| **Name:** | Wong Ru Peng |
| --- | --- |
| **Email:** | wong\_ru\_peng@mpa.gov.sg |

**Question 1**

In your own words explain the 4 sub constraints in REST’s Uniform Interface. Give an everyday example to illustrate each of the constraint.

1. **Identify resources thru identifiers**

**A resource name within the server. Eg, URL that we entered into the browser**

1. **Manipulation of resources through its representation**

**A single resource can be represented by many different types of media. Eg, a report can be download in PDF format, csv format etc.**

1. **Self describing message**

**Every message passed from a client to a server is self contained and everything to understand and decode the message is in the message. Eg, a postcard**

1. **Hypermedia as the engine of application states**

**A client interacts with a network application entirely through hypermedia provided dynamically by application servers. A REST client needs no prior knowledge about how to interact with any particular application or server beyond a generic understanding of hypermedia.**

**Question 2**

What is the difference between the following HTTP methods?

1. POST, PUT and PATCH

* **POST - Used to submit data to the specified resource**
* **PUT - Replaces the data of the specified resource**
* **PATCH - Partially modify the specified resource**

1. GET and HEAD

* **GET - Request a representation/data from the specified resource**
* **HEAD - Like GET but without the response payload (if any)**

**Question 3**

You have a monolithic web application for managing warehouses. The application exposes the following end points

* /warehouses – list of all warehouses
* /warehouse/<warehouse\_id> – returns the warehouse’s details
* /warehouse/<warehouse\_id>/inventories – inventory list for the warehouse
* /inventories – list of all the inventories
* /inventory/<inventory\_id> – inventory detail
* /inventory/<inventory\_id>/report – generate a report

Describe how you can scale this application

1. By duplication

**By adding additional web servers to handle the load but the web servers are connected to the same database.**

1. By functional decomposition

**By using microservice architecture.**

1. By data partitioning

**By spliting into 2 database. Warehouse and inventories.**

**Question 4**

Study the top headlines REST API from newsapi.org. Answer the following questions

1. List the different ways you can present the API key when performing an invocation

* **Via the apiKey querystring parameter.**
* **Via the X-Api-Key HTTP header.**
* **Via the Authorization HTTP header.**

1. Construct a URL to get 30 technologies headlines from Japan

[**https://newsapi.org/v2/top-headlines?country=jp&category=technology**](https://newsapi.org/v2/top-headlines?country=japan&category=technologies&pageSize=30&apiKey=API_KEY)

[**&pageSize=30&apiKey=API\_KEY**](https://newsapi.org/v2/top-headlines?country=japan&category=technologies&pageSize=30&apiKey=API_KEY)

1. What is the status code if an incorrect API key is used?

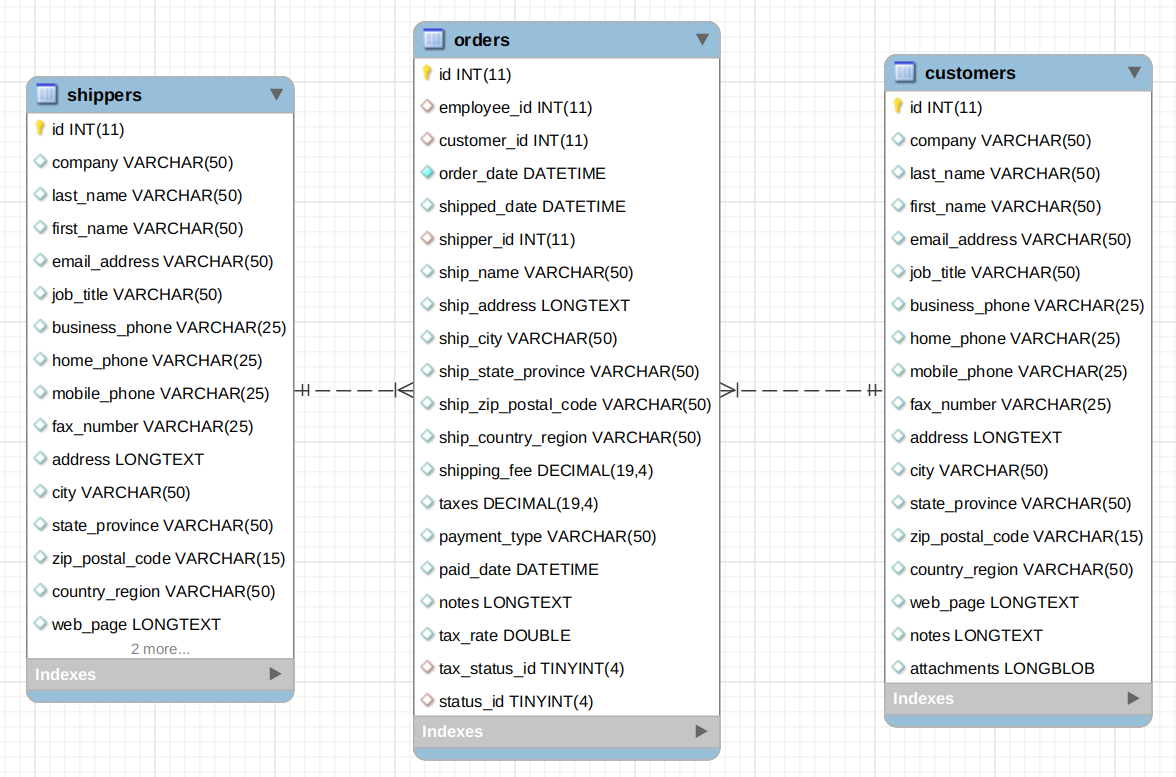
**401 - Unauthorized HTTP error.**

1. How long will the result be cached?

**3 minutes.**

**Question 5**

Study the following entity-relationship diagram



Both customer and shippers has a one to many relationships with orders.

Answer the following questions.

1. Design one or more API endpoints to return a list of customers and a single customer

**GET /api/customers**

**GET /api/customer/{id}**

**GET /api/customer?search={search terms}**

**GET /api/customers?offset=10&limit=20**

1. What are some criteria and how might you might include in your endpoint (wrt Q5a)?

* **fieldset**
* **pagination**
* **filtering**
* **searching**

1. Show a sample output of a customer’s list as a result of performing a GET on the resource. (wrt Q5a)

**{**

**“customer”: [{**

**“id”: “1234”,**

**“company”:”ABC Pte Ltd”,**

**.........**

**}]**

**}**

1. How do you provide flow control or pagination support (wrt Q5a)?

**Introduce throttling for flow control with offset and limit parameters for pagination support.**

**Question 6**

You have deployed a service to encode video viz. convert AVI to mp4, etc. Subscribers of your service uploads their video to the service; after conversion the converted video is returned to the subscriber (assume that the conversion time is short).

You charge the subscribers based on the 2 criteria.

1. Subscription rates based on the cumulative video sizes: 500GB, 1TB, 1.5TB, etc. A subscriber who subscribe to the 500GB package can upload a maximum amount of 500GB videos.
2. Charge the subscribe based on their ingress and egress traffic viz. the upload and downloads of the videos.

Design an API for this encoding service to give your subscribers control over their encoding process.

You can ignore authentication.

* **Subscription will be calculated using the apiKey passed in.**
* **API**
* **implement compression**
* **implement chunking**
* **implement multipart/form-data**

**POST /api/convert**

**Submission**

Copy this Word document to your repository and commit it.

git add .

git commit -m ‘worksheet01’

git push origin master