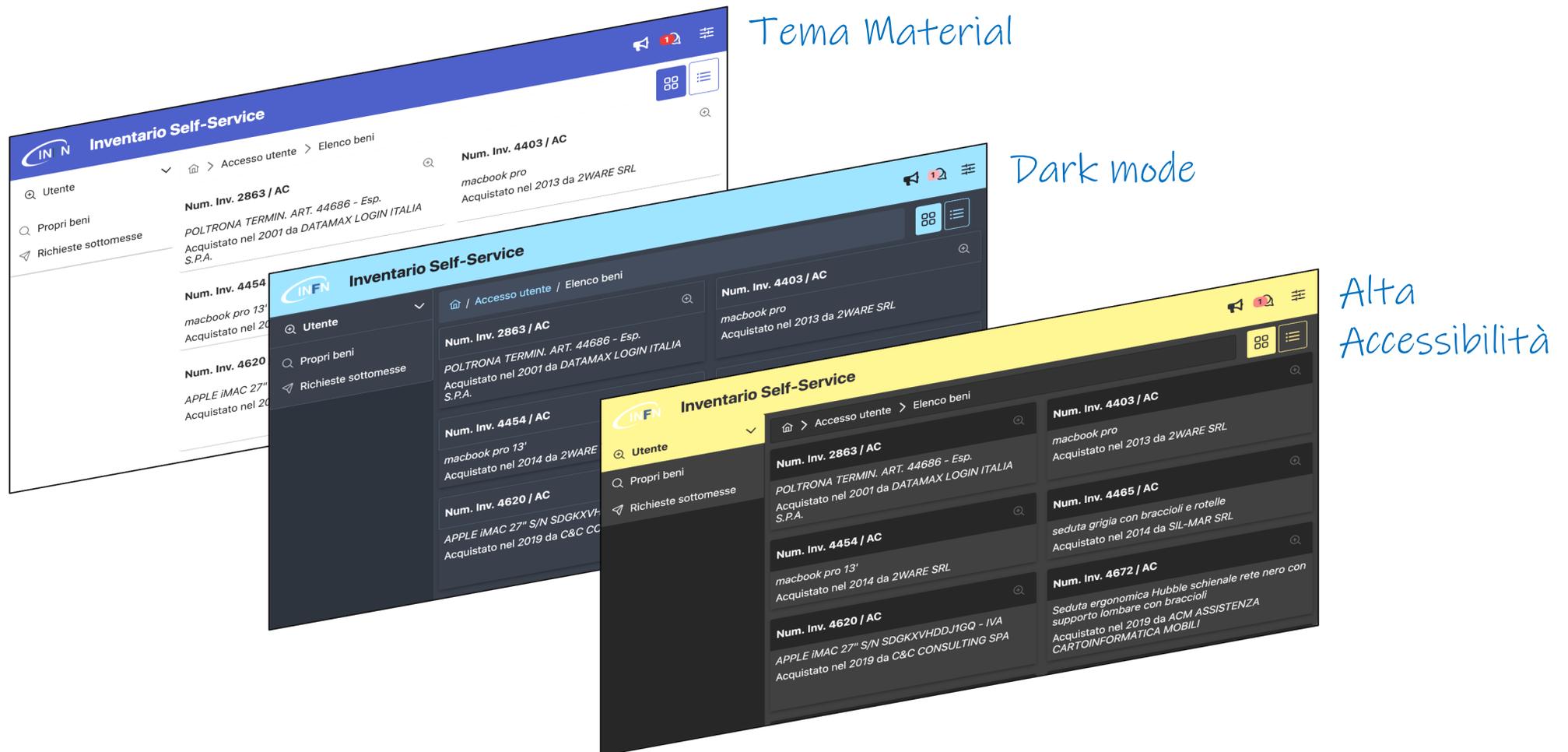


Sui dispositivi mobili è gestito con un pannello a scomparsa (hamburger menù)



FRONTEND: INFN-APP-SHELL

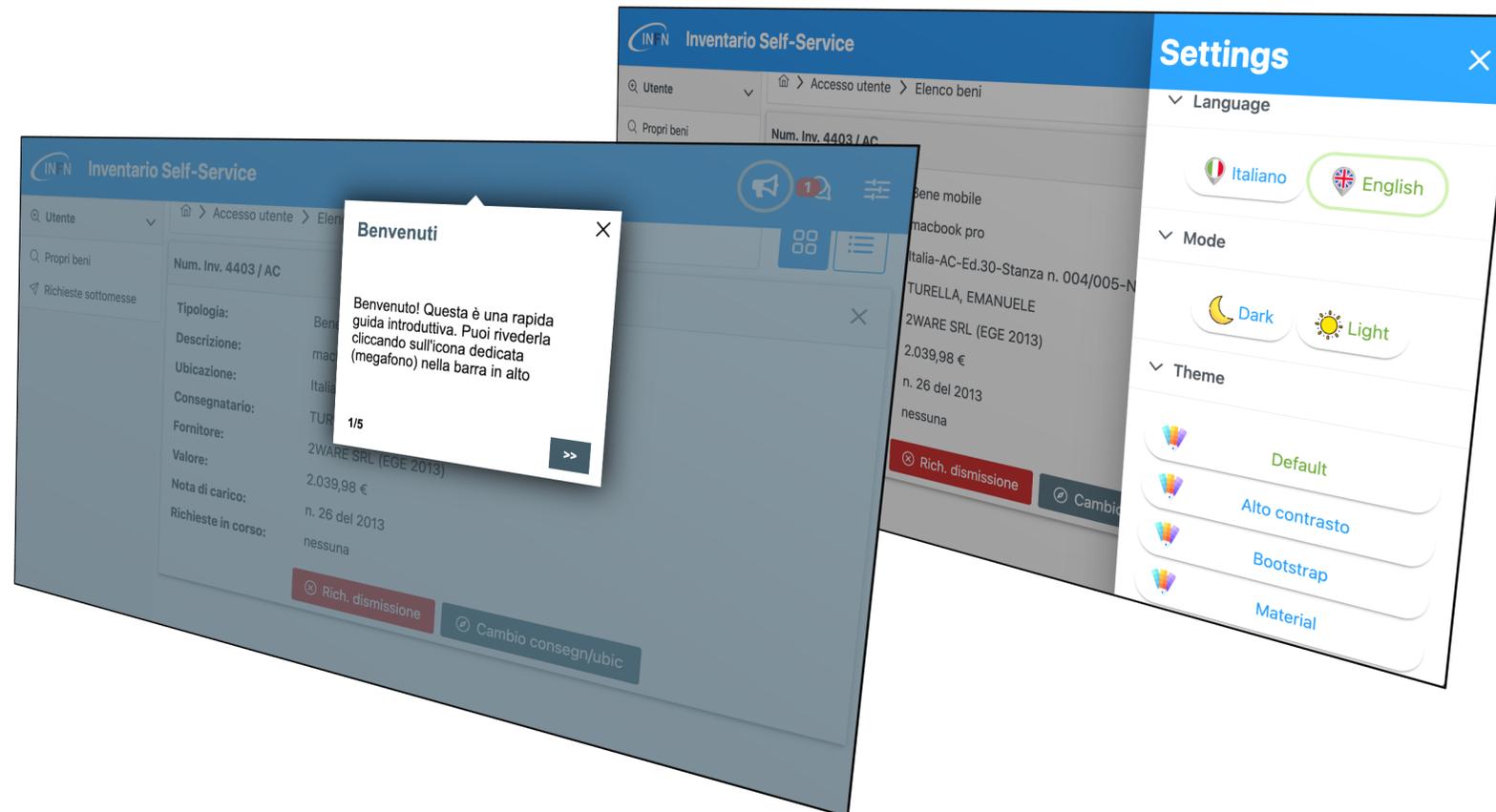
- Standardizzazione l'aspetto dell'applicazione: temi, dark mode, alto contrasto, lingue, guide di on-boarding, utilities



FRONTEND: INFN-APP-SHELL

- Standardizzazione l'aspetto dell'applicazione: temi, dark mode, alto contrasto, lingue, guide di on-boarding, utilities

Lo sviluppatore può configurare delle guide di on-boarding



Switch fra le lingue

FRONTEND: INFN-APP-SHELL

- Logging, server health, pannello assistenza

Un servizio centralizza la gestione dei log che possono essere consultati da una barra laterale

```
this.logger.showMessage({  
  severity: 'success',  
  summary: 'Recupero lavoro',  
  detail: 'Ripristinato il modulo non ancora salvato',  
  icon: 'pi pi-check-square',  
});
```



FRONTEND: INFN-APP-SHELL

- Logging, server health, pannello assistenza

Self-Service

Support requests can be submitted downloading the application LOG here and sending it via ticket

Download your LOG Open a new issue and attach your log

The dashboard collects all useful information for technical support

Opera (75.0.3969.250)	Mac (unknown)
Connected to the internet	INFN server is active
Version: v1.1.0 HEAD	Environment: PROD

Next PING to the server (15 / 20 seconds) Next HEALTH CHECK to the server (56 / 150 seconds)

ATTENTION: closing the browser these messages will be lost

Eventi di routing

```
{
  "ts": "2021-05-20T13:09:26.930Z",
  "livello": "INFO",
  "evento": "ROUTE",
  "id": "vzyCX",
  "message": "/priv/utente/inventari?idInv=430765",
  "options": {
    "url": "/priv/utente/inventari?idInv=430765",
    "params": {},
    "query": {},
    "data": {}
  }
},
```

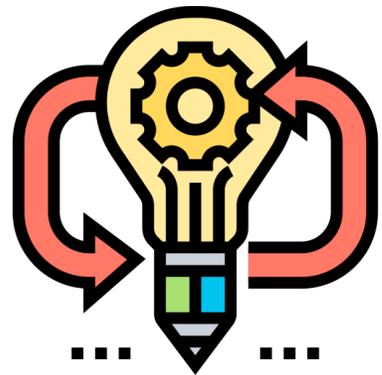
```
{
  "ts": "2021-05-20T13:09:26.939Z",
  "livello": "INFO",
  "evento": "CALL",
  "message": "",
  "id": "wxvCe",
  "options": {
    "method": "GET",
    "url": "https://inventario.dsi.infn.it/api/v1/uten
id_inv=430765&solo_bene=false"
  }
},
```

Request XHR

Response XHR

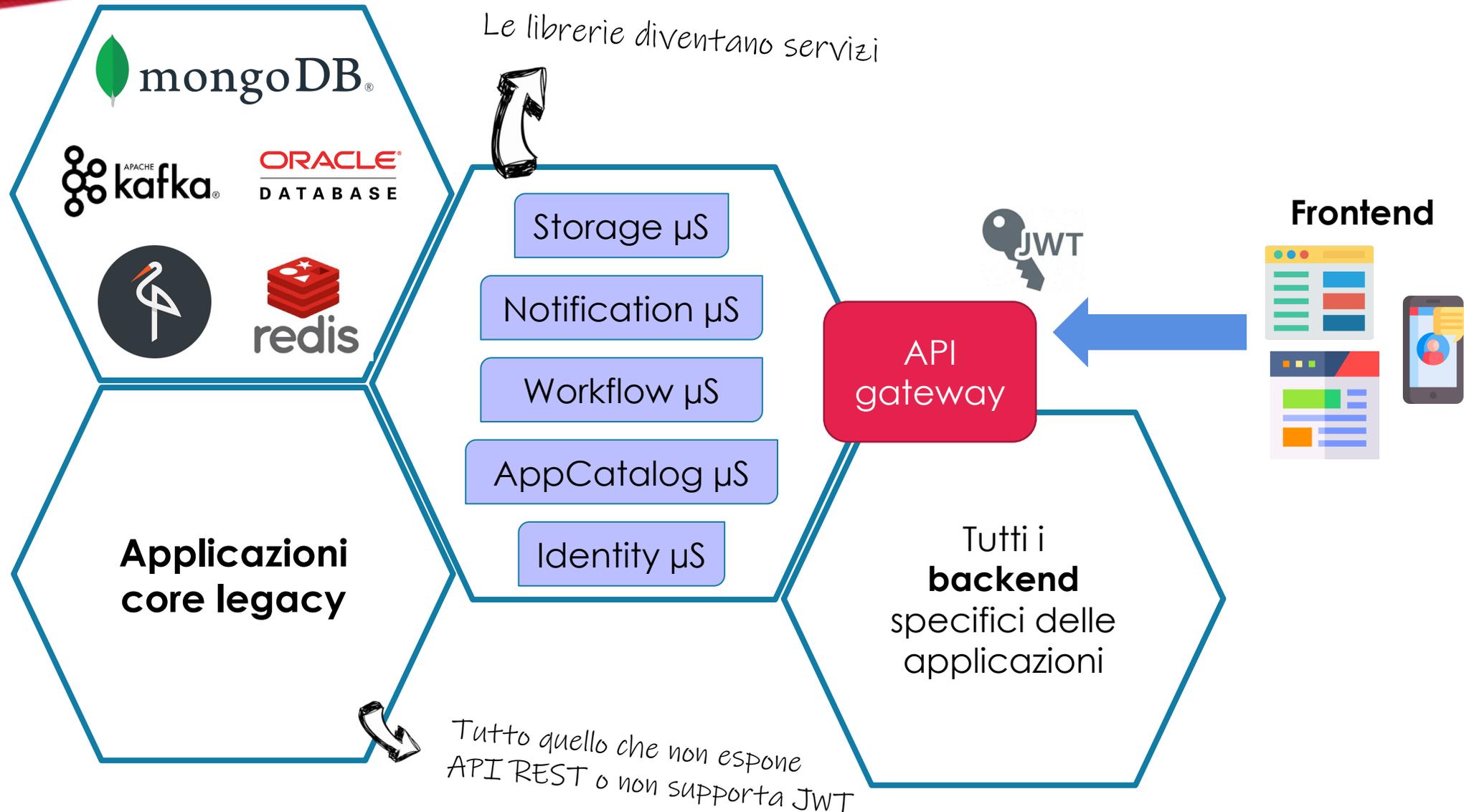
```
{
  "ts": "2021-05-20T13:09:27.338Z",
  "livello": "TRACE",
  "evento": "CALLOK",
  "message": "OK",
  "id": "06c9R",
  "options": {
    "status": "200",
    "data": {
      "bene": {
        "bene_id": 430765,
        "cdr": "AC",
        "tipo": "M",
        "num_inv": 4403,
        "ege": 2013,
        "descr": "macbook pro",
        "portatile": false,
        "valore_orig": 2039.98,
        "valore_real": 2039.98,
        "carico_num": 26,
        "carico_data": 2013,
        "discarico_tipo": null,
        "discarico_num": null,
        "discarico_ege": null,
        "fornitore_vid": null,
        "fornitore": "2WARE SRL",
        "consegn_pid": 9701,
        "consegn_nome": "TURELLA, EMANUELE",
        "ubicazione": "Italia-AC-Ed.30-Stanza",
        "ruolo": null,
        "valore_amm": null
      },
      "richieste": [],
      "autorizzazioni": {
        "id_inv": 430765,
        "perm_consubic": true,
        "perm_dismissione": true,
        "perm_portatile": true
      }
    }
  }
},
```

Il logging remoto è stato escluso sia per ragioni di performance sia per rispettare il GDPR

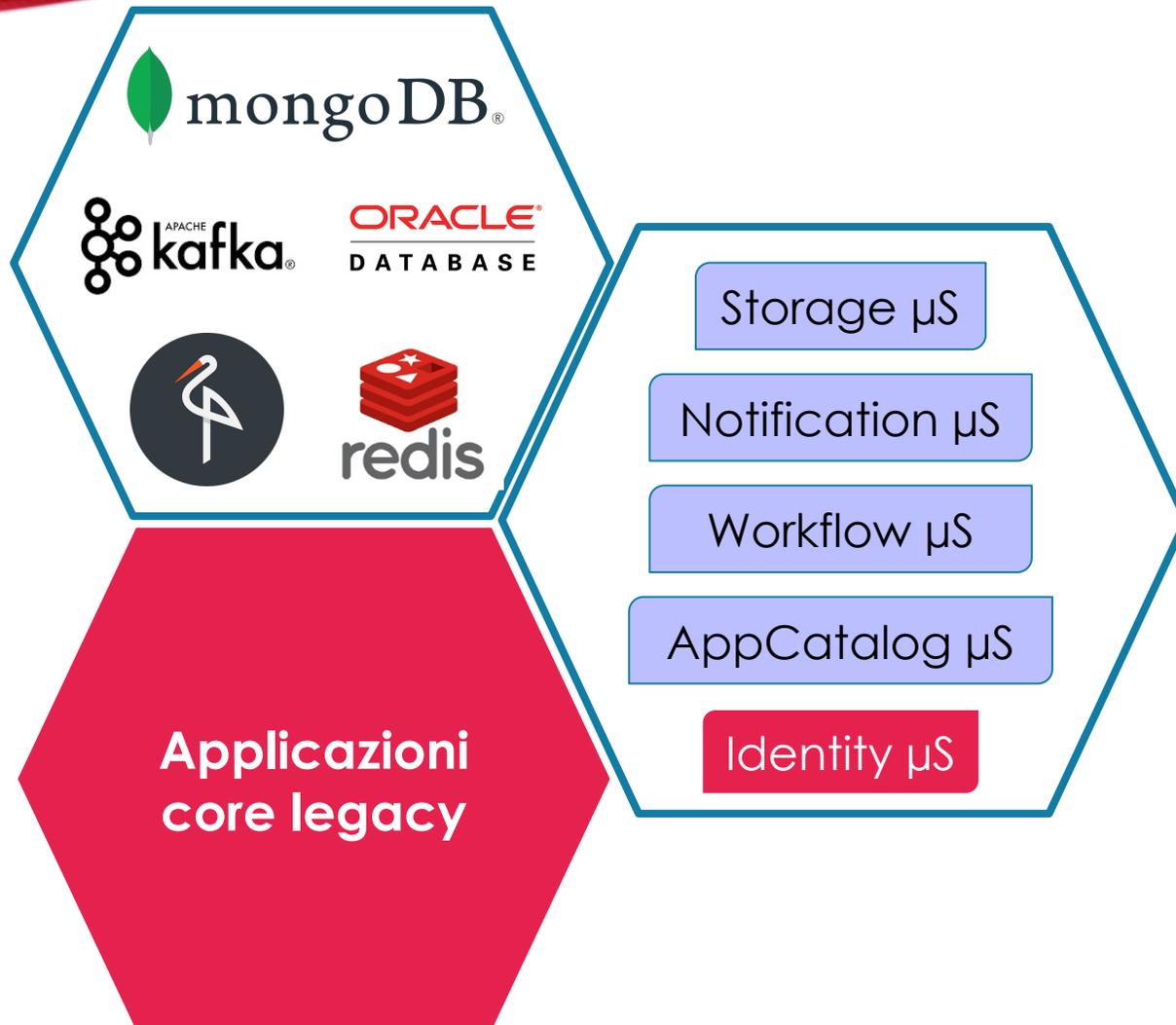


Servizi

SERVIZI RIUTILIZZABILI



SERVIZI RIUTILIZZABILI



Identity μS:

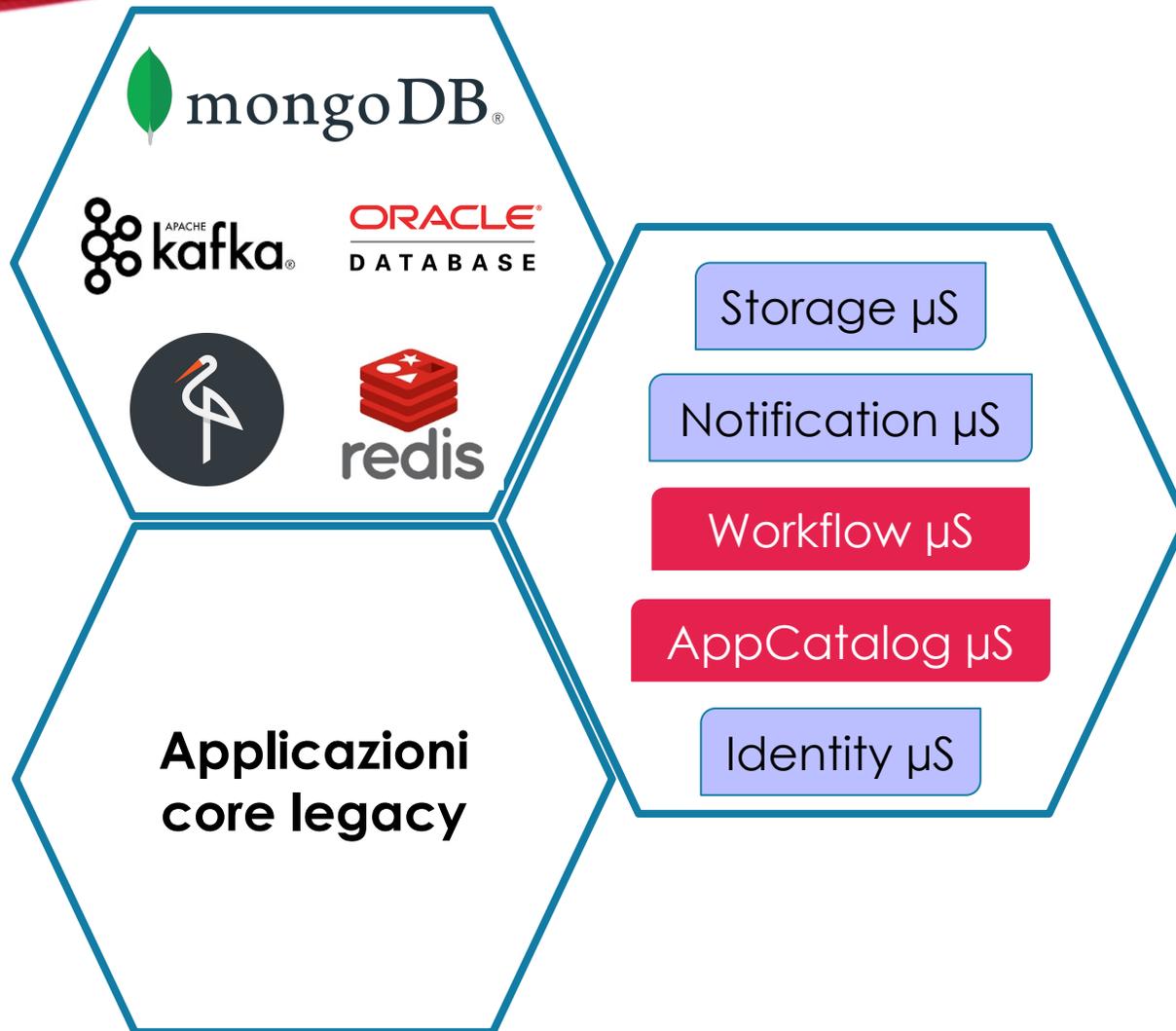
cache veloce delle identità e delle autorizzazioni. API complesse che incrociano identità, ruoli e domini.



Legacy wrapper μS:

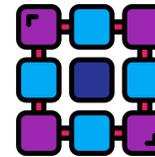
insieme di proxy REST verso sistemi core che espongono API in tecnologie diverse oppure non posso supportare alcune tecnologie

SERVIZI RIUTILIZZABILI



Workflow μ S:

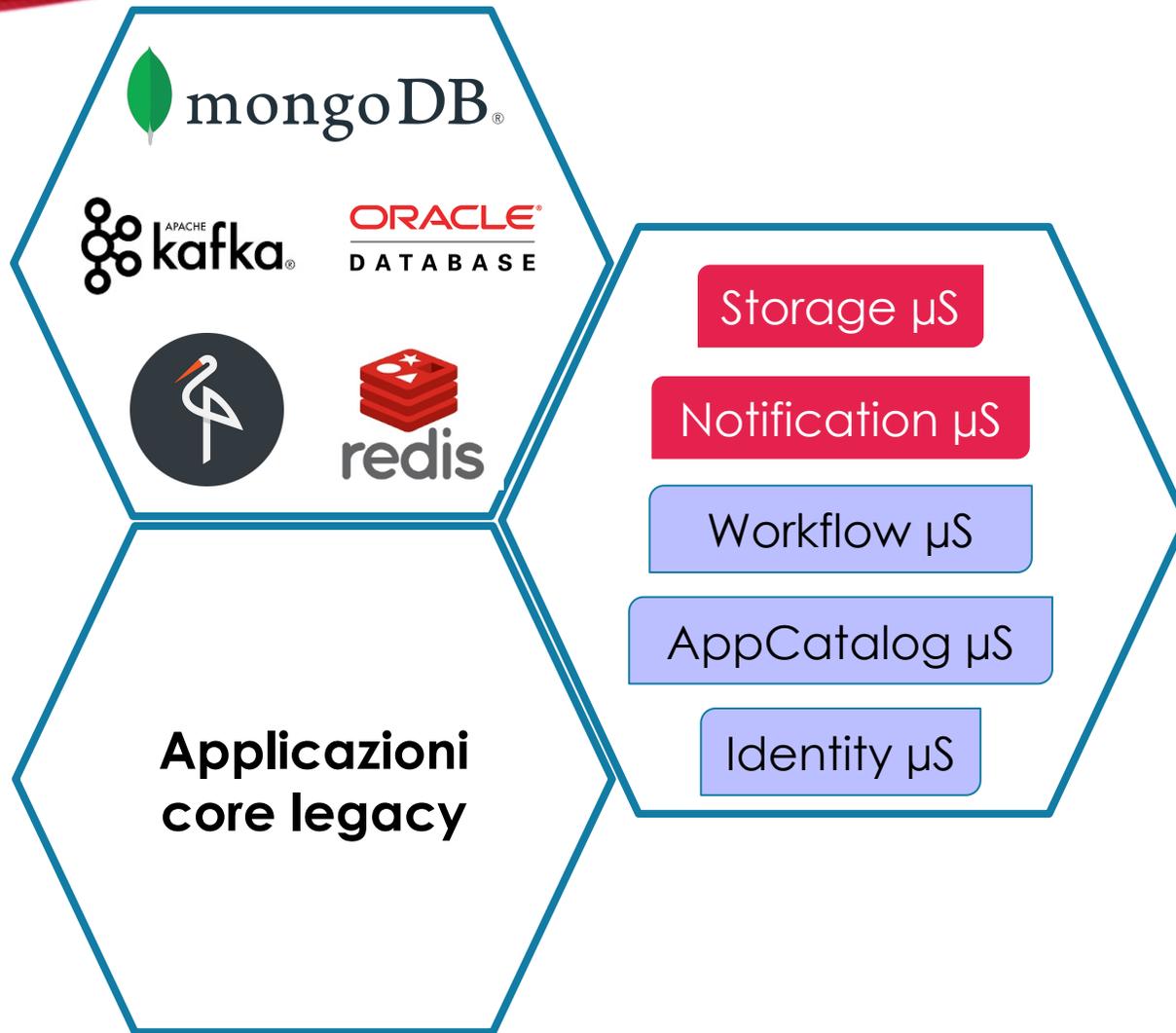
intento primario è avere una dashboard del backlog lavorativo personale (user-centric) e l'orchestrazione di processi semplici (es. prenotazione di impegno)



App.Catalog μ S:

portale INFN con gli steroidi: mapping isMemberOf -> entitlement; coordinamento di godiva e oidc; dashbord degli alert

SERVIZI RIUTILIZZABILI



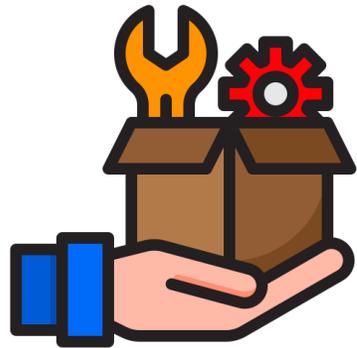
Storage μS:

servizio ad alte prestazioni per depositare file, catalogo delle tipologie di file, delle policy di conservazione, repository dei metadati per generare raccolte virtuali



Notification μS:

centro di ricezione delle notifiche applicative con la possibilità per l'utente di decidere quali topic seguire e con quali canali



Strumenti

STRUMENTI DI SVILUPPO





Discussione