



UCSC



Informes y analíticas con

FOLIO

Mauricio Peñailillo Acuña

Jefe Unidad de Tecnologías de la Información -
SIBUCSC

ucsc.cl

Caso: Campos personalizados en reportes

Se requiere conocer un listado de préstamo de salas con agrupación por sede, carrera y otros criterios almacenados en campos de usuario personalizados, sin embargo, al momento de crear un listado de tipo Loans en FOLIO esta información no está disponible.

- Loan policy – Name
- Loan policy – UUID
- Material type – Name
- Patron group – Name
- User – Active
- User – Barcode
- User – Expiration date
- User – Last name, first name
- User – User UUID

Se verifica existencia de característica en modulo BulkEdit, sin embargo, no está construida para este proposito.

Matricula:2025 Jornada:Diurno Sede:Concepción UA:55	<input type="checkbox"/> Preferred contact type id
Matricula:2025 Jornada:Diurno Sede:Cañete UA:1255 S	<input type="checkbox"/> Link to the profile picture
Matricula:2025 Jornada:Diurno Sede:Talcahuano UA:12	<input type="checkbox"/> Date enrolled
Matricula:2025 Jornada:Diurno Sede:Concepción UA:59	<input type="checkbox"/> Expiration date
Matricula:2025 Jornada:Diurno Sede:Concepción UA:47	<input type="checkbox"/> Tags
Matricula:2025 Jornada:Diurno Sede:Concepción UA:72 Sexo:M Carrera:CONTADOR AUDITOR Facult	<input checked="" type="checkbox"/> Custom fields
	<input type="checkbox"/> Preferred email communications

Manos a la obra

Tenemos la información de préstamos con un listado en FOLIO, sin embargo, ¿cómo realizar un cruce con campos personalizados de usuarios que sea viable en términos de rendimiento y, en especial, replicable para otros casos futuros?

Propuesta: Almacenamiento local de información de usuarios en BD Postgres, con **LDLITE**, para posteriormente cruzar los resultados obtenidos desde una lista de préstamos.

Consideración importante: ¿Se pueden trabajar las listas a nivel de APIs?, respuesta: **¡SI!**

The screenshot displays a REST client interface with two main sections:

Table View:

Grid	123 _id	{ } jsonb
1	19,299	{"id": "3ce35094-dedb-487b-8666-40b9f37ea12a", "meta": {"creation_date": "2022-11-05", "last_login_date": ""}, "type": "object", "active": false, "barcode": "186849450", "metadata": {"createdDate": "2023-01-27T22:41:06.14"}}
2	19,300	{"id": "3ce39117-3f0d-468e-8d95-a429e8dcc6e7", "meta": {"creation_date": "2022-11-05", "last_login_date": ""}, "type": "object", "active": false, "barcode": "173966342", "metadata": {"createdDate": "2023-01-27T22:33:43.86"}}
3	19,301	{"id": "3ce3ee5e-1e58-4b1f-894d-d9084ae0aff4", "meta": {"creation_date": "2022-11-05", "last_login_date": ""}, "type": "object", "active": false, "barcode": "166688183", "metadata": {"createdDate": "2023-01-27T22:29:43.38"}}

REST Client View:

Method: GET
URL: <https://okapi-ucsc.folio.ebsco.com/lists/0f179a55-7411-4be2-8406-a185d34b0fec/contents>
Status: 200 OK
Time: 1.53s
Size: 385.44KB

Response Body:

```
1 {"content":[{"holdings.version":"5","cispi.discovery_display_name":"Meson de atencion - Seccion Reserva Biblioteca Central","instance.created_at":"2023-01-27T21:55:01.992+00:00","loans.action":"checkedin","holdings.administrative_notes":["Identifier(s) from previous system: 000041999BSACSAL"],"loans.due_date":"2025-07-22T00:55:46.469+00:00","instance.staff_suppress":"false","instance.format_names":["unspecified -- unspecified"],"holdings.hrid":"ho00000052296","instance.instance_type_name":"other","holdings.notes":[""],"items.electronic_access":[""],"holdings.created_at":"2023-02-24T23:05:46.841+00:00","holdings.id":"ab5d8c39-a545-5eaf-8e82-8505"}
```

folio 10.1.0.250:5432

Databases

folio

Schemas

cabid

folio_authority

folio_circulation

folio_inventory

folio_users

Tables

users 104M

Foreign Tables

Comment:

Has Row-Level Security Partitions

Partition by:

- Columns
- Constraints
- Foreign Keys
- Indexes
- Dependencies
- References
- Partitions
- Triggers

Column Name	#	Data type	Identity	Collation	Not Null	Default	Comment
123__id	1	int4			[]		
{ }jsonb	2	jsonb			[]		

```
10 ld = ldlite.LDLite()
11 # Change the tenant, user and password here. User must be in User APP and have permissions to read required data:
12 ld.connect_okapi(url=OKAPI_URL, tenant=OKAPI_TENANT, user=OKAPI_USER, password=OKAPI_PASSWD)
13
14 load_dotenv()
15 db = ld.connect_db_postgresql(dsn='dbname={d} host=localhost user={u} password={p}'.format(d=os.getenv('DBNAME'), u=os.getenv('DBUSER'), p=os.getenv('DBPASS')))
16 #db = ld._connect_db_duckdb(filename="duckdb_folio.db")
17 allrec = 'cql.allRecords=1 sortby id'
18 bib_allrec = {'recordType': 'MARC_BIB', 'state': 'ACTUAL'}
19 auth_allrec = {'recordType': 'MARC_AUTHORITY', 'state': 'ACTUAL'}
20 loans_allrec = {'status.name': 'open'}
21
22 #ESTO NO FUNCIONA
23 lists_allrec = 'fields = cospi.name,instance.title,items.effective_call_number,loans.checkout_date,loans.due_date,loans.renewal_count,loans.return_date,lpoli
24 You, 5 months ago • General changes ...
25 queries = [
26 ('folio_users.users', '/users', allrec, 0),
```

Interacción con Django Rest Framework

```
You, 3 months ago | 1 author (You)
95 class FOLIOUser(models.Model):
96     id = models.IntegerField(primary_key=True, db_column='__id', blank=True, null=True)
97     jsonb = models.JSONField(blank=True, null=True)
98
You, 3 months ago | 1 author (You)
99 class Meta:
100     managed = False
101     db_table = 'users'
102     app_label = 'folio'
103
104     @property
105     def custom_fields(self):
106         return self.jsonb['customFields']
107
```

```

456 #LISTADO PRESTAMOS DE SALA SEMESTRE 2025-1 (TEST2)
457 @api_view(['GET'])
458 @permission_classes((IsAuthenticatedOrReadOnly,))
459 def loanlist_folio_v2(request):
460     result = {'total':0,'content':[]}
461     fields = 'cospi.name,instance.title,items.effective_call_number,loans.checkout_date,loans.due_date,loans.renewal_count,loans.return_date,lpolicy.name,mtypes.name,g
462     response = requests.get(settings.OKAPI_URL+"/lists/f65649d6-5525-4986-a066-ad0a227d7cd2/contents?fields=loans.id&size=1", headers = settings.OKAPI_HEADER)
463     prestamos = response.json()
464     result['total'] = prestamos['totalRecords']
465
466     response = requests.get(settings.OKAPI_URL+"/lists/f65649d6-5525-4986-a066-ad0a227d7cd2/contents?fields={f}&offset=0&size={t}&".format(f=fields,t=result['total']),
467     for loan in response['content']:
468         #IMPORTANTE, LA HORA SERA AJUSTADA A GMT-4 PERO POSTERIOR A CAMBIO DE HORARIO EL 5 DE SEPTIEMBRE DEBE SER A GMT-3
469         loan['loans.checkout_date_gmt3'] = datetime.strptime(loan['loans.checkout_date'], "%Y-%m-%dT%H:%M:%S.%fZ")-timedelta(hours=4)
470         loan['loans.return_date_gmt3'] = datetime.strptime(loan['loans.return_date'], "%Y-%m-%dT%H:%M:%S.%fZ")-timedelta(hours=4)
471         #DURACION EN MINUTOS, REDONDEADA HACIA ARRIBA
472         loan['loans.duration'] = math.ceil(((loan['loans.return_date_gmt3'] - loan['loans.checkout_date_gmt3']).total_seconds())/60)
473
474     user_list = list(LoanUser.objects.all().values("jsonb__barcode","jsonb__customFields"))
475     user_df = pl.DataFrame(user_list)
476     loan_df = pl.DataFrame(data=response['content'])
477     merged_df = loan_df.join(user_df, left_on='users.barcode', right_on='jsonb__barcode',how='inner')
478     flat_merged_df = merged_df.unnest("jsonb__customFields")
479     result['content'] = flat_merged_df.to_dicts()
480     | You, 3 months ago • Optimizacion listado prestamos
481     return Response(result)
482

```

Polars, Streamlit y Plotly: Visualización

```
import polars as pl
import streamlit as st
import plotly.express as px
import requests
from sys import argv
from lxml import etree
from xlwt import Workbook
from streamlit_dynamic_filters import DynamicFilters

json_schema = pl.Struct([
    pl.Field("items.status_name", pl.Utf8),
    pl.Field("cospi.name", pl.Utf8),
    pl.Field("groups.group", pl.Utf8),
    pl.Field("lpolicy.name", pl.Utf8),
    pl.Field("loans.return_date", pl.Utf8),
    pl.Field("loans.due_date", pl.Utf8),
    pl.Field("instance.title", pl.Utf8),
    pl.Field("users.barcode", pl.Utf8),
    pl.Field("items.effective_call_number", pl.Utf8),
    pl.Field("users.active", pl.Utf8),
    pl.Field("mtypes.name", pl.Utf8),
    pl.Field("users.last_name_first_name", pl.Utf8),
    pl.Field("loans.id", pl.Utf8),
    pl.Field("loans.checkout_date", pl.Utf8),
    pl.Field("loans.checkout_date_gmt3", pl.Utf8),
    pl.Field("loans.return_date_gmt3", pl.Utf8),
    pl.Field("loans.duration", pl.Utf8),
    pl.Field("ua", pl.Utf8),
    pl.Field("sede", pl.Utf8),
    pl.Field("sexo", pl.Utf8),
    pl.Field("carrera", pl.Utf8),
    pl.Field("ingreso", pl.Utf8),
    pl.Field("jornada", pl.Utf8),
    pl.Field("facultad", pl.Utf8),
    pl.Field("contractual", pl.Utf8),
])
```

```
response = requests.get("http://api.sibucsc.cl/common/folio/uti/loans/test2").json()['content']
df = pl.DataFrame(data=response, schema=json_schema)

df_sede_sexo = df.group_by(['sexo', 'sede']).len(name="n")
df_grupo_fac = df.group_by(['groups.group', 'facultad']).len(name="n")
#df_sede_only = df.group_by(['sede']).agg(pl.col("Count").sum())
#with Workbook('output.xlsx') as wb:
#    df.write_excel(workbook=wb, worksheet="Data", table_name="loan_data")
st.set_page_config(layout="wide")
st.write("Resulting DataFrame for Loan List")
st.dataframe(data=df)

st.write("Prestamos agrupados por sede/sexo")
st.dataframe(data=df_sede_sexo)
fig_sed = px.bar(df_sede_sexo['sede', 'sexo', 'n'], x="sede", y="n", color="sexo", barmode="group", text="n")
st.plotly_chart(fig_sed)

st.write("Prestamos agrupados por grupo/facultad")
st.dataframe(data=df_grupo_fac)
fig_sed = px.bar(df_grupo_fac['groups.group', 'facultad', 'n'], x="groups.group", y="n", color="facultad", barmode="group", text="n")
st.plotly_chart(fig_sed)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
o (oa_env) bocky@MinisBocky:~/Desktop/Development/OA$ streamlit run ./script_request_data.py
```

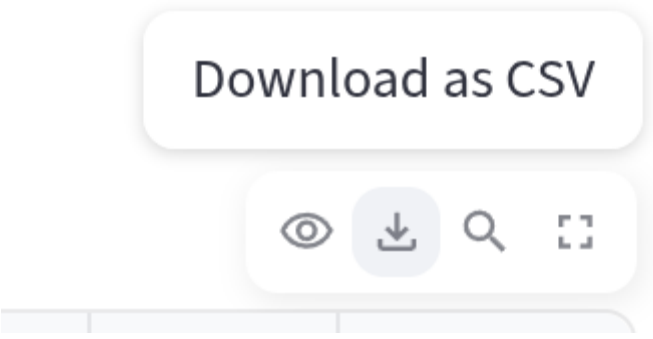
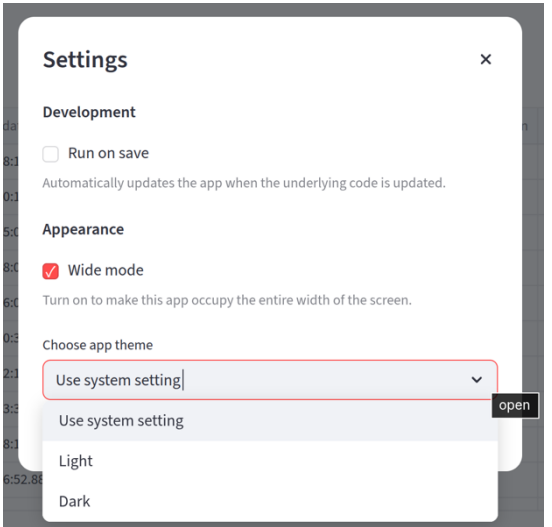
You can now view your Streamlit app in your browser.

Local URL: <http://localhost:8501>

Network URL: <http://192.168.4.107:8501>

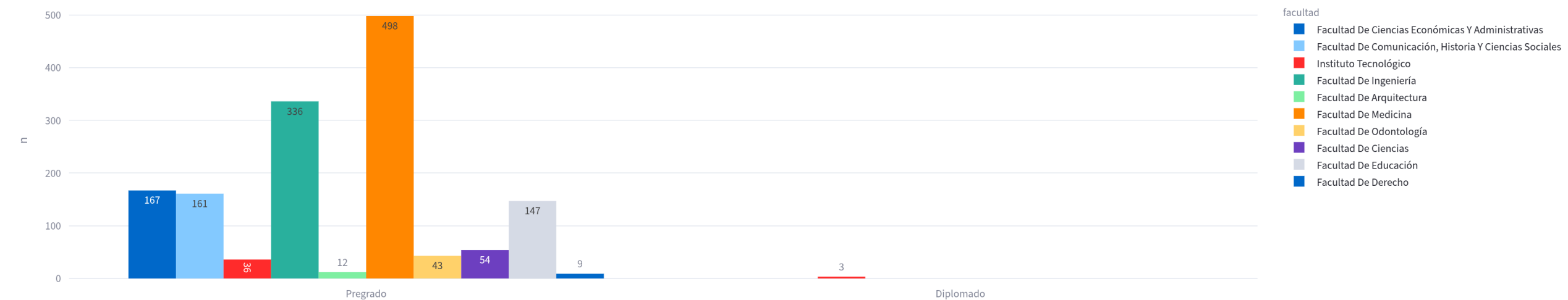
Resulting DataFrame for Loan List

_name	loans.id	loans.checkout_date	loans.checkout_date_gmt3	loans.return_date_gmt3	loans.duration	ua	sede	sexo	carrera	ingreso	jornada	facultad
GUADALUPE DEL CARMEN	1e3fb0de-0935-47c9-a259-a8c84aa92788	2025-06-25T22:38:13.588Z	2025-06-25T18:38:13.588000	2025-06-25T19:03:08.908000	25	62	Concepción	F	PEDAGOGÍA EN EDUCACIÓN MEDIA E	2025	Diurno	Facultad De Educación
ANTONELLA FERNANDA	591cd081-5d90-41ef-b8f5-fe2e2476e82b	2025-04-25T16:50:18.230Z	2025-04-25T12:50:18.230000	2025-04-25T13:51:59.945000	62	48	Concepción	F	LICENCIATURA EN ENFERMERÍA	2025	Diurno	Facultad De Medicina
ANTONELLA FERNANDA	a81b4e5c-ca52-43f8-b347-95653a3d7523	2025-05-22T17:45:00.045Z	2025-05-22T13:45:00.045000	2025-05-22T15:17:49.284000	93	48	Concepción	F	LICENCIATURA EN ENFERMERÍA	2025	Diurno	Facultad De Medicina
JORGE ANTONIO	683bf305-3094-4251-837e-863a794704ef	2025-04-23T13:58:01.321Z	2025-04-23T09:58:01.321000	2025-04-23T12:40:57.033000	163	53	Concepción	M	INGENIERÍA CIVIL INFORMÁTICA	2025	Diurno	Facultad De Ingeniería
JADE PATRICIA	d4cf96d8-7b44-4bab-b908-6c01d79890f1	2025-05-05T20:06:05.290Z	2025-05-05T16:06:05.290000	2025-05-05T16:10:20.409000	5	359	Concepción	F	ARQUITECTURA	2025	Diurno	Facultad De Arquitectura
NAHIS VALENTINA	7bf35de8-7d34-4caf-b7e0-537c3a198fa2	2025-04-14T18:00:32.699Z	2025-04-14T14:00:32.699000	2025-04-14T16:39:49.893000	160	47	Concepción	F	PSICOLOGÍA	2025	Diurno	Facultad De Comunicació
O, KARIM ESTEFANO	90c615e9-8fa9-495a-beb1-7fe219ace8a4	2025-04-08T15:02:16.734Z	2025-04-08T11:02:16.734000	2025-04-08T11:32:24.209000	31	124	Concepción	M	PEDAGOGIA EN EDUCACION FISICA	2024	Diurno	Facultad De Educación
O, KARIM ESTEFANO	a4b82fd2-db80-4692-aae3-9eb6711f7772	2025-07-01T13:23:34.508Z	2025-07-01T09:23:34.508000	2025-07-01T11:46:08.275000	143	124	Concepción	M	PEDAGOGIA EN EDUCACION FISICA	2024	Diurno	Facultad De Educación
O, KARIM ESTEFANO	bf2f29e7-8417-451e-bf77-690b182ae412	2025-04-22T17:28:15.105Z	2025-04-22T13:28:15.105000	2025-04-22T15:03:09.800000	95	124	Concepción	M	PEDAGOGIA EN EDUCACION FISICA	2024	Diurno	Facultad De Educación
, VALENTINA SOFÍA	6cad2e1c-9cf4-4d47-aaa8-7258c5593aec	2025-04-02T15:56:52.880Z	2025-04-02T11:56:52.880000	2025-04-02T13:59:32.419000	123	359	Concepción	F	ARQUITECTURA	2025	Diurno	Facultad De Arquitectura



Prestamos agrupados por grupo/facultad

groups.group	facultad	n
Pregrado	Facultad De Ciencias Económicas Y Administrativas	167
Pregrado	Facultad De Comunicación, Historia Y Ciencias Sociales	161
Pregrado	Instituto Tecnológico	36
Docentes Part-time	None	3
Pregrado	Facultad De Ingeniería	336
Pregrado	Facultad De Arquitectura	12
Pregrado	Facultad De Medicina	498
Pregrado	Facultad De Odontología	43
Pregrado	Facultad De Ciencias	54
Staff	None	5



Aprendizaje y consideraciones importantes

- No todo es traspasable a LDLITE, adicionalmente hay consideraciones respecto al trabajo con datos anidados dentro de un JSON, especialmente listados a mayor profundidad.
- El uso de Frameworks, JSONPath e Índices ayuda, pero se debe evaluar caso a caso.
- El uso de índices para atributos al interior del campo JSON es limitado, ya que se pierde al cargar nuevamente la tabla.
- El correcto uso de Pandas o Polars en Python puede reducir el tiempo de procesamiento de reportes más grandes, y facilitar la exportación inmediata a csv/xlsx

Proximamente: pruebas con **Apache Superset + Shillelagh + LDLITE/API**



UCSC