

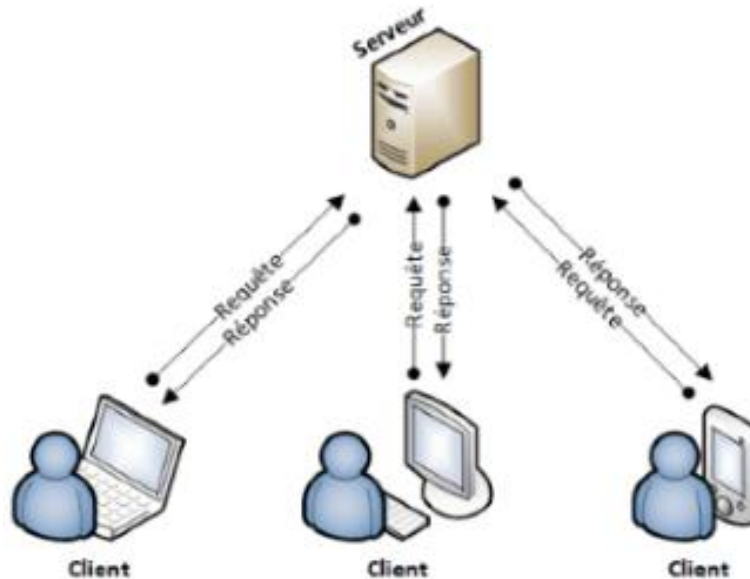


Technologies web

Http & API REST

Protocole http

- Architecture **2-tier** :

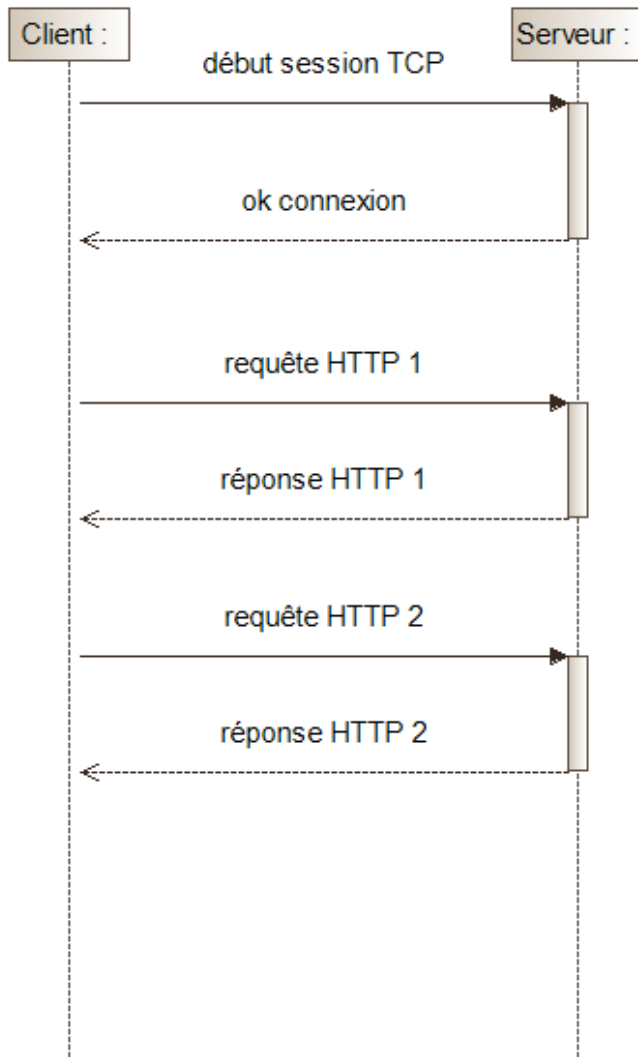


- Protocol HTTP (Hypertext Transfer Protocol) :
- Protocole d'échange HyperText (HTML : HyperText Markup Language)
- Communication normée avec **requête<->réponse**
- Développé pour le Web en 1996
- Dernière version : HTTP/3, juin 2022



- **RFC : Request For Comments :**
 - Série de document décrivant les **spécifications techniques** des technologies de l'internet
 - [HTTP = 34 RFC](#)
-
- **Internet Engineering Task Force**
 - Élabore les standards Internet
 - Auteur des RFC





- **GET :**

→ Récupère des données à partir d'un serveur

- **POST:**

→ Envoie des données sur un serveur

- **PUT:**

→ Met à jour des données sur un serveur

- **DELETE :**

→supprime des données d'un serveur

- **HEAD :**
 - Renvoie uniquement l'en-tête d'accès à une ressource
- **OPTIONS :**
 - Indique quelles sont les méthodes HTTP utilisables (options de communication)
- **PATCH :**
 - Mise à jour partielle d'une ressource
- **CONNECT :**
 - Établit un tunnel de communication bi-directionnel entre client / serveur
- **TRACE :**
 - Demande au serveur de renvoyer ce qui a été reçu

- Exemple - première page web mondiale :

<http://info.cern.ch/hypertext/WWW/TheProject.html>

Méthode	Chemin vers la ressource	Version
GET	/hypertext/WWW/TheProject.html	HTTP/1.1
Host: info.cern.ch Accept: text/html, application/xhtml+xml, text/css Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3		
En-tête		

Version Code réponse + message

HTTP/1.1 200 OK

```
Date: Sat, 16 Jul 2022 09:15:48 GMT
Server: Apache
Last-Modified: Thu, 03 Dec 1992 08:37:20 GMT
ETag: "8a9-291e721905000"
Accept-Ranges: bytes
Content-Length: 2217
Connection: close
Content-Type: text/html
```

En-tête

```
<HEADER> <TITLE>The World Wide Web project</TITLE>
<NEXTID N="55"> </HEADER> <BODY>
```

...

Corps de la réponse

- Liste complète des codes :
<https://developer.mozilla.org/fr/docs/Web/HTTP/Status>

Catégorie de réponse	Code	Message	Descriptif
Informatif	100	Continue	Tout est ok pour le moment, le client peut continuer sa requête
Succès	200	OK	Requête réussie !
	201	Created	Requête réussie, nouvelle ressource créée
	202	Accepted	Requête reçue mais pas encore traitée
<u>Redirection</u>	301	Moved permanently	Document déplacé définitivement
	302	Found	Document déplacé temporairement

Catégorie de réponse	Code	Message	Descriptif
Erreur client	400	Bad request	Syntaxe de la requête invalide
	401	Unauthorized	Le client doit s'authentifier
	403	Forbidden	Le client n'a pas le droit d'accès
	404	Not found	Un classique qu'on ne présente plus
	405	Method not allowed	La méthode HTTP n'est pas acceptée
Erreur serveur	500	Internal server error	
	501	Method not implemented	Méthode de requête non supportée
	502	Bad gateway	Le serveur passerelle a reçu une réponse invalide d'un autre

- Logiciel



Insomnia



Postman

- Ligne de commande



- Extension VSCode :



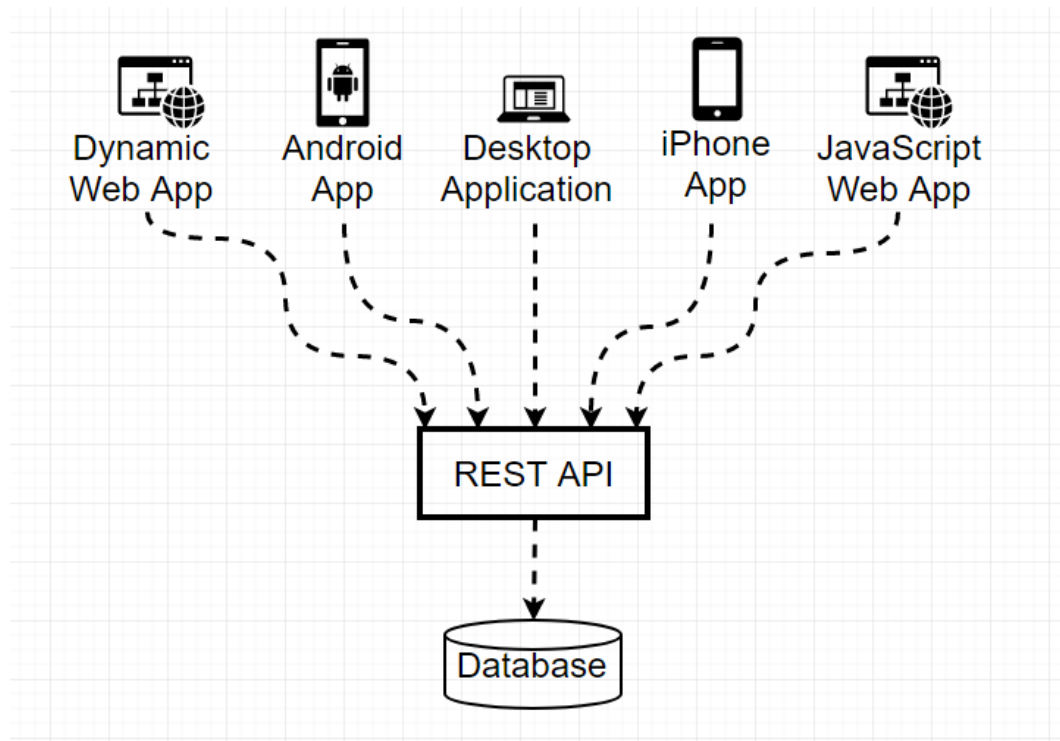
- Extension navigateur :



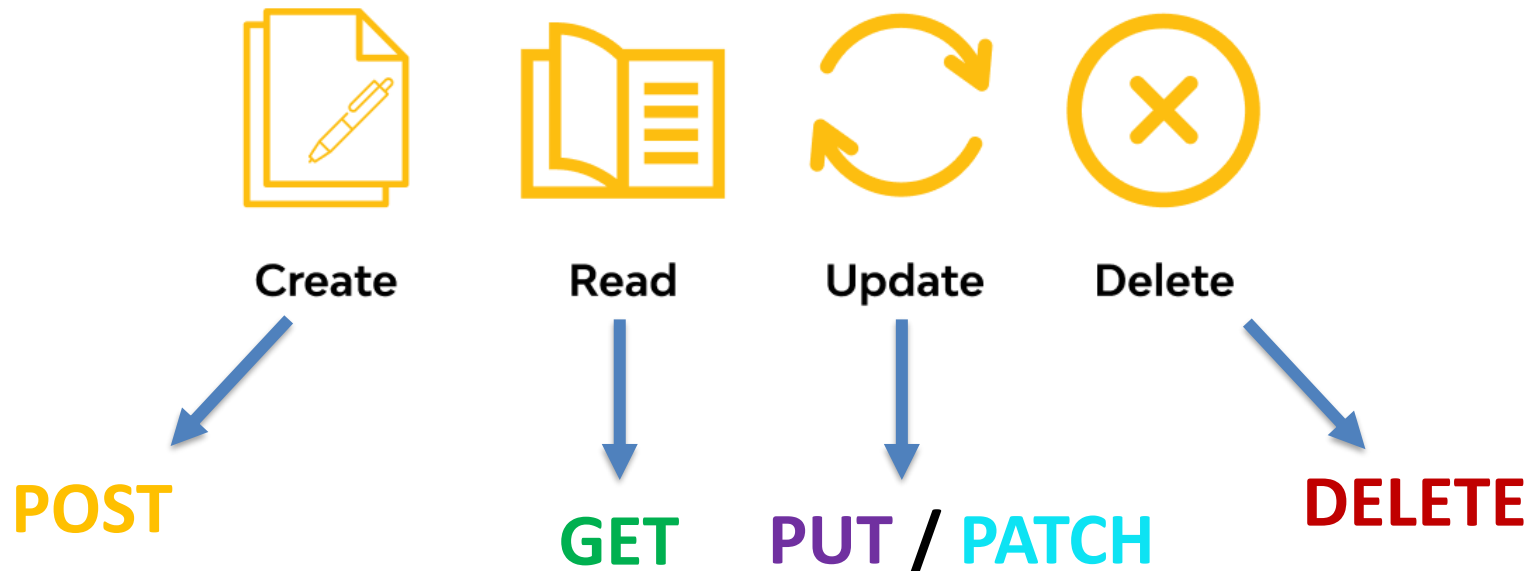
Chrome - Advanced Rest Client

API Rest

- **Rest : Representational State Transfer**
- Architecture logicielle exploitant le protocole HTTP
- Requêtes **HTTP → API**



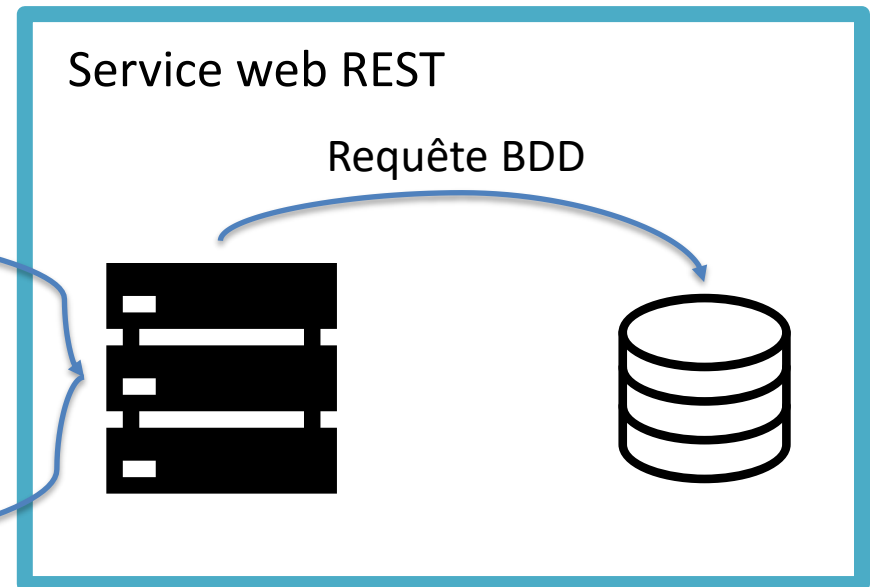
- Exemple de service web :
- Application de **gestion de recettes** (style Marmiton)
- Fonctionnalités : **CRUD**



- Cas d'utilisation :
- Création d'une recette

API End-point (URL)

Requête : **POST /recipes**



Réponse



Les échanges se font via des données **JSON**

- Exemple de JSON pouvant être envoyé :

```
{
  "name": "Tarte aux poireaux",
  "cookingDuration": 60,
  "personsNumbers": 4,
  "ingredients": [
    {
      "name": "farine",
      "quantity": 250,
      "quantityUnit": "g"
    },
    {
      "name": "Beurre",
      "quantity": 140,
      "quantityUnit": "g"
    },
    {
      "name": "Oeuf",
      "quantity": 3
    },
    {
      "name": "Poireau",
      "quantity": 2
    }
  ],
  "recipeSteps": [
    "Préchauffer le four à 210°C (thermostat 7).",
    "Faire la pâte à tarte : malaxer le beurre et la farine, l'eau, étaler puis mettre dans le plat."
  ]
}
```


- Les endpoints sont préfixés de « /api »

✓ /api/recettes

✗ /recettes

- **endpoints** = noms communs au pluriel **pas un verbe**

→ c'est la **méthode HTTP** qui indique l'action

✓ Requête POST sur /api/recettes

✗ Requête sur /api/recettes/create

- Gestion des erreurs :

→ Utiliser les bons codes http en réponse serveur

400 : les données envoyées par le client n'ont pas pu être validées

401 : l'utilisateur n'est pas autorisé à accéder à la ressource et non authentifié

403 : l'utilisateur n'est pas autorisé à la ressource mais déjà authentifié

404 : ressource non trouvée

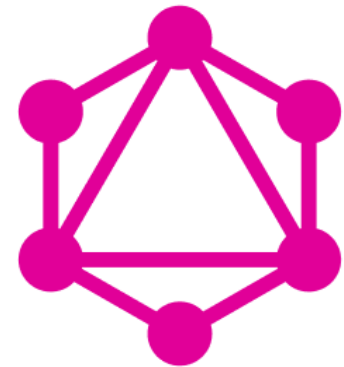
- Sécuriser l'accès aux données :



- SOAP : Simple Object Access Protocol
- Utilise la normalisation HTTP pour le requêtage
- Échanges basés sur des **fichiers XML**



- GraphQL : **G**raph **Q**uery **L**anguage
- Langage de requêtage plus poussé
- Le client indique plus précisément la donnée à récupérer





Agence nationale pour la formation professionnelle des adultes
Établissement public à caractère industriel et commercial
Tour Cityscope
3 rue Franklin, 93100 Montreuil
824 228 142 RCS BOBIGNY