



REST Web Services

with JAX-RS



Tata DA SILVEIRA

Software Engineer @  HACKAGES

 /adjetetatadasilveira

 /ghostGuiggs

tata@hackages.io

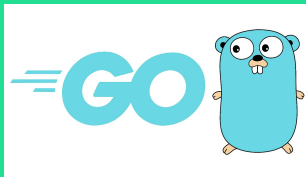


android 

Maven



Jenkins



muleESB 

What about you ?

- Introduce yourself
- What are your expectations for this training?



What you can expect from this session

1. Up to date relevant knowledge, concepts
2. Hands-on and challenge driven-training
3. Pair programming
4. Sharing
5. Answers to Software Engineering Questions

Course Sections

Section 1 REST API concepts

Section 2 Implementation with JAX-RS

What is a Web Services?

“ Web Services are services that are exposed to the internet for programmatic access. They are online API that you can call from your code. ”

Classic Application



The diagram illustrates a 'Classic Application' architecture. On the left, a large black rectangle represents the 'Client App', with the text 'Client App' centered in green. To the right of the client app, there is a vertical stack of three red rectangles, each containing the word 'Library' in white text. This represents a client application that depends on multiple external libraries.

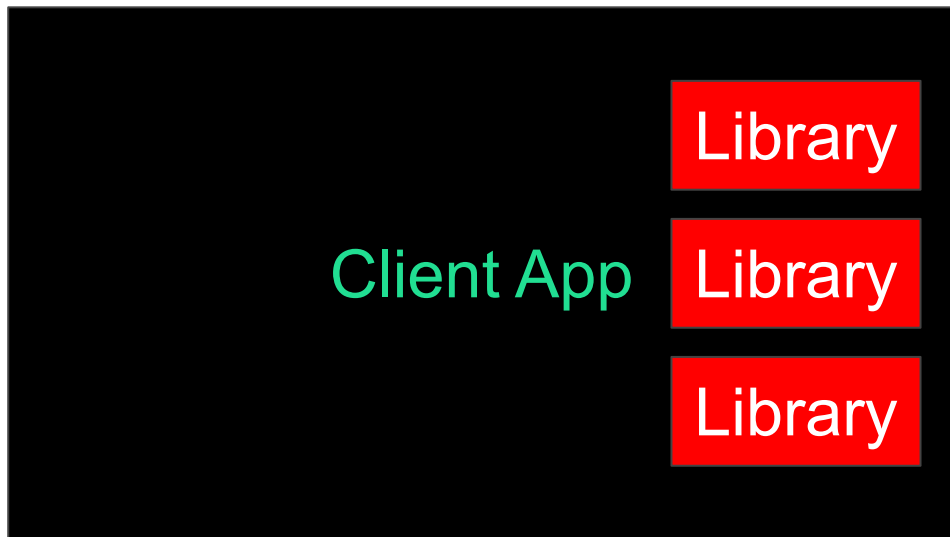
Client App

Library

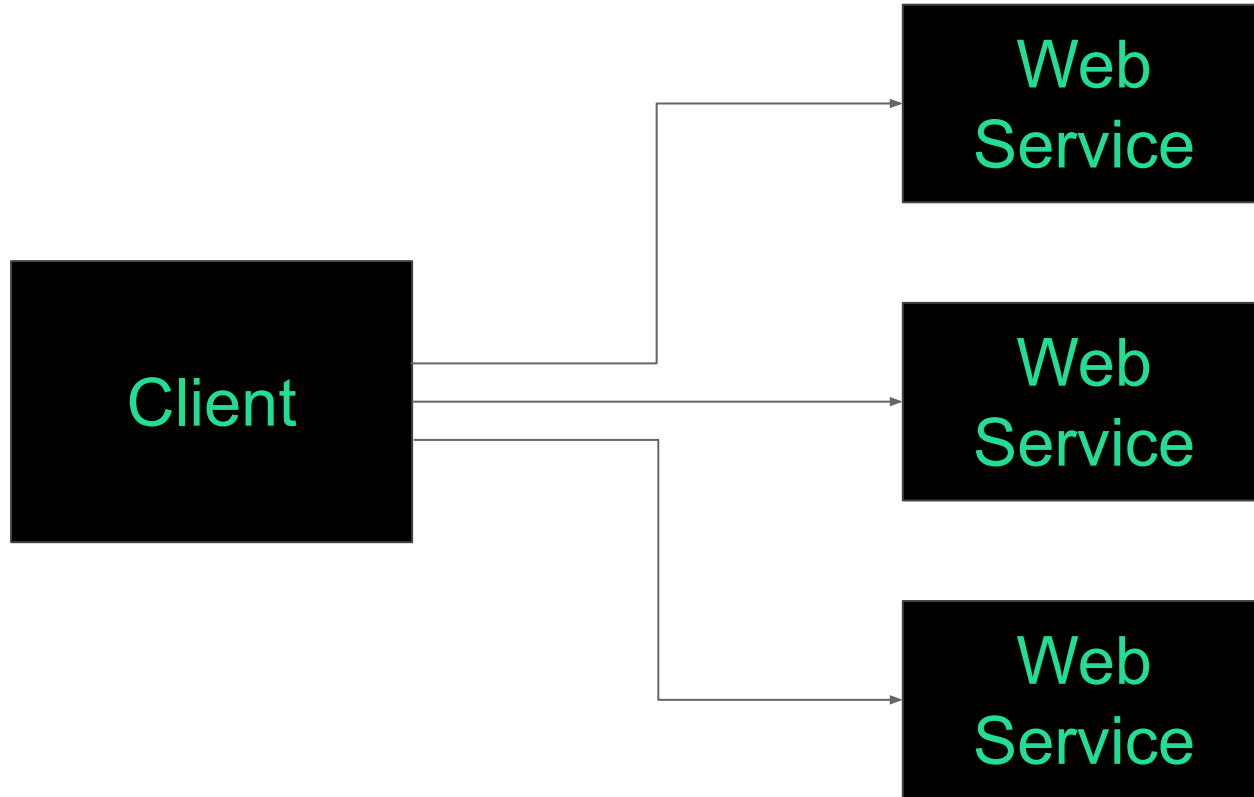
Library

Library

Classic Application



Web Service



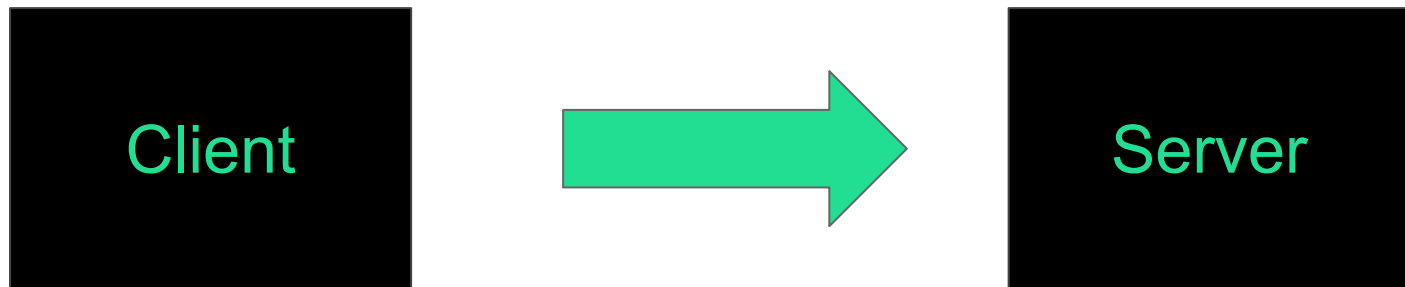
Web service characteristics

WEB

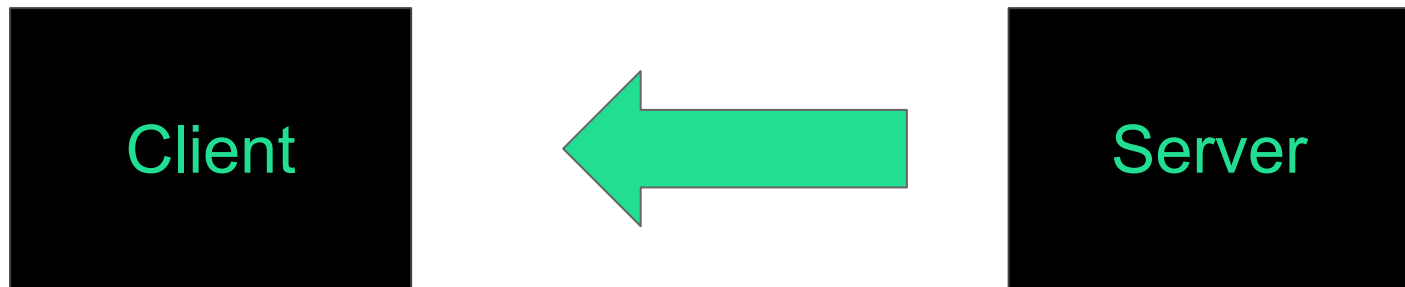


HTTP

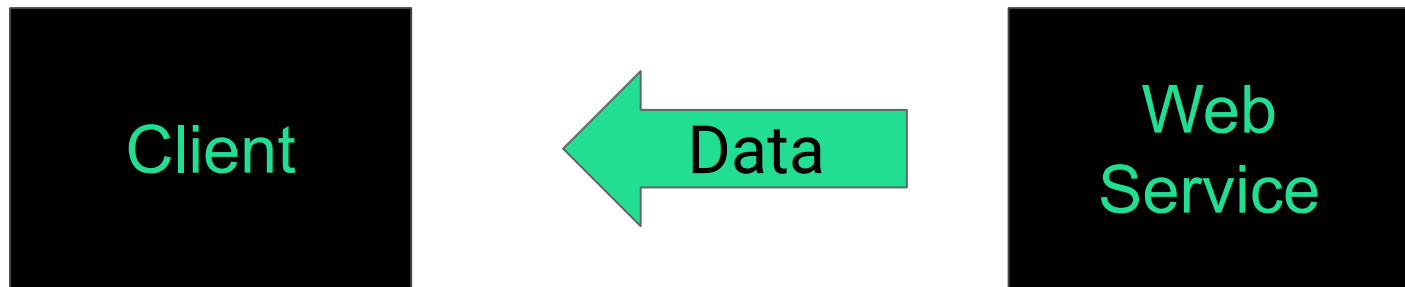
HTTP Exchange



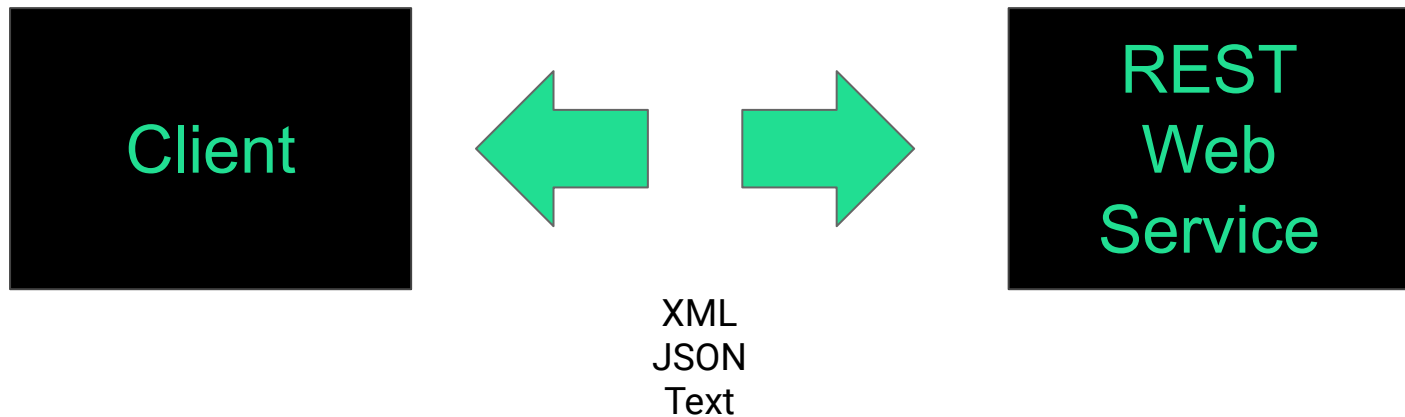
HTTP Exchange



HTTP Exchange



Protocol

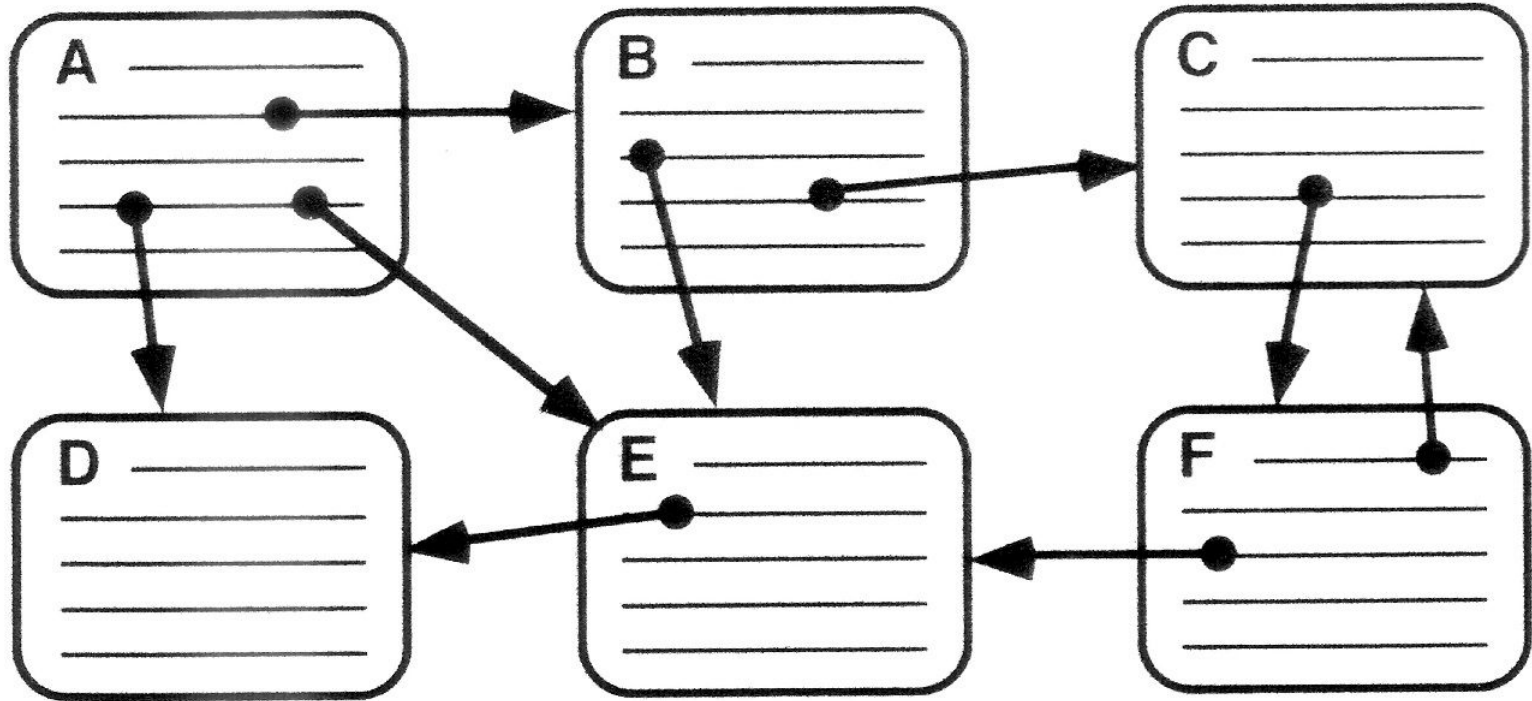


REST and HTTP

HTTP

Hyper Text Transfer Protocol

HyperText & HyperLink



REST

REpresentational State Transfer

Resource Locations

- Resource based URI
- Use Nouns in URI
- Use Plurals

`/users/{userId}`

Resource Locations

- Resource based URI
- Use Nouns in URI
- Use Plurals

/users/1

Resource Relations



- Articles URI `/articles/{articleId}`
- Comments URI `/comments/{commentId}`

Resource Relations

Article 1

Comment 1

Comment 2

Comment 3

Article 2

Comment 4

Comment 5

Resource Relations

Article 1

Comment 1

Comment 2

Comment 3

Article 2

Comment 4

Comment 5

Resource Relations

Comment 2

/articles/1/comments/2

/articles/{articleId}/comments/{commentId}

/articles/{articleId}/likes/{likeId}

/articles/{articleId}/shares/{shareId}

Instance

Resource URIs

/profiles/{profileName}

/articles/{articleId}

/articles/{articleId}/comments/{commentId}

/articles/{articleId}/likes/{likeId}

/articles/{articleId}/shares/{shareId}

Collection

Resource URIs

/profiles/

/articles/

/articles/{articleId}/comments/

/articles/{articleId}/likes/

/articles/{articleId}/shares/

Filtering results

`/articles?offset=30&limit=10`

Starting point

Page size

Custom Filters

`/articles?year=2017&offset=30&limit=10`

HTTP Methods

Common Methods

GET

PUT

POST

DELETE

Uncommon Methods

HEAD

OPTIONS

HTTP Methods

| Resource | GET (Read) | POST (Create) | PUT (Update) | DELETE (Delete) |
|-------------|---------------------------|--------------------------|---------------------------|---------------------------|
| /articles | Return a list of articles | Create a new article | Bulk update of articles | Delete all articles |
| /articles/1 | Return a specific article | Method not allowed (405) | Update a specific article | Delete a specific article |

Method Idempotence

Read-Only Method

| | |
|-----|--------|
| GET | POST |
| PUT | DELETE |

Write Methods

Method Idempotence

- It is safe to make multiples GET request?
- How about PUT, POST and DELETE ?

Method Idempotence

Repeatable
And
Non-Repeatable
Operations

Method Idempotence

Repeatable

| | |
|-----|--------|
| GET | POST |
| PUT | DELETE |

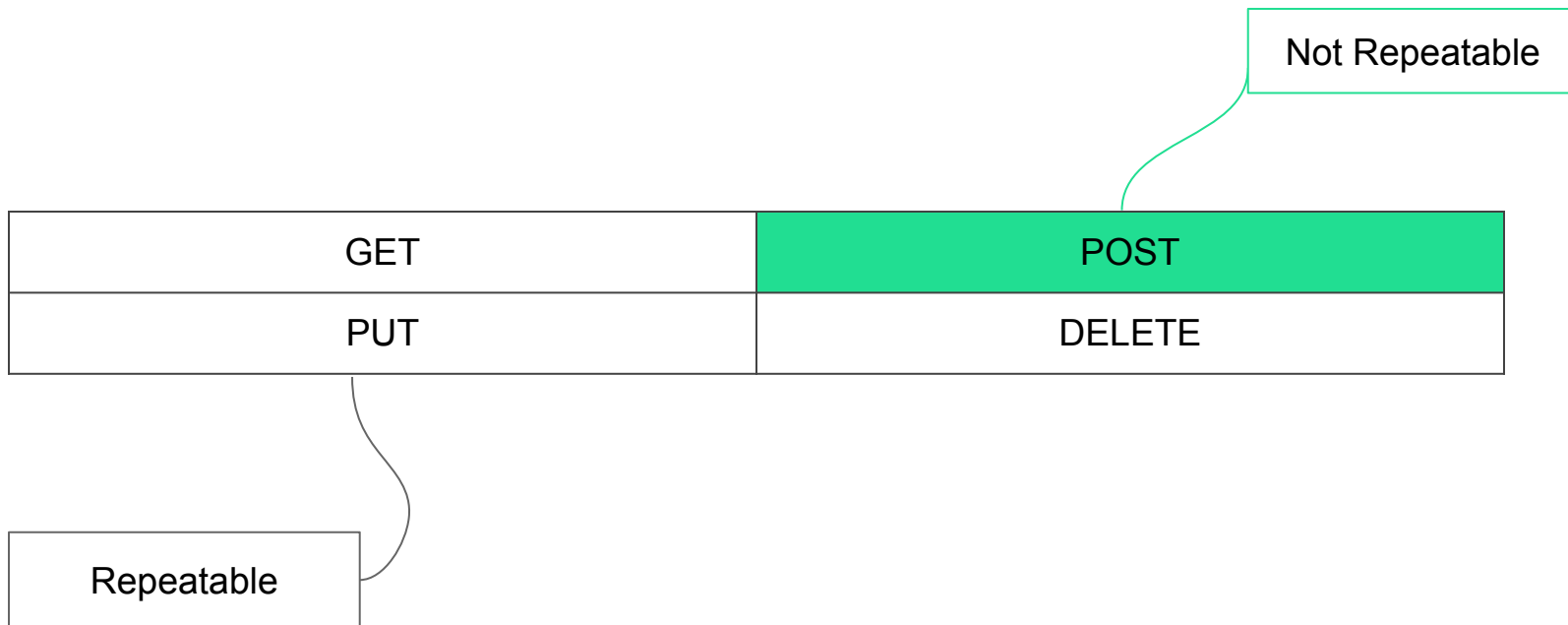
?

Method Idempotence

| | |
|-----|--------|
| GET | POST |
| PUT | DELETE |

Not Repeatable

Repeatable

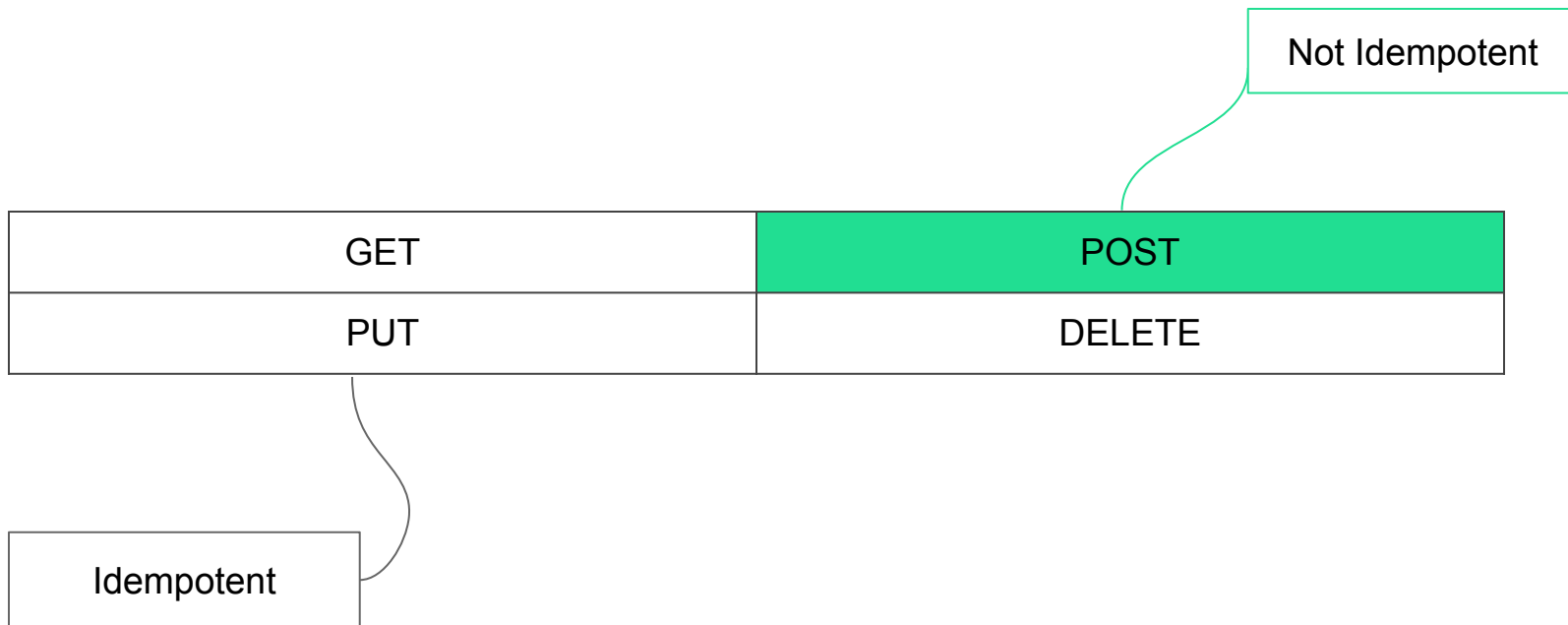


Method Idempotence

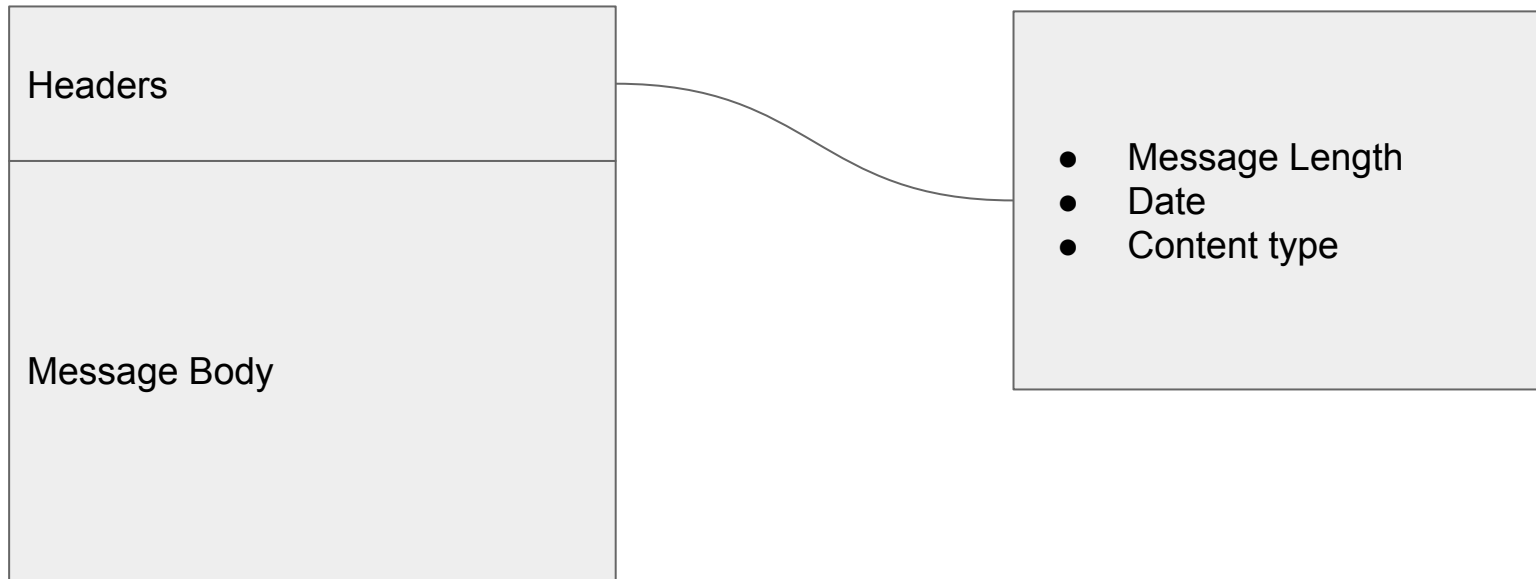
| | |
|-----|--------|
| GET | POST |
| PUT | DELETE |

Not Idempotent

Idempotent



REST Response



Status Code

- 1XX: Informational
- 2XX : Success Code
- 3XX : Redirection
- 4XX : Client Error
- 5XX : Server Error

HATEOAS

Hypermedia **A**s **T**he **E**ngine **O**f **A**pplication **S**tate

HATEOAS

REST
Has
No Service
Definition

HATEOAS

```
{  
  "author": "guy",  
  "id": 1,  
  "message": "Hello SFPD.",  
  "links": [  
    {  
      "link": "http://localhost:8080/sfpd-blog/webapi/articles/1",  
      "rel": "self"  
    },  
    {  
      "link": "http://localhost:8080/sfpd-blog/webapi/articles/1/comments",  
      "rel": "comments"  
    }  
  ]  
}
```

JAX-RS Annotations

HTTP Request Methods

- @GET
- @POST
- @PUT
- @DELETE
- @HEAD
- @OPTIONS

Resource

- @Path

Request-Response Media Types

- @Produces
- @Consumes

Request Parameters

- @PathParam
- @QueryParam
- @MatrixParam
- @HeaderParam
- @CookieParam
- @DefaultValue
- @Context
- @BeanParam
- @Encoded

Now is time for Hacking



Workshop: Fork the Github repository

<https://github.com/hackages/sfpd.training.lot6.g2>

Default branch : starting-point

`git remote add upstream https://github.com/hackages/sfpd.training.lot6.g2.git`

Keep the repo synced

<https://docs.github.com/en/free-pro-team@latest/github/collaborating-with-issues-and-pull-requests/syncing-a-fork>