Brayn.io Code Challenge

We at brayn.io value good code quality and your possible future colleagues want to see how you write and structure code. Also, we want to see how you make your code easily understandable for other developers (readability, comments).

We prepared two tasks for you that use our existing API-Billing API to give you an opportunity to show off your skills! API-Billing is a web application that offers a real-time financial dashboard and automates accounting and payment processes. It can also be easily integrated in different systems (one of these systems is your code challenge, so to speak 😊).

We created two tasks; one is for Frontend Cracks and the other for Logic Freaks. You may do both tasks or decide for only one of them. Please solve the task and send us your solution. Necessary resources & credentials can be found below the tasks. Please do not share the tasks, resources & credentials, or the code publicly.

Frontend Cracks

(1) Please create an application which gives an overview of created invoices using the endpoint `GET` **/v1/invoices/debit/list**

The overview should include at least the following information:

- Invoice number
- Invoice date
- Net amount (netto amount)
- Gross amount (brutto amount)
- Open amount (balance)
- Debitor

(2) Additionally, provide a detail-view of a single created invoice using `GET` **/v1/invoices/debit/list/{id}**

Here we should have at least all the following information of an invoice:

- Invoice number
- Invoice date
- Debitor
- Service period
- Due_date
- All the items including Description, Amount, Price, Vat_rate
- Sum (total netto)
- Brutto (Total Brutto)
- Open amount (balance)
- Total VAT would be great addition!

(3) Optional A download of the invoice-PDF which is also provided by our API invoice would be a great addition!

Logic Freaks

(1) Please retrieve the invoices of the last 12 months using `GET`/v1/invoices/debit/list
And give a reasonable forecast for the current month.

(2) Additionally, give the possibility to export this, for example as an Excel-file. The export should include at least the following information for the invoices of the last 12 months:

- Invoice number
- Invoice date
- Net amount (netto amount)
- Gross amount (brutto amount)
- Open amount (balance)
- Debitor

(3) Optional A graphical presentation of the invoices of the last 12 months and the forecast would be a great addition!

4 von 4

Resources & Credentials

The basis for the task is our existing API.

The documentation can be found here:
https://petstore.swagger.io/?url=https://api.fynbill.fynbird.io/swagger.json
The API is available here: https://api.fynbill.fynbird.io
The authentication endpoint here: https://api.fynbill.fynbird.io/oauth

Please use the following user data for accessing the API:

username: frontendDemo@secu-ring.de
password: n*E%<_eAYd$Gvixe{eV4je>H
client_id: demo_fe
client_secret: WSBTUj2pfqrR4KMYSyMcnDQrMfQTrHMV/9Mf/TLjcRQ=
grant_type: password

The API uses OAuth2 authentication with a password. The authentication endpoint expects the above user credentials as a JSON-formatted POST request. Include the returned Access-Token as a **Bearer-Token** (append "Bearer " to the token) in the **Authorization** header field of any following HTTP-Requests to the API.

If you lose access to the API, you might need to request a new token. For more information on how OAuth2 authentications works, check out:
https://www.oauth.com/oauth2-servers/access-tokens/password-grant/.

For the tasks, you don't have to implement a login page or similar. It is sufficient when the user credentials are supplied from stored variables. If you have any questions or get stuck, feel free to contact us.