| **Name:** | **Alvin Tandian** |
| --- | --- |
| **Email:** | **alvin\_tandian@mpa.gov.sg** |

**Question 1**

In your own words explain the 5 sub constraints in REST’s Uniform Interface. Give an everyday example to illustrate each of the constraint.

* **Resource Identification  
  every resource must be uniquely identified.  
  Example: *User is able to access customer records by accessing /api/customer endpoint.***
* **Manipulate Resource Thru its Representation  
  Any resource can be represented by various different types of media.  
  Example: *User is able to retrieve bill / invoice in the form of PDF / Image / text file.***
* **Self Describing Message  
  Each message should have enough information on how to process them.  
  Example: *The response from the Web server should include “Accept” header flag to indicate what is the expected content type for the response (e.g. application/json).***
* **Hypermedia as the engine of application states (HATEOAS)  
  Using hyperlink / hypertext to navigate / change the application state.  
  Example: *User is able to click a link to navigate from Home screen to Profile screen.***

**Question 2**

What is the difference between the following HTTP methods?

1. POST, PUT and PATCH

**POST: Submit / create a new data for the specified resource.**

**PUT: Updates the entire data for the specified resource.**

**PATCH: Updates a partial data existing data for the specified resource.**

1. GET and HEAD

**GET: Request for a specified resource.**

**HEAD: Request the header (without the payload) of a specified payload.**

**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 deploying the web application to multiple servers and provide load balance to balance the incoming traffic. E.g. Duplicate the web server.**

1. By functional decomposition

**By converting the monolithic application to microservices which will manage their own data. E.g. warehouse microservice, inventory microservice.**

1. By data partitioning

**By splitting the database to cater for different resources. E.g. database A is catered for warehouse, database B is catered for inventory.**

**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&pageSize=30&apiKey=API\_KEY**

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

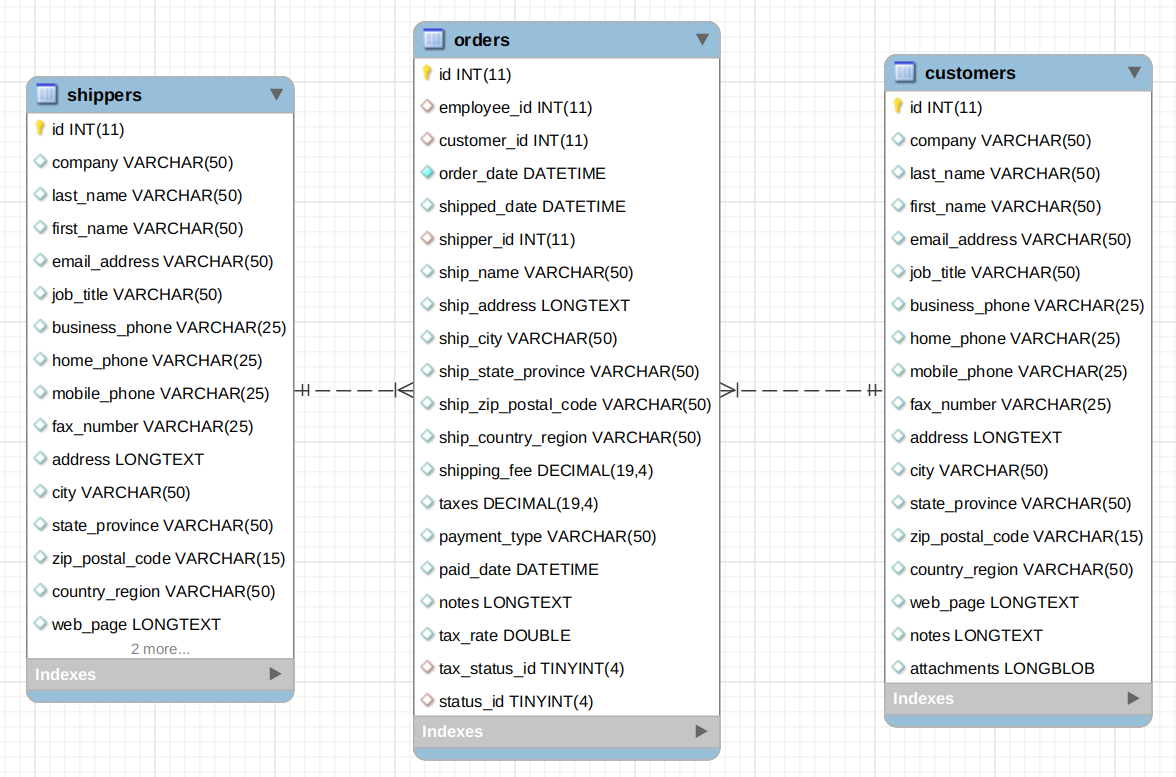
**401 - Unauthorized**

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

**Return a list of customers  
GET api/v1/customers  
GET api/v1/customers?offset=1&limit=10**

**Return a single customer  
GET api/v1/customer/<id>  
GET api/v1/customer?q=[first\_name=?|last\_name=?|company=?|email\_address=?]**

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

* **Allow searching capabilities by introducing query parameter. (e.g. search based on first name / email address).**
* **Allow pagination by having offset and limit.**

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

"status": "ok",

"totalResults": 2,

,"data": [{

"id": 1,

"company": "MPA",

"last\_name": "Murdock",

"first\_name": "Matt",

"email\_address": "matt\_murdock@mpa.gov.sg",

"job\_title": "Manager",

"business\_phone": "+6581818181",

"home\_phone": "+6582828282",

"mobile\_phone": "+6583838383",

"fax\_number": null,

"address": "mTower",

"city": "Singapore",

"state\_province": "Singapore",

"zip\_postal\_code": "119963",

"country\_region": "Singapore",

"web\_page": "https://www.mpa.gov.sg",

"notes": "",

"attachments": null

}, {

"id": 2,

"company": "MPA",

"last\_name": "Zhen",

"first\_name": "Jane",

"email\_address": "jane\_zhen@mpa.gov.sg",

"job\_title": "Director",

"business\_phone": "+6581818181",

"home\_phone": "+6582828282",

"mobile\_phone": "+6583838383",

"fax\_number": null,

"address": "mTower",

"city": "Singapore",

"state\_province": "Singapore",

"zip\_postal\_code": "119963",

"country\_region": "Singapore",

"web\_page": "https://www.mpa.gov.sg",

"notes": "",

"attachments": null

}]

1. How do you provide flow control or pagination support (wrt Q5a)?
2. **Implement throtlling.**
3. **Reducing payload size by performing compression.**
4. **By introducing the *offset* and *limit* as parameter.**

**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.

**POST /api/v1/encode**

Request Object

* apiKey: The API key of the subscriber.
* encodeType: The targeted encoding type.
* video: The input video.

Response Object

* status: The status of the API.
* message: The status message for the API.
* data: The converted video.

Based on the API key, the API will determine the subscriber’s subscription rate.

Upon receiving the data, the backend will detect the size of the video payload and return error status if the file size is more than the allowed subscribed rate.

The backend will also pull the traffic info from the egress-ingress and insert a charge record to the specified subscriber.

**Submission**

Copy this Word document to your repository and commit it.

git add .

git commit -m ‘worksheet01’

git push origin master