

## GUIDELINES

- 1. TITLE (Raleway, All Capital Letters, Bold, left centered and Size 14)**
2. Subtitles (Raleway, left centered, and Size 12)
3. Body (Lato and Size 11)
4. Bullet points:
  - ...
  - ...
5. 2 empty rows before a Title and one empty row after (row = 11 size)
6. No special line spacing between paragraphs, only separated by a row (size 11)

## Activity: to build a clean interface to fetch and visualize documents

The purpose of the activity is to create a clean interface where you can visualize the documents that are available on the API. You will receive a credential to access the API, and the documentation to interact with. There is no need to create a stunning web interface, but a clean and organized one. Extra points are rewarded if any CSS framework is used.

### Details of the activity

The deadline is 2023-10-18. The mini application must meet the following requirements:

Requirement classes:

- Required: the requirement is mandatory and must be met.
- Extra point: the requirement is optional and could suppose extra effort, however it is a nice to have.

Requirement Types:

- Non-functional: Constraints that the system must meet and be in accordance with.
- Functional: Specific business requirements

ID	Class	Type	Description
1	Required	Non-functional	The solution must use the Vue.js ecosystem.
2	Extra point	Non-functional	The solution should use TypeScript in the proper way.
3	Required	Non-functional	The solution must be of the type Single Page Application (SPA)
3.1	Required	Non-functional	The solution <b>must not</b> use JSX
4	Extra point	Non-functional	The solution should use state management (Pinia) for saving previously loaded data
5	Extra point	Non-functional	The solution should have at least one unit test for every route
6	Required	Non-functional	The solution's code must be cleaned and ordered with Prettier
7	Required	Non-functional	The solution's code must be clear of linting errors. Use ESLint for this task.
8	Required	Functional	A user shall be able to log-in using an e-mail and a password
9	Required	Functional	A user shall be able to see the list of documents provided by the API
10	Extra point	Functional	A user shall be able to filter documents by a Legal Entity Identifier (LEI)
11	Required	Functional	The system shall implement an option to implement a pagination system. It must show the total number of elements. If a traditional pagination system, it must show the number of the current page, and the total of the pages. If an <a href="#">autoscroll</a> is implemented, then only the total number of

			elements is required. It must be always visible (eg. sticky nav or headers, etc.. it is up to you)
12		Functional	A user shall be able to see the owner's name of the document
13	Required	Functional	The system shall render via iframe the contents of the document, which is in all the cases a PDF format.
15	Required	Functional	The user shall be able to logout
16	Required	Functional	The system shall show the history of documents that the user opened in the current session
17	Required	Functional	The system shall show the names of the people the document is shared with
18	Required	Functional	The system shall intercept http errors and show a generic error message when it happens. The system must not be halted, unless there is an error with the JWT token.
19	Required	Non-functional	The system shall be fluid. No screen frozen when loading or retrieving data from the API. Instead, it shall show a loading wheel or a preloader indicator.

## API access

The API is a REST API. It means, that it offers CRUD-based operations given the following HTTP verbs:

- GET: Read data
- POST: Create data
- PUT: Update data
- DELETE: delete data

URL	<a href="https://api-uat.researchpool.com">https://api-uat.researchpool.com</a>
Username	<a href="mailto:lucie.paper@yopmail.com">lucie.paper@yopmail.com</a>
Password	Researchpool22!
Documentation to login	<a href="https://api-uat.researchpool.com/docs?ui=re_doc#section/Authentication">https://api-uat.researchpool.com/docs?ui=re_doc#section/Authentication</a>

## Note on requirements 2 – 7

When creating a Vue.js project, a list of options are asked. Most of the requisites could be implemented by answering "Yes". A little remark for JSX support: you must answer "No".

## Note on requirement 10

The document could be linked to a Legal Identifier. This is a financial term to define companies that issue stock in the stock exchanges. The most big companies have an LEI. For example, Apples' is HWUPKR0MPOU8FGXBT394. This code may be used to filter documents in the API. There are four query parameters to be performed in order to get the documents belonging to a company:

- cases.issuers.lei
- cases.stocks.issuer.lei
- issuers.lei

- stocks.issuer.lei

If you opt for meeting this requirement, you must do four GET requests, each one with a different query, and then you must merge the four results:

Result1 = /documents?cases.issuers.lei=HWUPKR0MPOU8FGXBT394

Result2 = /documents?cases.stocks.issuer.lei= HWUPKR0MPOU8FGXBT394

Result3 = /documents?issuers.lei=HWUPKR0MPOU8FGXBT394

Result4 = /documents?stocks.issuer.lei=HWUPKR0MPOU8FGXBT394

FullResult = Result1 + Result2 + Result3 + Result4

## Note on requirement 11

The API is able to retrieve data in application/ld+json format. It gives you all the information you need to create the pagination system.

Example:

```
GET https://api-uat.researchpool.com/legal_entities
Accept: application/ld+json
Authorization: Bearer ...
```

Will give you this response:

```
{
  "@context": "\contexts\LegalEntity",
  "@id": "\legal_entities",
  "@type": "hydra:Collection",
  "hydra:totalItems": 2394586,
  "hydra:member": [
    {
      "@id": "\legal_entities\391200XDEE3AGQOY5389",
      "@type": "LegalEntity",
      "lei": "391200XDEE3AGQOY5389",
      "legalName": "!Wir: Kommunikation und Unternehmensberatung GmbH.",
      "stocks": []
    },
    ...
  ],
  "hydra:view": {
    "@id": "\legal_entities?page=1",
    "@type": "hydra:PartialCollectionView",
    "hydra:first": "\legal_entities?page=1",
    "hydra:last": "\legal_entities?page=79820",
    "hydra:next": "\legal_entities?page=2"
  },
  ...
}
```

The total items count is given by the property "hydra:totalItems"

The pagination information is given by the object "hydra:view".

## Note on requirement 12 and 17

The API works with IRI references. An IRI reference is a URI that can be retrieved from the API in the same way as the initial request. This is the case of the owner.

Let's suppose you retrieve a document, and this is the result:

```
{
  "@context": "\contexts\Document",
  "@id": "\documents\ee82a28e-fb19-445c-b292-f52621829567",
  "@type": "Document",
  "uuid": "ee82a28e-fb19-445c-b292-f52621829567",
  "name": "This document belongs to CARBOIL S.R.L",
  "owner": "\users\8agP4nxN4r",
  "authors": [],
  "creationDate": "2023-10-11T16:30:43+02:00",
  "cases": [],
  "metadata": [],
  "sharedWith": [],
  "stocks": [],
  "issuers": [
    "\legal_entities\815600B0F82B04759F52"
  ]
}
```

The property "owner" gives you the owner IRI. In order to get the name, you must retrieve it from the API:

### GET API\_URL + IRI

In this case:

```
GET https://api-uat.researchpool.com/users/8agP4nxN4r
Accept: application/ld+json
Authorization: Bearer ...
```

Gives you:

```
{
  "@context": "\contexts\User",
  "@id": "\users\8agP4nxN4r",
  "@type": "User",
  "userid": "8agP4nxN4r",
  "name": "lucie",
  "email": "lucie.paper@yopmail.com",
  "surname": "paper"
}
```

Then, you can put the name and surname as the name of the owner.

## Note on requirement 13

The subresource /documents/{uuid}/resource gives you a resource link to the PDF. If you access to that resource over a document, you will receive this object:

```
{
  "resourceLink":
  "https://researchpoolstoragedev.blob.core.windows.net/f434c0ab-8824-51ff-8f76-7aae10a8411a/b5197a39-ddf8-4cc4-84d1-795cdabd2bda.pdf?sv=..."
}
```

This is the link to a PDF file. You can embed using an iframe. NOTE: for some documents there could be no resource. In those cases, the API will return HTTP 404. You must catch this error and show a message saying the resource is not available.

That's all, and good luck!