

## Diamond



## Platinum



## Gold



## Media Partners



## Training Partners



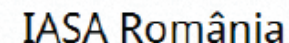
## Cultural Partners



## Wellness Partners



## Partner Communities



# Designing RESTfull WebServices and Web APIs

Best Practices

**Occasion:** CodeCamp Iasi

**Date:** 20-04-2013

**Present:** Remus Pereni / Software Architect / [remus.pereni@tss-yonder.com](mailto:remus.pereni@tss-yonder.com)

**Classification:** Public

Software Architect  
TSS-Yonder

Web & Mobile Web & Java

Twitter: @remuspereni

Web: <http://remus.pereni.org>

Yonder Romania

<http://tss-yonder.com>

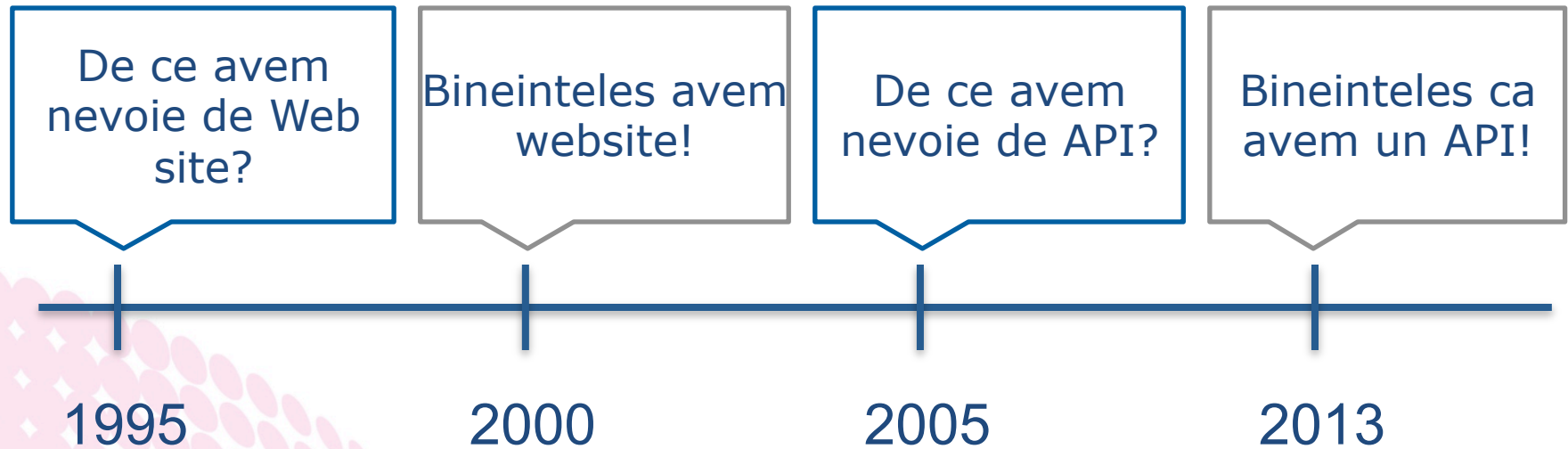
@tssyonder

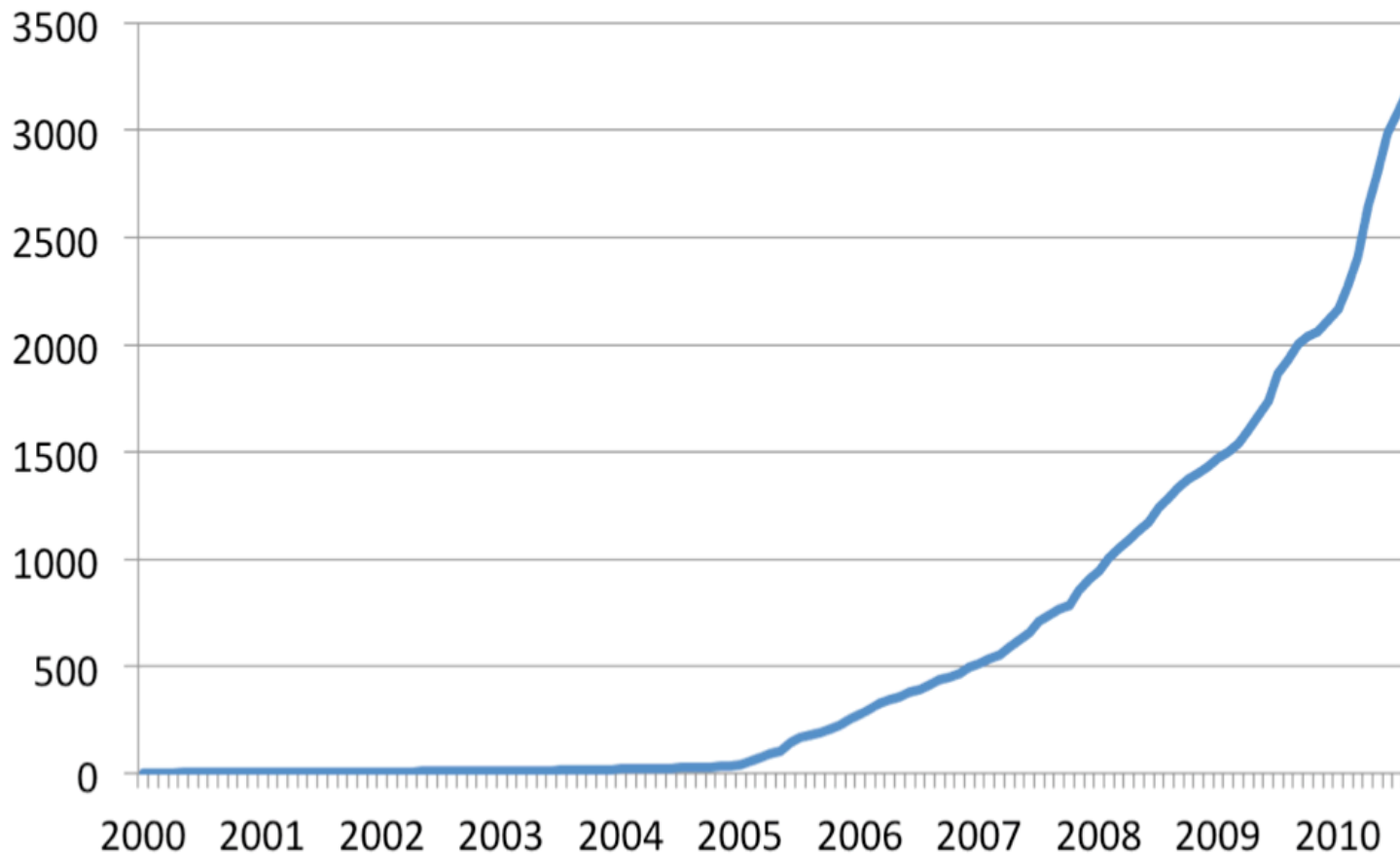
Cluj Napoca

*Calea Dorobantilor 18-20 Power  
Business Center*

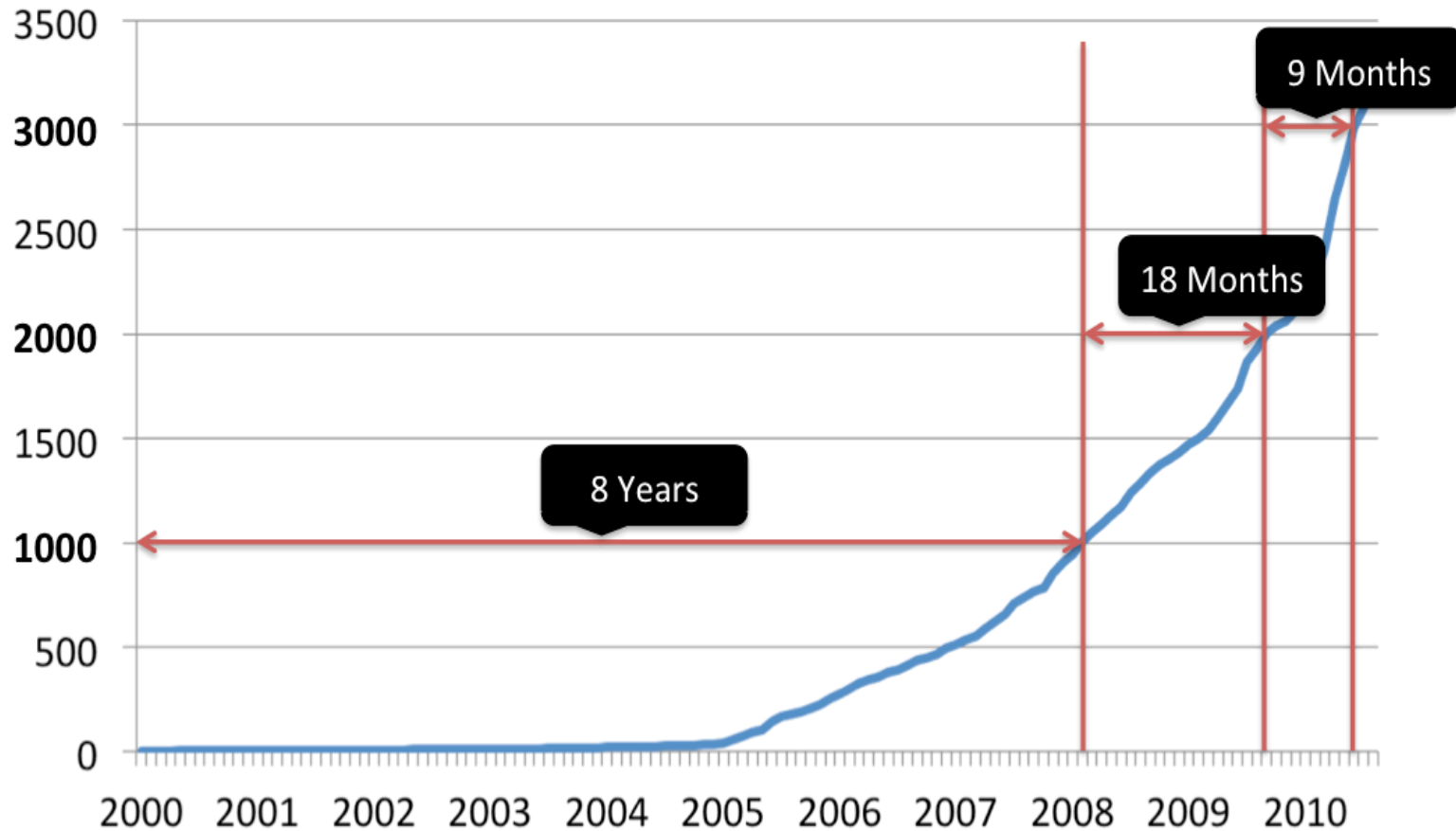
Iasi

*Sos. Pacurari 138*



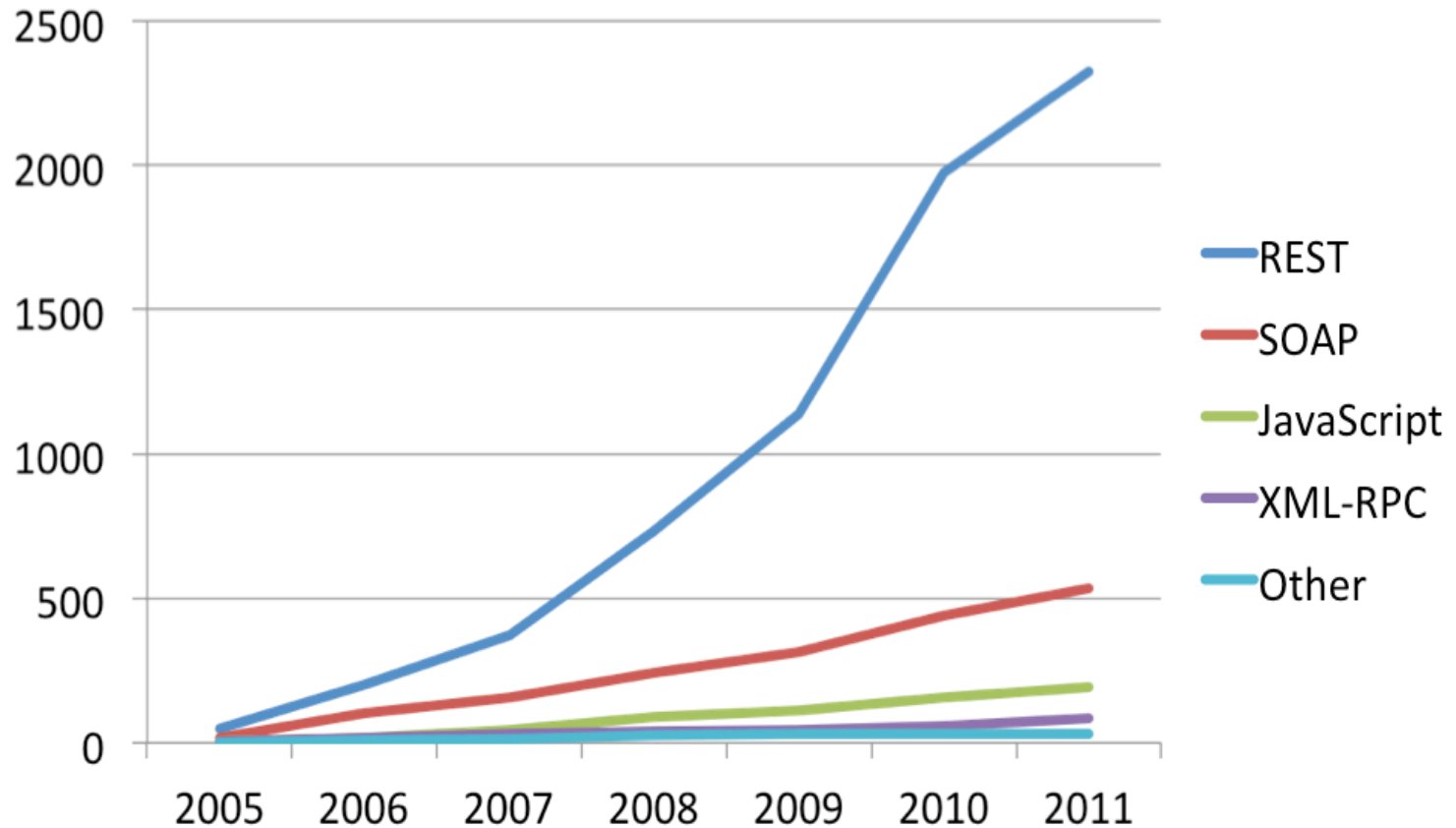


From Open APIs: State of the Market by John Musser, ProgrammableWeb



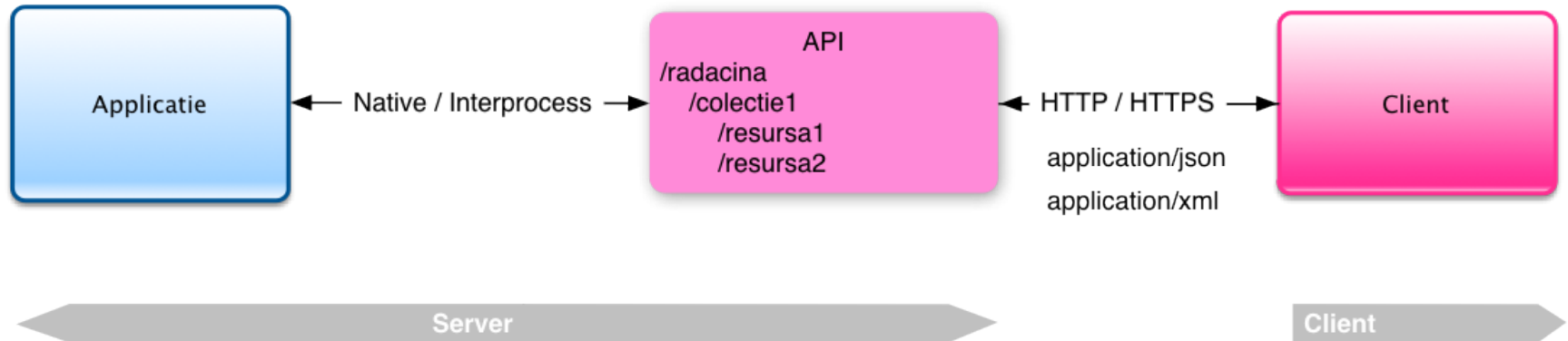
From Open APIs: State of the Market by John Musser, ProgrammableWeb

## REST vs. SOAP: Simplicity wins again



From Open APIs: State of the Market by John Musser, ProgrammableWeb

*Based on directory of 3,200 web APIs listed at ProgrammableWeb, May 2011*



- <https://api.taskmgm.com/createCompany>
- <https://api.taskmgm.com/deleteCompany>
- <https://api.taskmgm.com/getCompany>
- <https://api.taskmgm.com/getAllCompanies>
- <https://api.taskmgm.com/getAllProjects>
- <https://api.taskmgm.com/getProject>
- <https://api.taskmgm.com/createProject>
- <https://api.taskmgm.com/updateProject>
- <https://api.taskmgm.com/deleteProject>
- <https://api.taskmgm.com/createLog>
- <https://api.taskmgm.com/updateLog>
- <https://api.taskmgm.com/deleteLog>
- <https://api.taskmgm.com/getLogs>
- <https://api.taskmgm.com/getLog>

`/ {colecții}`

conțin ->

`/ {store-uri}`

care stochează ->

`/ {documente}`

care au datele propriuzise

/companies

- colectie
- lista de companii

/1234

- document
- detalii companie 1234

- <https://api.taskmgm.com/v1/companies>
- <https://api.taskmgm.com/v1/companies/1234>
- <https://api.taskmgm.com/v1/projects>
- <https://api.taskmgm.com/v1/projects/1234>
- <https://api.taskmgm.com/v1/logs>
- <https://api.taskmgm.com/v1/logs/12182>

## /projects

- colectie
- lista de proiecte

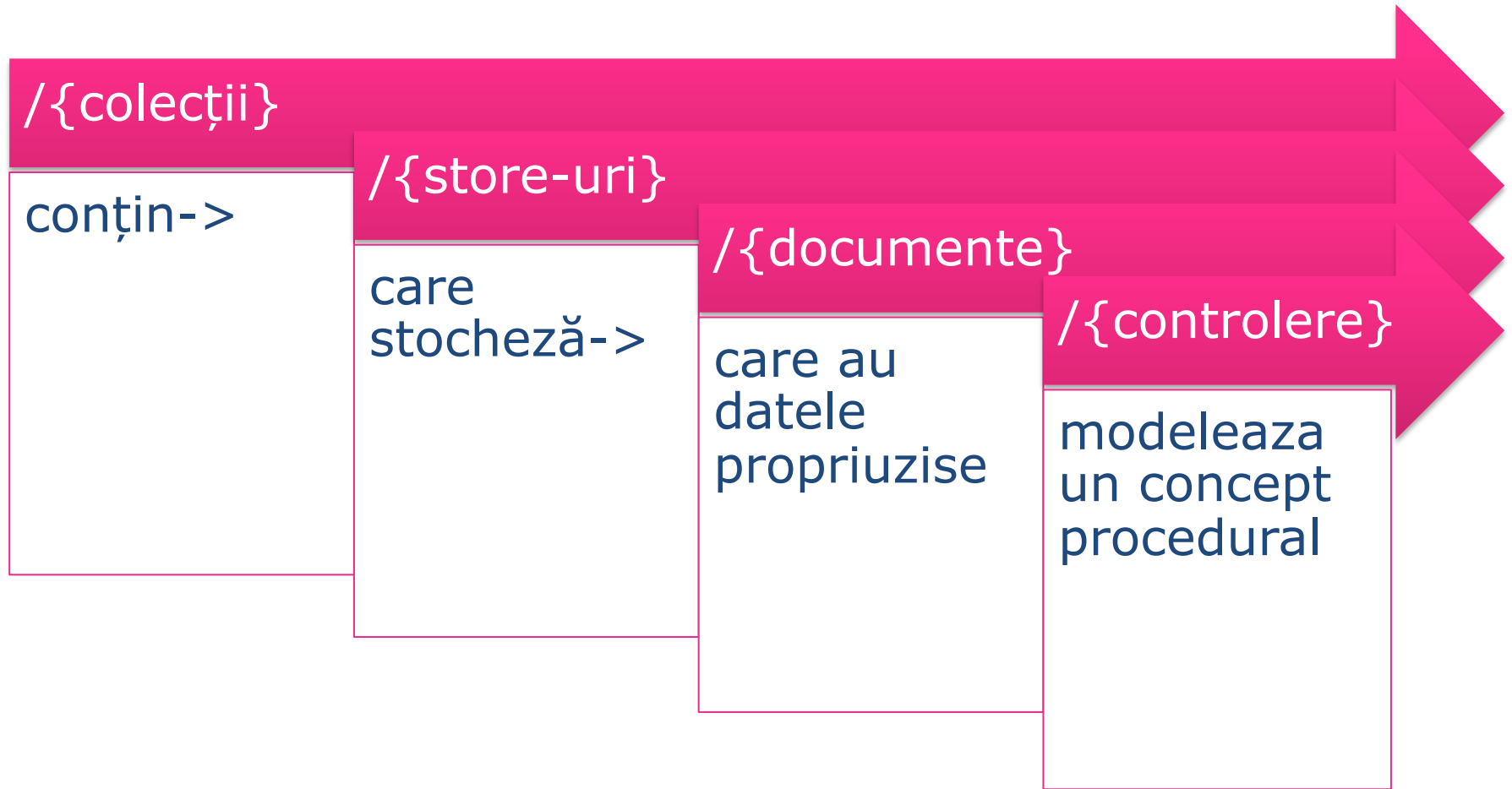
## /favorites

- un store
- returneaza tot o listă
- cu proiectele favorite

## /1234

- document
- detalii proiect favorit

- <https://api.taskmgm.com/v1/projects>
- <https://api.taskmgm.com/v1/projects/1234>
- <https://api.taskmgm.com/v1/projects/favorites>
- <https://api.taskmgm.com/v1/projects/favorites/1234>



/users

- colectie
- lista de utilizatori

/1234

- document
- detalii unui anumit user

/resetPassword

-controler  
reseteaza  
parola

- <https://api.taskmgm.com/v1/users>
- <https://api.taskmgm.com/v1/users/1234>
- <https://api.taskmgm.com/v1/users/1234/resetPassword>
- <https://api.taskmgm.com/v1/users/login>

# Operații asupra resurselor / Metode HTTP

| Resursă         | POST<br>Crează                                         | GET<br>Citește                                       | PUT<br>Actualizează                                                                         | DELETE<br>Sterge                                                                            |
|-----------------|--------------------------------------------------------|------------------------------------------------------|---------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|
| /companies      | <b>201 Created</b><br>Crează o nouă companie           | <b>200 OK</b><br>Obține lista de companii            | <b>405 Method Not Allowed</b> sau <b>200 OK</b><br>Actualizeaza toată lista de companii     | <b>405 Method Not Allowed</b> sau <b>200 OK</b><br>Nimic sau Sterge toata lista de companii |
| /companies/1234 | <b>405 Method Not Allowed</b> sau <b>404 Not Found</b> | <b>200 OK</b><br>Obține detalii despre compania 1234 | <b>200 OK</b><br>Daca exista compania 1234 actualizeaza altfel eroare: <b>404 Not Found</b> | <b>200 OK</b><br>Sterge compania 1234 sau <b>404 Not Found</b> daca nu există               |

- RFC 3986
- Syntaxa unui URI e:
  - scheme
  - "://"
  - authority
  - "/" path
  - [ "?" query ]
  - [ "#" fragment ]
- `http://example.com/v1/users#name?q=Remus`

- Folositi substantive la plural si nu verbe
  - /companies
  - /users
  - ~~/getUsers~~
  - ~~/getCompanies~~
- Nu amestecati plural si singular, nu e consistent
  - /companies
  - ~~/user~~
  - /users

## Utilizati caractere mici in URI-uri (lowercase)

- `https://api.taskmgm.com/v1/users`
  - o resursa
- `HTTP://Api.Taskmgm.com/v1/users`
  - aceiasi resursa ca mai sus
- `https://api.taskmgm.com/v1/Users`
  - o alta resursa

## De ce?

RFC 3896 defineste URI-urile case sensitive exceptind Protocol-ul si componentele host-ului

scheme "://" authority "/" path [ "?" query ] [ "#" fragment ]

Nu includeti extensie de fisiere in URI-uri

- <https://api.taskmgm.com/v1/users/1234.json>

Mecanismele HTTP pentru negocierea continutului

Headerele

- ACCEPT
- ContentType

- Implicit parte din HTTP
- header-ul: Accept

```
Accept          = "Accept" ":"  
                  #( media-range [ accept-params ] )  
media-range     = ( "*"/*"  
                  | ( type "/" "*" )  
                  | ( type "/" subtype )  
                  ) *( ";" parameter )  
accept-params   = ";" "q" "=" qvalue *( accept-extension )  
accept-extension = ";" token [ "=" ( token | quoted-string ) ]
```

**Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;**

**<https://api.taskmgm.com/v1/companies?format=json>**

- Forward slash separator (/) indica relatii hierarhice
  - <http://example.org/vinzari/2012/08/01>

- Trailing forward slash (/) nu trebuie sa termine un URI
  - ~~http://example.org/companies/1234/~~
  - http://example.org/companies/1234
- De ce?
  - Nu adauga valoare semantica
  - Poate cauza confuzie (2 uri-uri diferite trebuie sa identifice 2 resurse diferite)

Hypens (-) pot fi folosite pentru a imbunatatii  
readability-ul URI-urilor

<http://example.org/blogs/remus-pereni/entry/primul-post>

Underscore (\_) nu ar trebui folosite

Link-urile de obicei sunt afisate cu underline si  
atuncea underscore poate fi confundat

- Folosit pentru a obtine starea unei resurse
- Cererea poate contine headere dar nu body
- Cereri repetate la GET nu e voie sa aiba efecte secundare / duca la modificari de stare
- Cache-urile depind de abilitatea de a servi un raspuns fara a mai contacta serverul original

- Exact ca si GET, returneaza doar headerele fara body
- Folosit la verifica existenta unei resurse sau a metadatelor legat de ea

- Folosit pentru insert si update-uri
- Mesajul trebuie sa contina o reprezentare a resursei pe care clientul doreste sa o stocheze pe server
- Continutul mesajului poate sa nu fie identic cu cea ce ar primi de la un request GET. Poate contine doar valorile mutabile / variabile din starea resursei
- Operatiune Idempotentă

- Folosit pentru a crea o resursa noua intr-o colectie
- Folosit pentru a invoca controleri
- Permis pentru orice actiune fara legatura cu repetabilitatea sau efectele colaterale (unsafe & non-idempotent)
- Nu este garantata repetabilitatea
- A nu se folosi pentru a obtine / sterge / actualiza resurse

- Folosita pentru a sterge o resursa dintr-o colectie
- O data stearsa cu DELETE resursa nu mai trebuie sa apartina colectiei (orice GET sau HEAD pe resursa respectiva trebuie sa se termine cu 404 NOT FOUND)

- Folosita pentru a sterge o obtine lista de operatii posibile asupra unei resurse
- Allow: GET, PUT, DELETE

- Mai multe abordari
  - Versiune in path
    - <http://example.com/timetrack/v1/companies/yonder>
  - Versiune in header
- Versiune in header / parte din negocieri
  - Un URI trebuie sa identifice constant o resursa
  - Modificarea URI-ului presupune o resursa noua
  - Deci V1 & V2 denota 2 resurse diferite -> Incorect

- Ascundeti complexitate in spate la ?
  - <https://api.taskmgm.com/v1/companies?limit=10&offset=0>
- Puteti folosi limit și offset
- Să aveți valori implicite pentru ele

## Fie prin controller

<https://api.taskmgm.com/v1/companies/search>

## Ascundeti complexitate in spate la ?

<https://api.taskmgm.com/v1/companies?search=Yonder>

Http status code : 401 Unauthorized

- {
  - "developerMessage" : "Verbose, plain language description of the problem for the app developer with hints about how to fix it."
  - , "userMessage": "Pass this message on to the app user if needed."
  - , "errorCode" : 12345
  - , "more info": <http://dev.teachdogrest.com/errors/12345>
- }

# Coduri de răspuns, tratarea exceptiilor

Success

Totul a  
funcționat

Erroare  
client

Aplicatia a  
facut ceva  
greșit

Erroare  
server

Ups s-a  
intinplat  
ceva aiurea

# Coduri de răspuns, tratarea exceptiilor

- HTTP Defineste 5 categorii

| 1XX | Informational | Comunica informatii la nivel de protocol                                |
|-----|---------------|-------------------------------------------------------------------------|
| 2XX | Success       | Cererea clientului acceptata cu success                                 |
| 3XX | Redirectari   | Indica ca sunt nevoie actiuni suplimentare pentru a complecta requestul |
| 4XX | Errori client | Erroare datorata in general request-ului / Clientului                   |
| 5XX | Errori server | Erroare datorata in general serverului                                  |

# Coduri de răspuns, tratarea exceptiilor

Success

200 OK

Erroare  
client

400 Bad  
Reuest

Erroare  
server

500 Internal  
Server Error

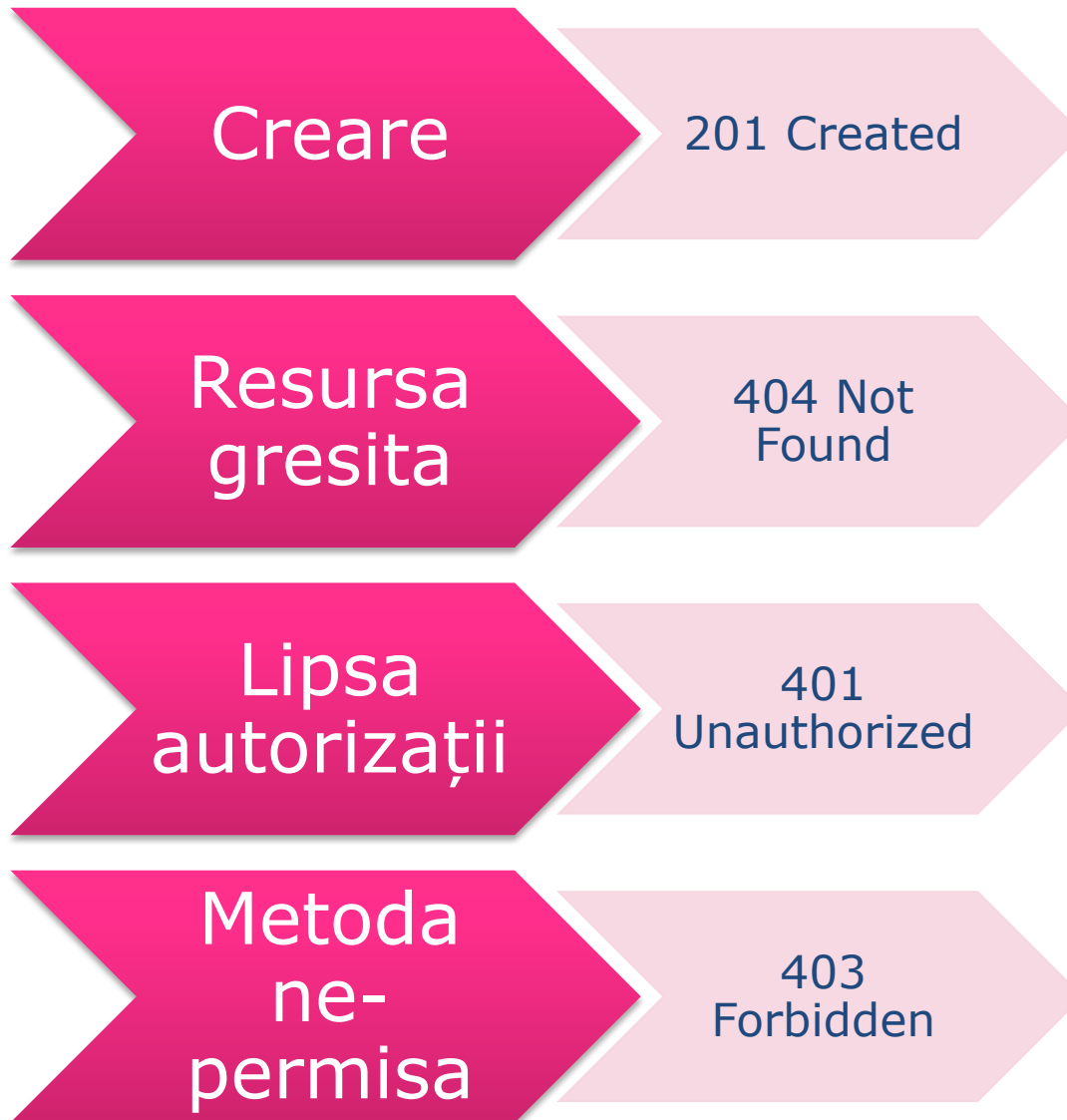
# Coduri de răspuns, tratarea exceptiilor



418

I'm a teapot

# Coduri de răspuns, tratarea exceptiilor



# Coduri de răspuns, tratarea exceptiilor

- 200 ("OK")
  - Codul pe care clientii spera sa-l vada
  - Indica success
  - In general trebuie sa includa un raspuns in body
  - Nu trebuie folosit in a comunica errori in continutul mesajului
- 201 ("Created")
  - Folosit de cite ori o colectie sau un store creaza o resursa noua bazat pe cererea clientului
  - Poate fi si raspuns al apelului unui controller in cazul in care o resursa este creata
- 202 ("Accepted")
  - Folosit pentru raspunsuri asincrone
  - O operatiune lunga a fost inceputa, pare valida dar poate inca genera errori
  - In general folosit la controlere

# Coduri de răspuns, tratarea exceptiilor

- 204 ("No Content")
  - Operatia e reusita dar nu exista continut de returnat (DELETE, de exemplu)
- 301 ("Moved Permanently")
  - Resursa s-a mutat si exista o noua locatie pentru ea
  - Locatia se trimite ca parte a Location header-ului
- 303 ("See Other")
  - Controler-ul si-a terminat treaba dar in loc sa trimita un raspuns potential nedorit poate trimite in Location un URI la o resursa care poate prezenta interes pentru clienti
- 304 ("Not Modified")
  - Similar cu 204 ("No Content") in sensul ca body-ul e gol
  - Clientul are deja cea mai recenta versiune
  - Folosit impreuna cu alte headere ce determina HTTP Conditionale

# Coduri de răspuns, tratarea exceptiilor

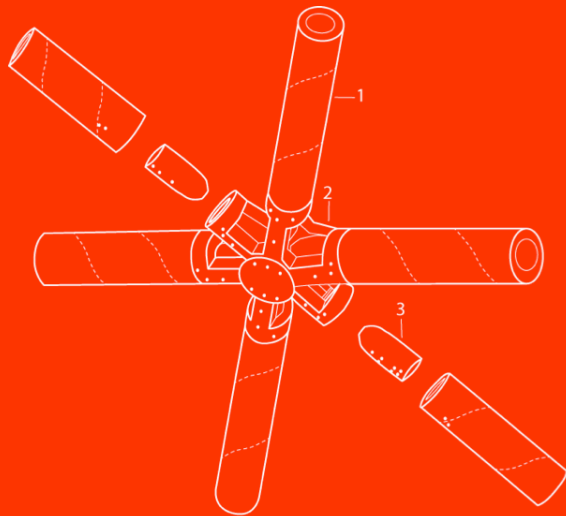
- 400 ("Bad Request")
  - Mesaj de eroare generic cind nici un alt 4xx nu este potrivit
  - Raspunsul poate contine body cu mesaj detaliat legat de eroare
- 401 ("Unauthorized")
  - Nu are autorizarea necesara sau nu a asigurat nici un fel de autorizare
- 403 ("Forbidden")
  - Indica ca request-ul e corect dar backend-ul refuza sa-l onoreze
  - Nu este o problema de autorizare (ar fi 401), poate clientul nu are permisiunea de a apela acea parte din API
- 404 ("Not Found")
  - Cererea nu poate fi mapata pe o resursa
- 405 ("Method Not Allowed")
  - Daca clientul a incercat o operatie ne-permisa (ex: DELETE pe o resursa read only)
  - Trebuie sa se trimita inapoi header-ul Allow: GET, POST cu metodele suportate

# Coduri de răspuns, tratarea exceptiilor

- 406 ("Not Acceptable")
  - Formatul cerut de client nu este suportat (Ex. clientul cere application/xml prin intermediul headerului Accept dar serverul are pregatit doar application/json)
- 409 ("Conflict")
  - Daca se incearca punerea unei resurse intr-o stare inconsistenta
  - Ex, se incearca stergerea unei colectii care nu e goala
- 415 ("Unsupported Media Type")
  - Indica ca serverul nu poate parsa cererea in formatul descris de header-ul Content-Type
  - Ex. Body-ul contine application/xml dar serverul stie sa trateze doar application/json
- 500 ("Internal Server Error")
  - Probleme pe partea de server, ex exceptii aruncate si ne tratate

## Web API Design

Crafting Interfaces that Developers Love



**apigee**

Brian Mulloy

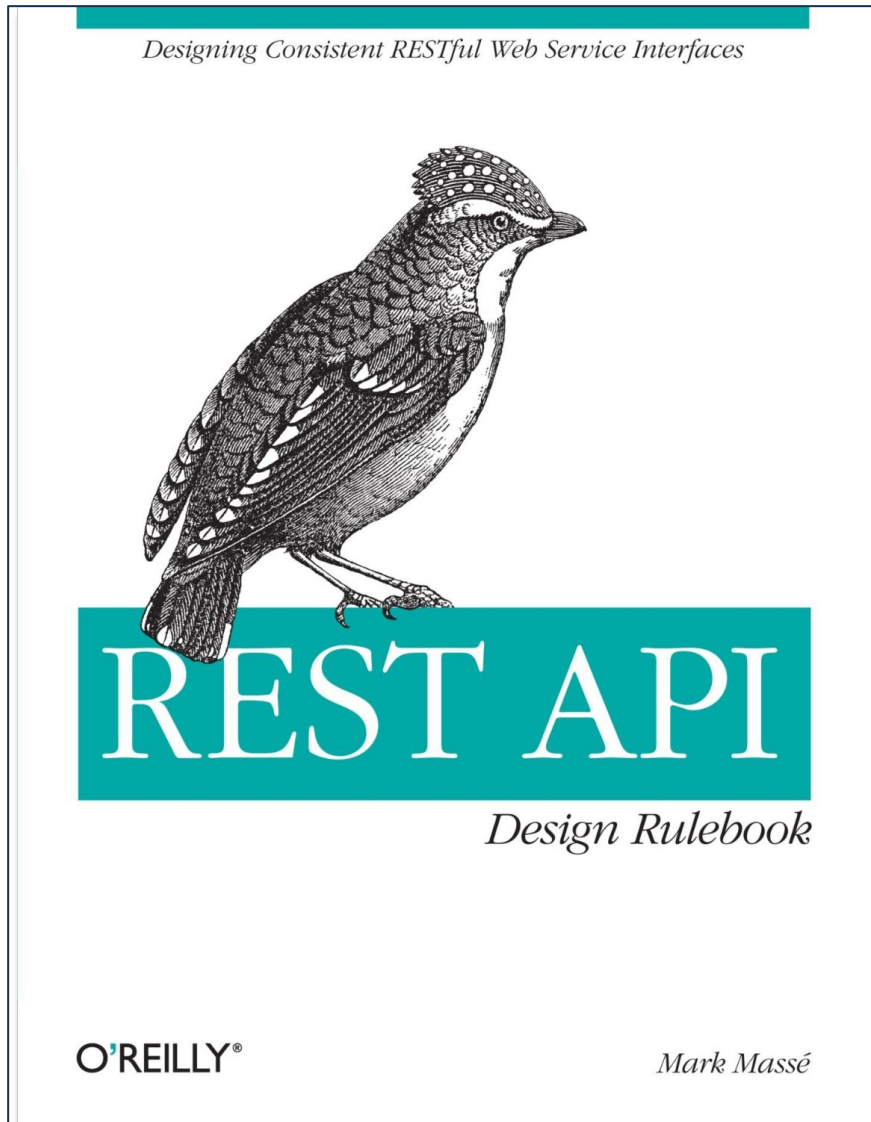
## Web API Design?

Crafting Interfaces that Developers Love

By Brian Mulloy

Ebook gratuit oferit de apigee

<http://apigee.com/about/content/web-api-design>



## **REST API Design Rulebook**

Designing Consistent RESTful  
Web Service Interfaces

By Mark Masse

Publisher: O'Reilly Media

Released: October 2011

Pages: 116

## 1. Atlassian REST API Design Guidelines version 1

<https://developer.atlassian.com/display/DOCS/Atlassian+REST+API+Design+Guidelines+version+1>

## 2. Thoughts on RESTful API Design

Lessons learnt from designing the Red Hat Enterprise Virtualization API

<https://restful-api-design.readthedocs.org/en/latest/>

# Întrebări?



photo cc by <http://www.flickr.com/photos/horiavarlan/4273168957/>

Please fill the evaluation form slide

