Quality Assurance: Practical Task

Hi there and thank you for your interest in working with us 😊

We would really like to know more about the way you work, so we think that starting with a small Quality Assurance Practical Task is a great way to get an idea of your style. Since we value your time a lot, we do not think that you should spend more than 2 hours on this task.

We are sharing a real Shop Apotheke case in which we would like you to:

- Write a test plan / test approach in bullet point format.
- Design one detailed test case. Others can be reduced to headlines.
- Implement test automation for one of the two following (you decide which one you prefer):
  - endpoint defined by contract, using JAVA or JavaScript/TypeScript
  - functionality displayed in GUI, using Java or JavaScript with Selenium or Cypress
- Document any findings in bullet point format

Please store the end result as a public GitHub repository. We will have a follow up call, where you can guide us through your implementation and discuss your decisions.

Task details

Below please find the:

- Business requirement(s)
- API Contract (YAML)
- Screen designs (Abstract)
- Link to frontend implementation

Business requirements

A customer is able to successfully log in with valid credentials

Acceptance Criteria

- Using valid credentials, a customer can successfully log in.
- Using invalid credentials will not lead to a successful login.

Service development

An endpoint has been provided to log in a customer with given credentials
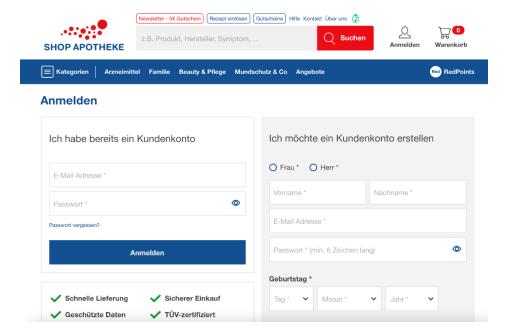
Contract

```
openapi: 3.0.3
info:
  title: API documentation
  contact:
    name: Shop Apotheke
    url: https://www.shop-apotheke.com
    email: info@shop-apotheke.com
  description: API Server
  license:
    name: private
  termsOfService: 'www.shop-apotheke.com'
  version: 1.0.0
servers:
  - description: default
    url: 'https://{host}/{cacheable}'
    variables:
      cacheable:
        enum:
          - cacheable
          - static
        default: cacheable
      host:
        enum:
          - api.sa-tech.de
        default: api.sa-tech.de
paths:
  '/auth/v1/{tenant}/login':
    post:
      parameters:
        - allowEmptyValue: false
          deprecated: false
          description: ControllerParam tenant
          in: path
          name: tenant
```

```yaml
          required: true
          schema:
            enum:
              - at
              - ch
              - com
              - fr
              - es
              - it
            type: string
        - $ref: '#/components/parameters/customer-authentication-device-type-
header'
      requestBody:
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/LoginRequest_v1'
        required: true
      responses:
        '201':
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/LoginResponse_v1'
          description: SuccessResponse
        '400':
          $ref: '#/components/responses/BadRequest-authentication'
        '504':
          $ref: '#/components/responses/504'
      tags:
        - AuthController
components:
  parameters:
    customer-authentication-device-type-header:
      deprecated: false
      in: header
      name: Sae-Device-Type
      required: false
      schema:
        enum:
          - browser
          - iosApp
          - androidApp
        type: string
        example: 'browser'
        default: 'browser'
        description: >-
          Describes the type of a device that is used to access the API.
Defaults to Browser if none is specified
  schemas:
    LoginRequest_v1:
      example:
        customerNumber: '123456789'
        password: aA1!bB2@cC3#dD4$
      properties:
        customerNumber:
          type: string
```

```yaml
      password:
        type: string
    LoginResponse_v1:
      example:
        token:
eyJhbGciOiJSUzM4NCJ9.ewogICJzdWIiOiAiMTExOTg5OTM0IiwKICAicm9sZSI6ICJSRUdJU1RF
UkVEIiwKICAidGVuYW50IjogImNvbS1xcyIsCiAgInR5cGUiOiAiQUNDRVNTIiwKICAia2V5VmVyc
2lvbiI6ICJmMjQ1MzVlOC0yMzQyLTQ2N2QtODVmYy1jNmJkYWIxZDY4MmUiLAogICJleHAiOiA5Nj
A1ODg1MDc3LAogICJleHBSZWZyZXNoIjogIjIwMjItMTEtMTUtMTY6MTcuODY2WiIsCiAgImp
0aSI6ICI1NzljZjk5OC1kNjBkLTQ2YTktOWN1Ni1iYmE5ODJjY2YzZDMiLAogICJpc3MiOiAiYXV0
aC5yZWR0ZWNsYWIuY29tIiwKICAiaWF0IjogMTYwNTc5ODY3Nwp9
        tokenType: bearer
      properties:
        tokenType:
          type: string
        token:
          type: string
    Authentication400:
      properties:
        statusCode:
          type: number
        error:
          type: string
        message:
          type: string
        invalidParams:
          type: array
          items:
            $ref: '#/components/schemas/InvalidParam'
      example:
        statusCode: 400
        error: Bad Request
        message: Validation Exception
        invalidParams:
          - field: password
            key: parameter.error.empty
    GatewayTimeout:
      properties:
        statusCode:
          type: number
        error:
          type: string
        message:
          type: string
      example:
        statusCode: 504
        error: Gateway timed out
        message: Connection to service timed out
    InvalidParam:
      properties:
        field:
          type: string
        key:
          type: string
      example:
        field: subscriptionType
        key: mailing.subscription.missing.mandatory.field
```

```
responses:
  BadRequest-authentication:
    description: Bad Request
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/Authentication400'
  '504':
    description: Gateway Timeout
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/GatewayTimeout'
```

Screen design

The screen design can be seen below. The current implementation is also available on https://www.shop-apotheke.com/nx/login/

Have fun! We look forward to receiving your solution 😊