

Documentazione collector site service

Collector site service Documentazione

Dipendenze software

- **Jquery** : `1.12.1` con Ajax per creare pagina dinamica e gestire le operazioni di tipo *POST* che necessitano del *BearerToken*.
 - **Jackson**: per gestire, codificare e decodificare i *JSON*.
 - **JWT**: per decodificare un *JSON Web Token* in un'oggetto di tipo `JavaScript`.
 - **Jersey**: consente di sviluppare servizi *RESTful* in Java implementando le specifiche *JAX-RS*.
-

Funzionalità realizzate

1. Login/logout con username e password ☒
 2. Elenco collezioni (personali) dell'utente (qui e successivamente con "utente" intendiamo quello autenticato) ☒
 3. Elenco collezioni condivise con un utente ☒
 4. Elenco dei dischi in una collezione (accessibile dall'utente) ☒
 5. Dettagli singolo disco contenuto in una collezione ☒
 6. Inserimento di un nuovo disco in una delle collezioni dell'utente ☒
 7. Ricerca di un disco (in base a vari criteri) tra le collezioni personali, tra le collezioni condivise o tra quelle pubbliche ☒
 8. Elenco di tutti gli autori presenti nel sistema ☒
 9. Elenco di tutti i dischi di un certo autore presenti nelle collezioni pubbliche ☒
 10. Aggiornamento di un disco in unacollezionepersonaledell'utente ☒
 11. Estrazione di alcune statistiche dalla collezione di un utente ☒
-

Servers

https://localhost:8080/

Authorize



Auth



POST /rest/auth/login do login



DELETE /rest/auth/logout do logout



Collections



GET /rest/collections/{COLLECTIONTYPE} get my collections



GET /rest/collections/{COLLECTIONTYPE}/{OUID} get shared collection with another one user



Records



GET /rest/collections/{COLLECTIONTYPE}/{COLLECTIONID}/records get collection records



GET /rest/collections/{COLLECTIONTYPE}/{COLLECTIONID}/records/{RECORDID} get collection record info



POST /rest/collections/{UID}/newrecord new record insert into user collection



GET /rest/collections/{COLLECTIONTYPE}/records get record by name, date, author



GET /rest/collections/{COLLECTIONTYPE}/records/{AUTHORNAME} get all author records



POST /rest/collections/{COLLECTIONTYPE}/{COLLECTIONID}/modifyrecord modify record into user collection



GET /rest/collections/{COLLECTIONTYPE}/{COLLECTIONID}/stats get stats for user collection



Authors



GET /rest/authors get all authors



Schemas



collectionSchema >



collectionListSchema >



trackSchema >



recordSchema >



recordListSchema >



recordCreatedSchema >



authorSchema >



authorsSchema >



statsSchema >



genreSchema >

