

APIs

REST best practices

See, how to code APIs



API is a contract to connect to any Web - Application.

Razorpay



We should register ourself as a client on razorpay

client / secret  
credentials

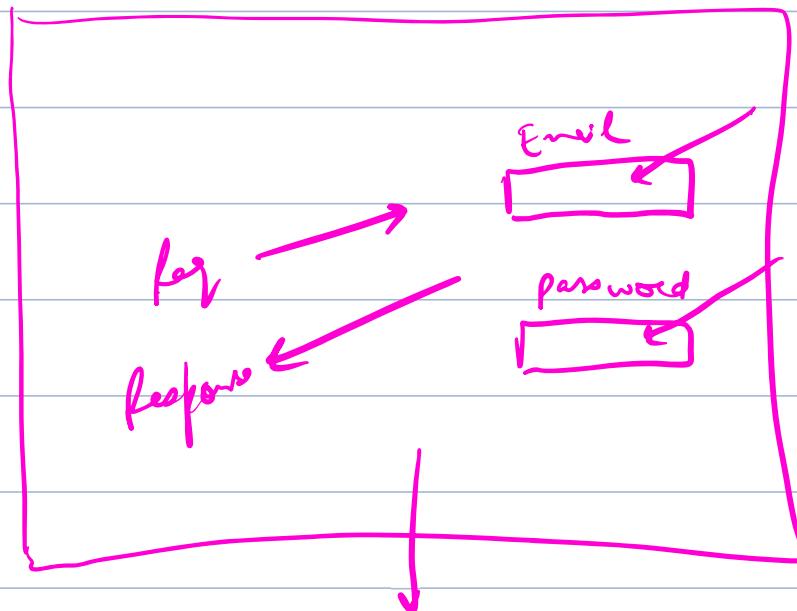


API

in which we will give  
details about, sender of,  
receiver a/c, ...

API is 2 way thing

↳ Reg Response



you are new user

Create account

you have added

wrong P/w

forget password

write exact  
password

https

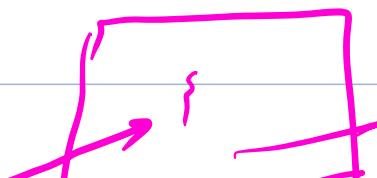
↓  
email  
of  
P/w



fronted

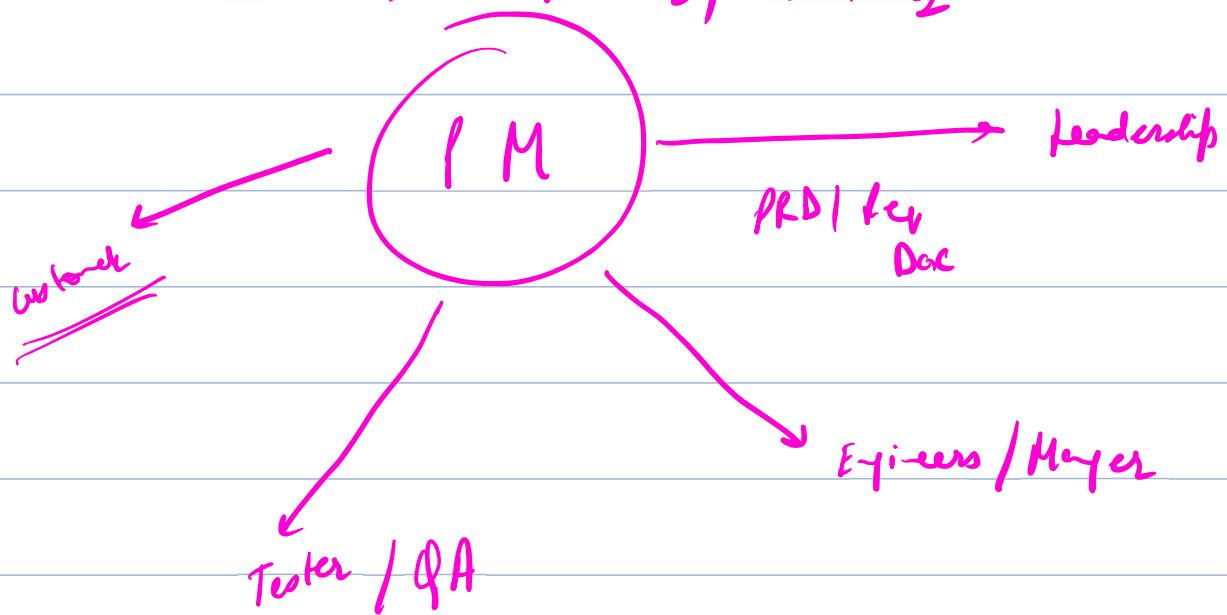


Backend





In All these PAANG / MAANG



Sign up - Page

↓  
Backend Team  
↳ 20 days

Frontend Team

↳ 10 days

30 days

2-3 days

B/E

d

F/E

API Contract

↳ Req  
Response

Possible Exceptions

10-12 days

B/E

|

F/E

4-5 days

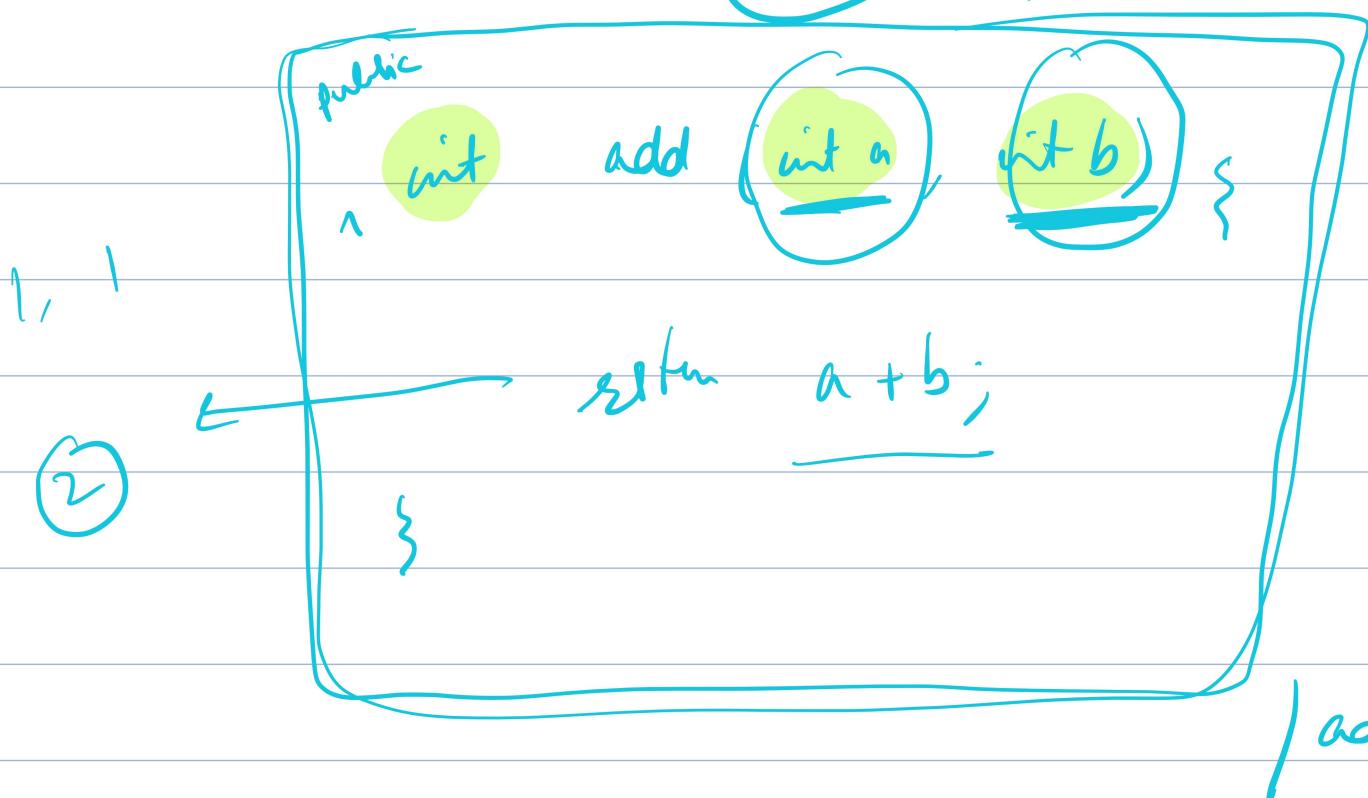
Testing

$$3 + 12 + 5 = 20 \text{ days}$$

API Contract



@ word path



scales.com playground

hosted

are run in |

host

| add / 1 / 2 ← API

VS

add(1,2) ← Method

int - 2

Method

first pose

2

# Status Code

# API

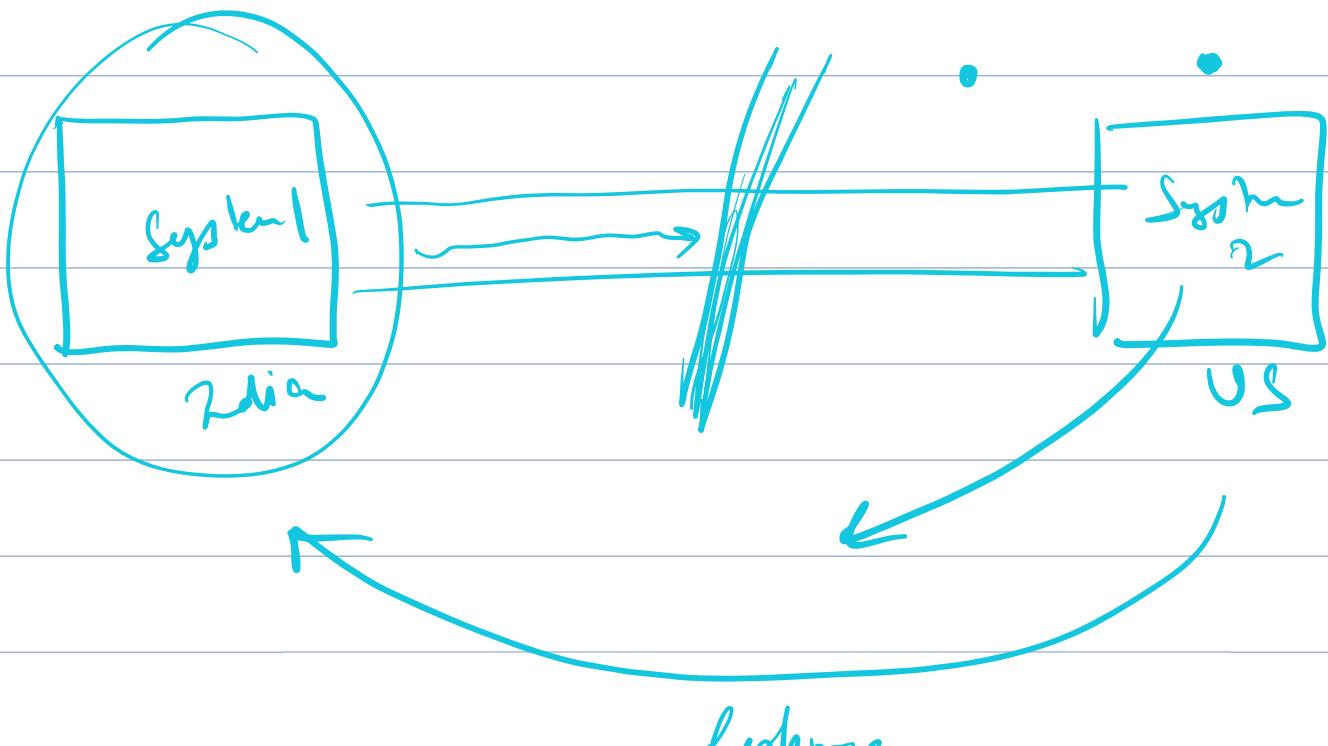
Header

# Exception

→ Send love

Meteolate

# Fractional



# REST Best Practice

I do - something - something



Yes / No

REST - ?

REST

Representational State Transfer



good to follow like  
SOLID

but there is No compilation

# What are these REST practices?

1.

REST says All should be defined around **resource** which is going to be affected.

Resource → which is getting stored in DB  
or

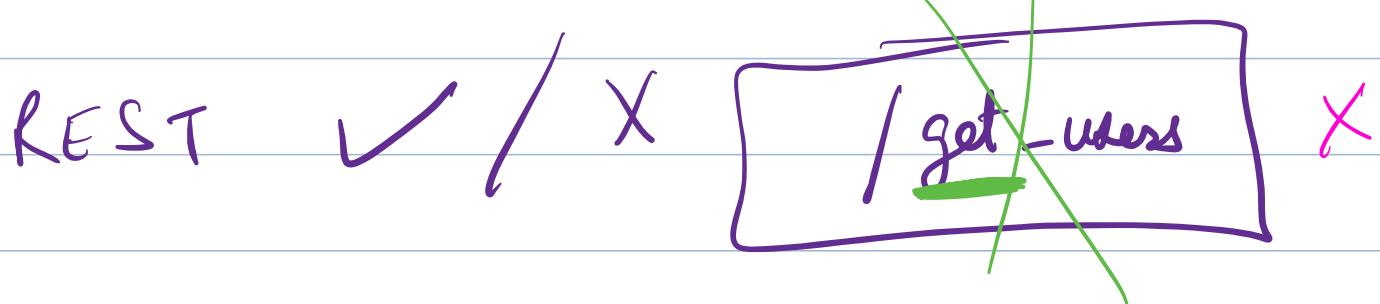
Entities / Tables

Affect



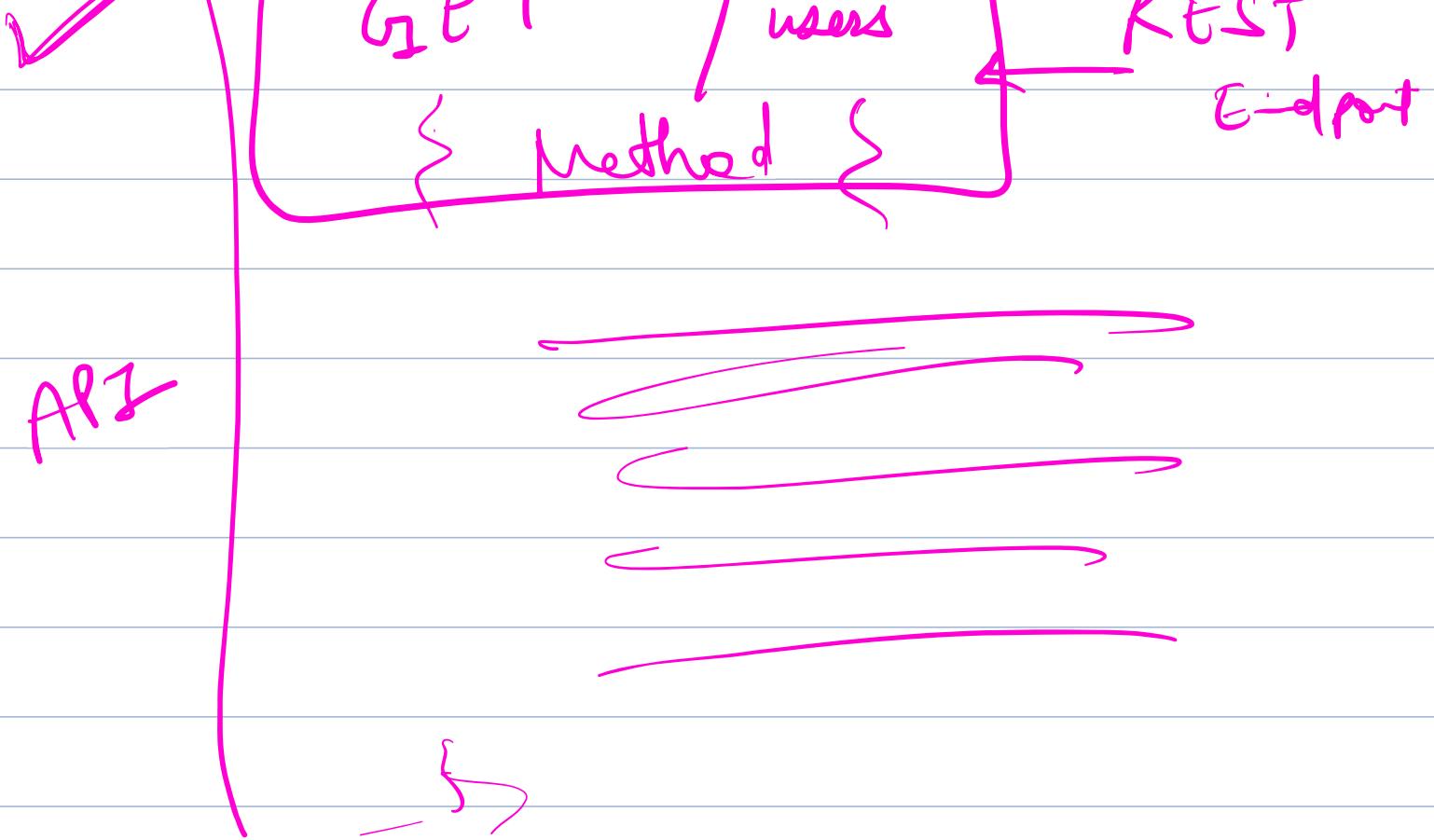
CRUD

Greek ✓  
Read ✓  
update ✓  
Delete ✓

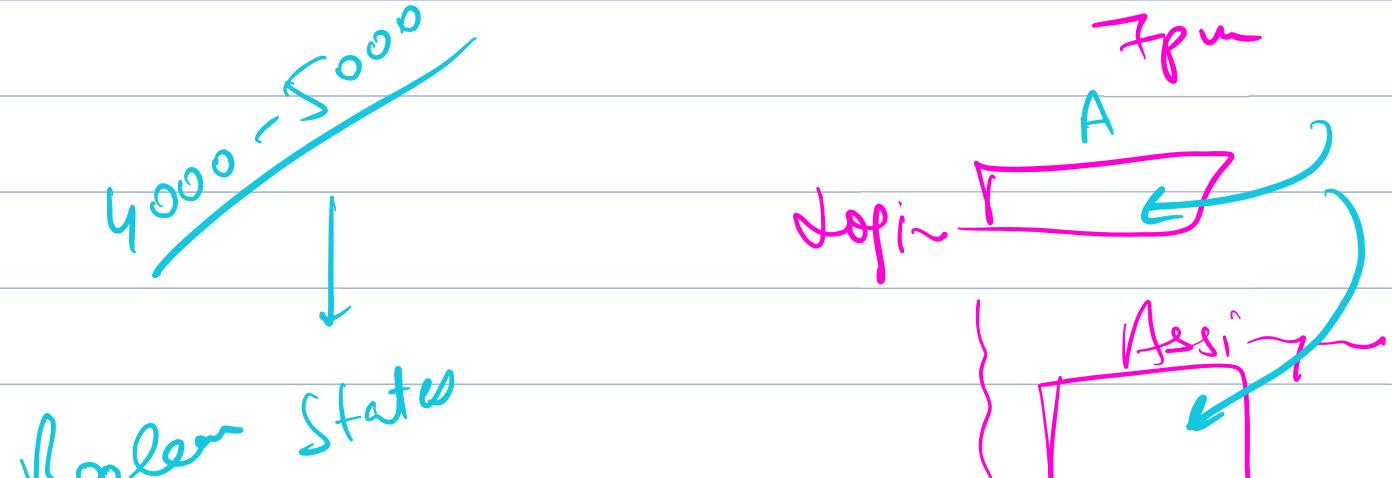


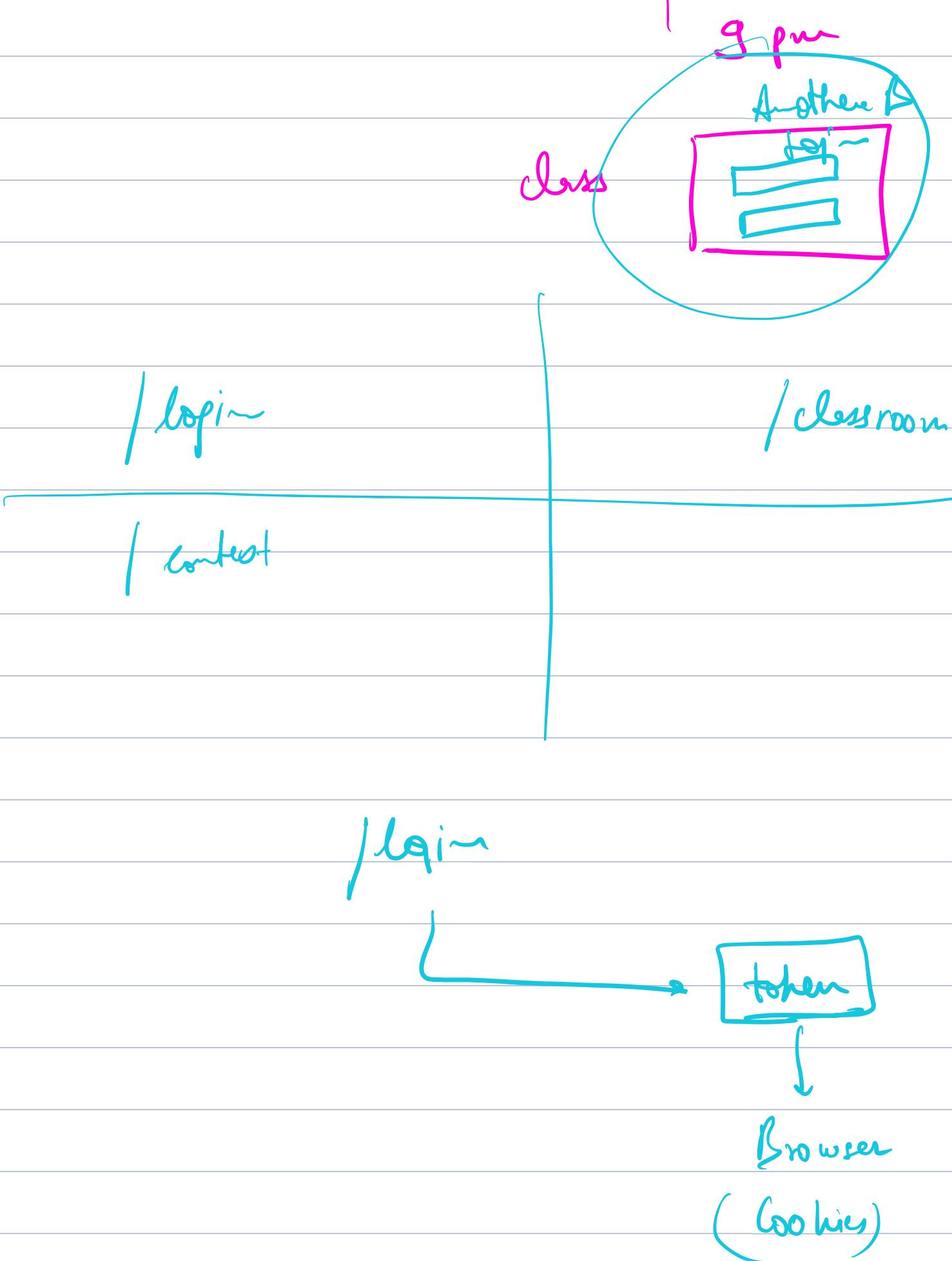
## API Method





2. REST APIs should be stateless





3. REST API should have relation with N tables instead of only one (Mapping)

Product

Name	Seller Id	Category Id	
------	-----------	-------------	--

Seller

Name	Phone No	Address	Rating
------	----------	---------	--------

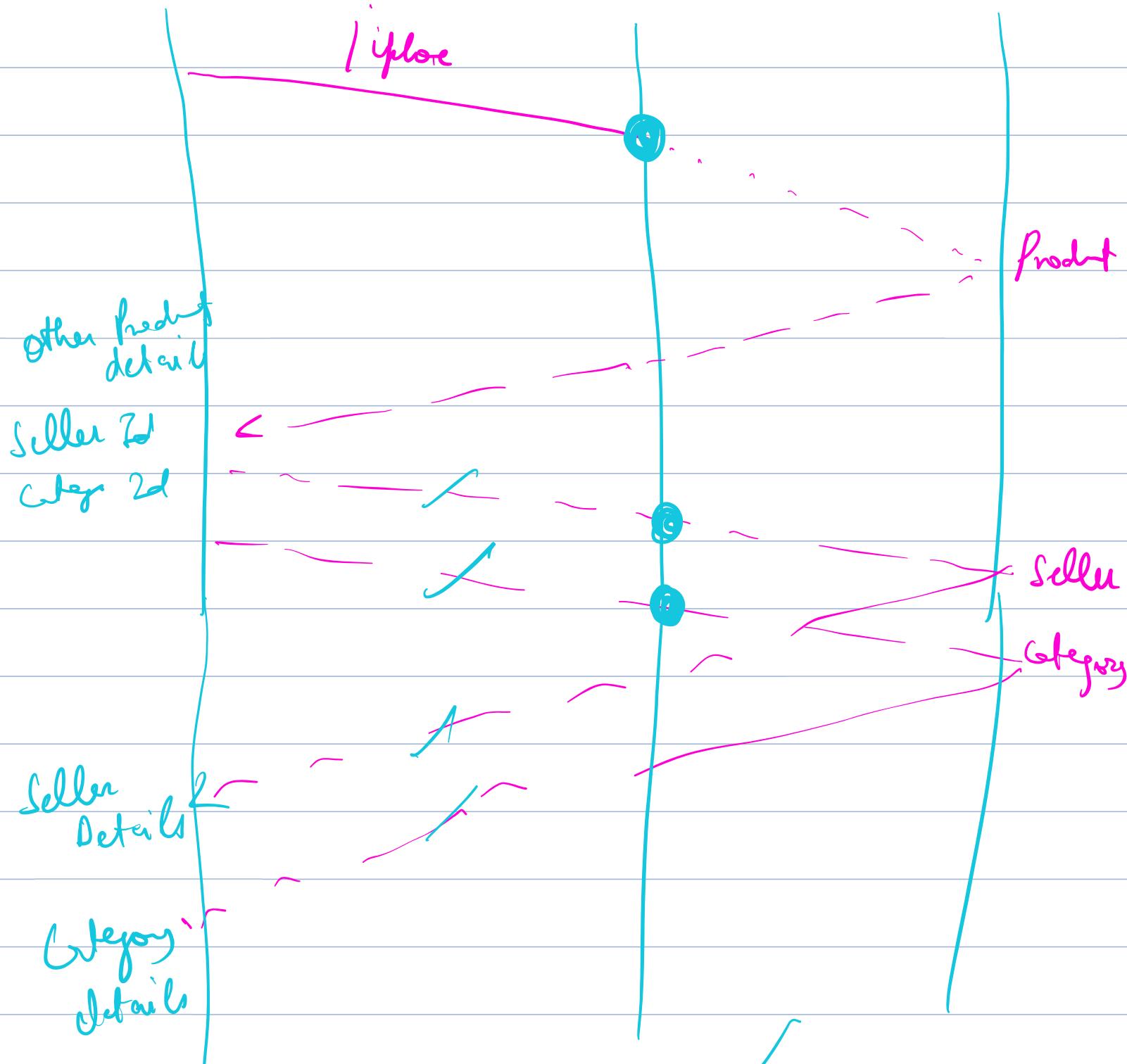
Category

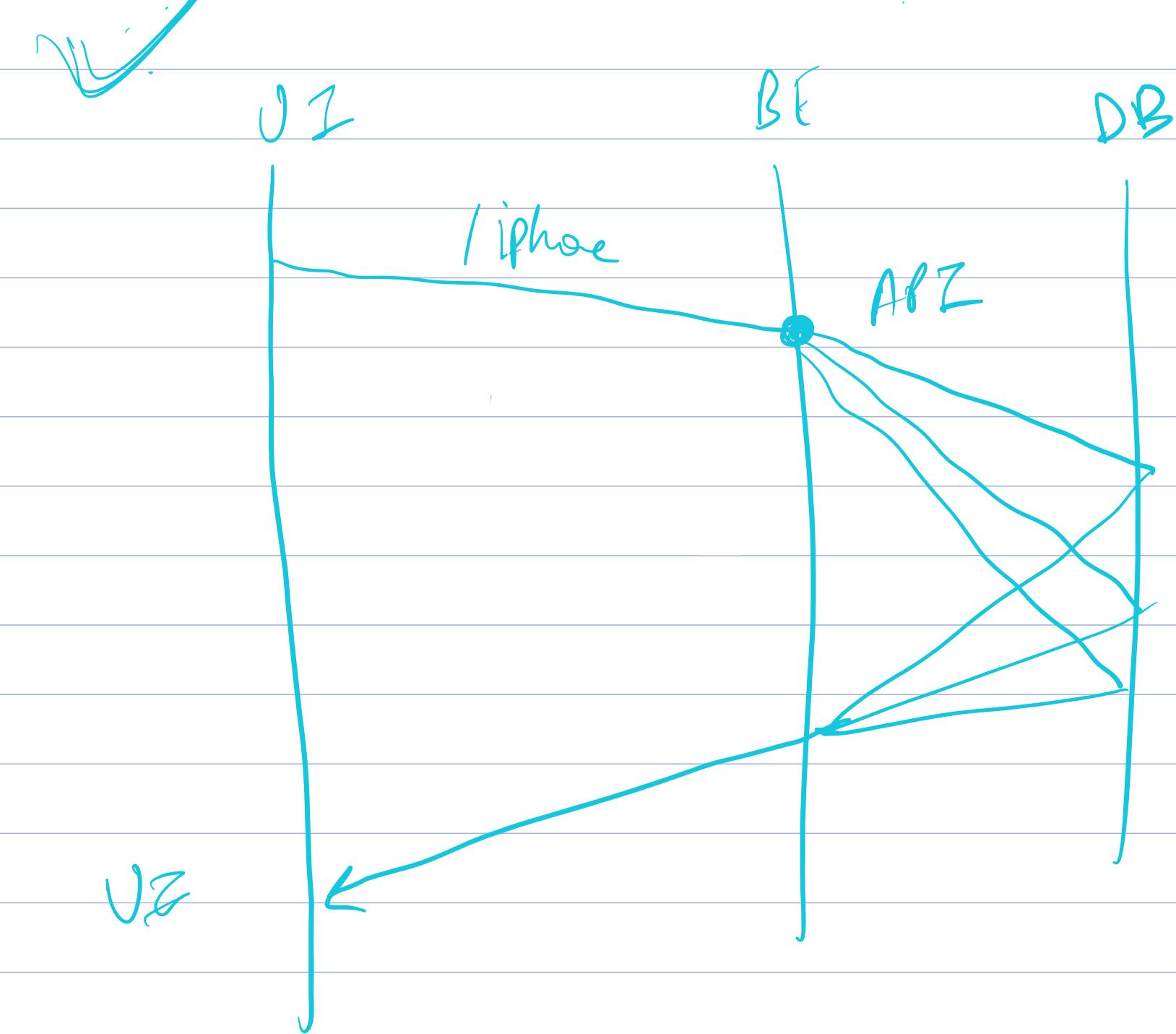
Name	Description	Rating	...
------	-------------	--------	-----

UI

B/E

Db





1 API



N Tables

(3)

4<sup>th</sup>

REST APIs should

talk in terms

of JSON | XML

JSON {  
key : value  
}

XML      <key> ~ </key>  
          <value> ~ </value>

1. How you can create first SB App

Brek