SERTIS: QA Take Home Test

Duration : 7 days since the test was received

Feel free to use any tools, technologies, programming languages or libraries to complete the exercises below. In case that we don't provide enough information, you can make assumptions and keep a note about it in your answer.

We encourage you to give your best effort, and please know that it is fine if you are unable to complete every task. We are more interested in seeing how you approach the problem and the quality of your work than in complete solutions to every challenge.

If anything is unclear, feel free to reach out to us to clarify. Upon completion, please submit all relevant code and notes/documentation.

1. Imagine the team is tasked with testing the functionality of this website
   a. Please list out the tasks required to test the website and estimate how long each should take (in days). In addition, please prioritize each task by importance.
   b. Design the end-to-end test case or test scenario for the product purchasing flow, i.e. the process from selecting one or more products, adding them to the cart, to submitting an order. Be sure to cover both the happy and unhappy path.
   c. Develop an automated test script for the test case/test scenario which you designed in '**b**'
   d. How could you performance test this website? Please explain. (There is no need to actually implement the performance tests, but feel free if you want to)

2. You can find a simple API hosted at https://sertis-qa.glitch.me/ . There are three endpoints:
   - **[GET] /user/ids** - Get all user ids
   - **[GET] /user/:id** - Get user information for the specified user id
   - **[POST] /signin** - Sign in by specifying "phone_no" and "otp" of a particular user id in a request body

   To try the API, you can first get a list of all user ids, to then get the information of a specific user, and use it to perform the sign-in. API Documentation can be found in the appendix below.

   Please create a script which tests the functionality of the API.

3. We would like to further automate the test(s) you created in exercise **1** and **2**. Please create a CI pipeline configuration which automatically runs the tests upon merging to a specific branch. We use Gitlab CI in our team, but you are free to use any CI/CD tool you like (e.g. Gitlab CI, Github Actions, Jenkins, CircleCI, etc.)

Appendix: API Documentation

```javascript
JavaScript
openapi: 3.0.3
info:
 title: Sertis QA API
 description: |-
  This is a QA API Server for Sertis's QA Take Home Test

  _To try the API, you can first get a list of all user ids, to then get the information of
a specific user, and use it to perform the sign-in._

  Please create an automated test to test the functionality of the _"/signin"_
endpoint.

 contact:
  email: nroon@sertiscorp.com
 license:
  name: Apache 2.0
  url: http://www.apache.org/licenses/LICENSE-2.0.html
 version: 1.0.0
externalDocs:
 description: Find out more about Sertis
 url: https://www.sertiscorp.com/
servers:
 - url: https://sertis-qa.glitch.me/
tags:
 - name: user
  description: Operations about user
paths:
 /user/ids:
  get:
   tags:
    - user
   summary: Get user ids
   description: Get all available user ids in the system
   responses:
    '200':
     description: A JSON object contains all available user ids
     content:
      application/json:
       schema:
        type: array
        items:
         type: string
        example: ["01", "02", "03"]
    '5XX':
     description: Unexpected error

 /user/{id}:
  get:
   tags:
    - user
   summary: Get a user by ID
   description: Get all available user ids in the system
   parameters:
```

```yaml
      - in: path
        name: id
        schema:
         type: string
        required: true
        description: ID of the user to get
      responses:
       '200':
        description: A JSON object of a particular user
        content:
         application/json:
          schema:
           type: object
           properties:
            first_name:
             type: string
            last_name:
             type: string
            permission:
             type: string
            phone_no:
             type: string
            otp:
             type: string
           example:
            first_name: John
            last_name: Doe
            permission: admin
            phone_no: "020011893"
            otp: "123456"

       '400':
        description: Invalid user id
        content:
         application/json:
          schema:
           type: object
           properties:
            status_code:
             type: string
            message:
             type: string
           example:
            status_code: "400"
            message: "123 is not a valid id"
       '5XX':
        description: Unexpected error

  /signin:
   post:
    tags:
     - user
    summary: Sign in user
    description: Sign user into the system
    requestBody:
     description: User object
```

```yaml
      content:
        application/json:
          schema:
            type: object
            properties:
              phone_no:
                type: string
              otp:
                type: string
            example:
              phone_no: "020011893"
              otp: "012345"
responses:
  '200':
    description: successful operation
    content:
      application/json:
        schema:
          type: object
          properties:
            status_code:
              type: string
            status:
              type: string
            data:
              type: object
              properties:
                id:
                  type: string
                first_name:
                  type: string
                last_name:
                  type: string
                permission:
                  type: string
            message:
              type: string
          example:
            status_code: "200"
            status: "Pass"
            data:
              id: "01"
              first_name: QA
              last_name: Sertis
              permission: admin
            message: "Sign in success"
  '404':
    description: User not found
    content:
      application/json:
        schema:
          type: object
          properties:
            status_code:
              type: string
            status:
```

```yaml
      type: string
     data:
      type: object
     message:
      type: string
    example:
     status_code: "404"
     status: "Not found"
     data: {}
     message: "Sign in success"
'500':
 description: Internal Server Error
 content:
  application/json:
   schema:
    type: object
    properties:
     status_code:
      type: string
     status:
      type: string
     data:
      type: object
     message:
      type: string
    example:
     status_code: "500"
     status: "Fail"
     data: {}
     message: "Internal Server Error"
```