

# **R209**

## **Initiation au développement web**

# Prérequis



**R107**

Fondamentaux  
de la programmation



**R109**

Introduction aux  
technologies Web



**R207**

Sources de données



**R208**

Analyse et traitement  
de données  
structurées

# Contenus



## Mise en forme de pages Web

balises HTML avancées ;

- structure d'une page avec son DOM ;
- CSS avancé ou Framework ;
- initiation au dynamisme côté client

failles XSS, XSS stockée, injections SQL



## Sensibilisation à la sécurisation de sites



Des  
compétences  
transversales

## Interaction client-serveur



Interrogation d'un SGBD ou d'une API

## Scripts côté serveur.

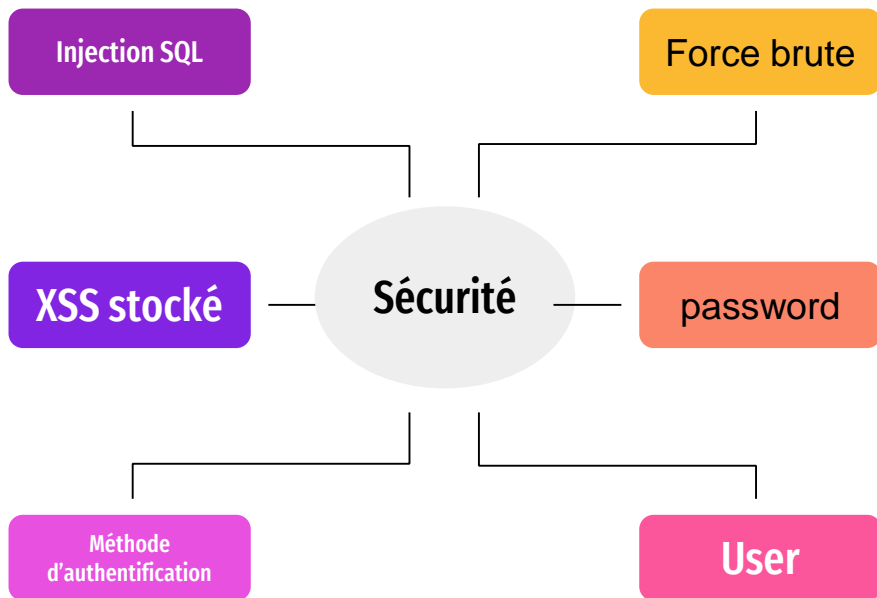


# Quelques failles de sécurité

Groupe de méthodes d'exploitation de faille de sécurité d'une application interagissant avec une base de données. Elle permet d'injecter dans la requête SQL en cours un morceau de requête non prévu par le système et pouvant compromettre la sécurité.

Le cross-site scripting est un type de faille de sécurité des sites web permettant d'injecter du contenu dans une page, provoquant ainsi des actions sur les navigateurs web visitant la page

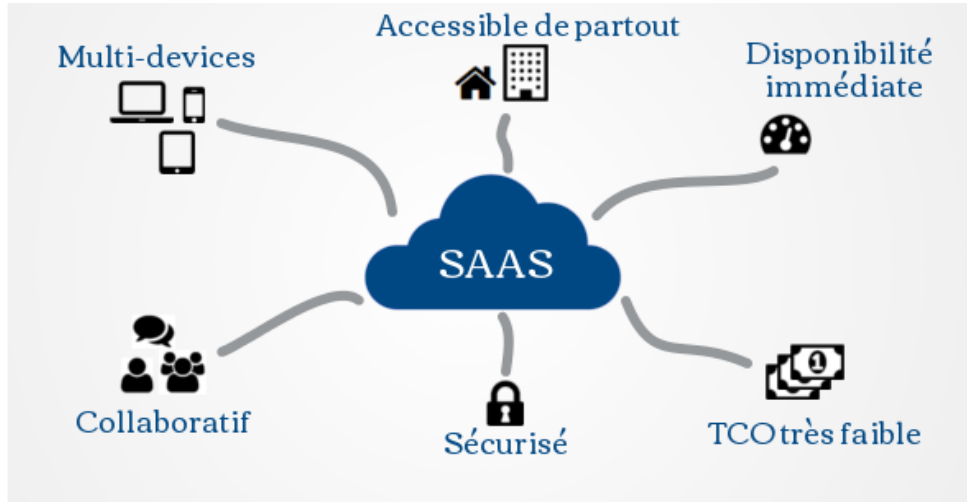
Stratégie de sécurité



L'attaque par force brute est une méthode utilisée en cryptanalyse pour trouver un mot de passe ou une clé. Il s'agit de tester, une à une, toutes les combinaisons possibles. Cette méthode est en général considérée comme la plus simple concevable

Faiblesse des passwords

L'utilisateur de l'application



## Projet en mode Saas

Le **mode Saas** est la mise à disposition d'un logiciel accessible aux utilisateurs via internet.

Aucune installation sur les serveurs de l'entreprise cliente n'est requise. Chaque utilisateur dispose d'un compte, avec des niveaux de droit variables, lui permettant d'accéder au logiciel.

## AVANT-PROPOS

Face à d'éventuels effets dans le cyberspace liés au conflit en cours entre l'Ukraine et la Russie,

l'Agence nationale de la sécurité des systèmes d'information (ANSSI) préconise la mise en œuvre de 5

mesures préventives prioritaires :

- renforcer l'authentification sur les systèmes d'information ;
- accroître la supervision de sécurité ;
- sauvegarder hors-ligne les données et les applications critiques ;
- établir une liste priorisée des services numériques critiques de l'entité ;
- s'assurer de l'existence d'un dispositif de gestion de crise adapté à une cyberattaque.

Ces mesures prioritaires de cybersécurité sont essentielles et leur mise en œuvre à court terme

permet de limiter la probabilité d'une cyberattaque ainsi que ses potentiels effets. Pour être

pleinement efficaces, elles doivent cependant s'inscrire dans une démarche de cybersécurité globale

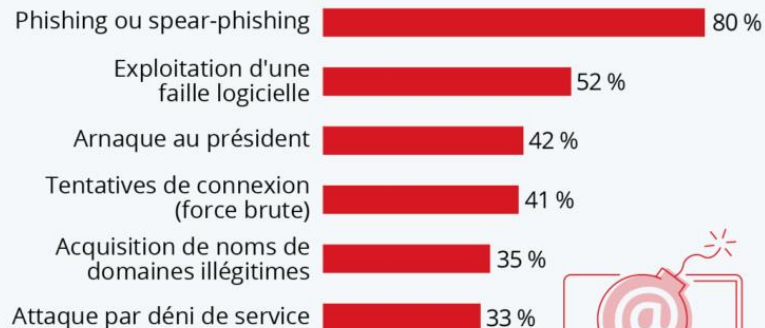
et de long terme.

# Texte de l'ANSSI

Agence nationale de sécurité des systèmes  
d'information.  
Article de février 2022

# Les cyberattaques les plus courantes contre les entreprises

Types d'attaques les plus courants constatés par les entreprises françaises en 2020 \*



## Principales conséquences des attaques :

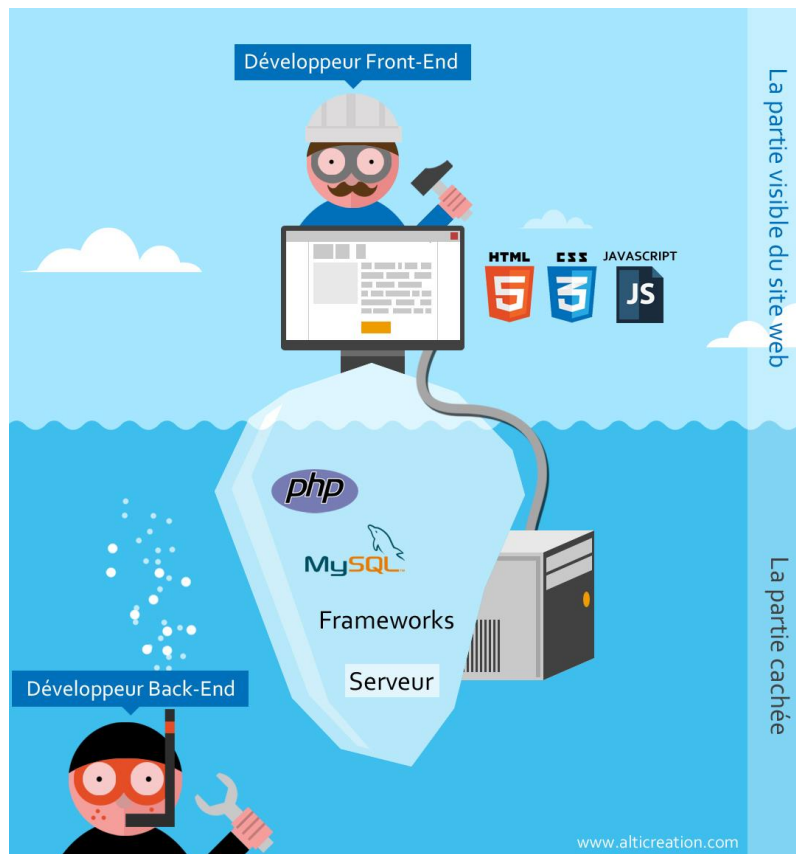


\* Plusieurs réponses possibles, sélection des plus fréquentes. Les entreprises ciblées ayant répondu à l'enquête ont subi en moyenne 3,6 attaques et 2,3 conséquences.

Sources : CESIN, OpinionWay



## Chiffres 2020



# Le développement Web



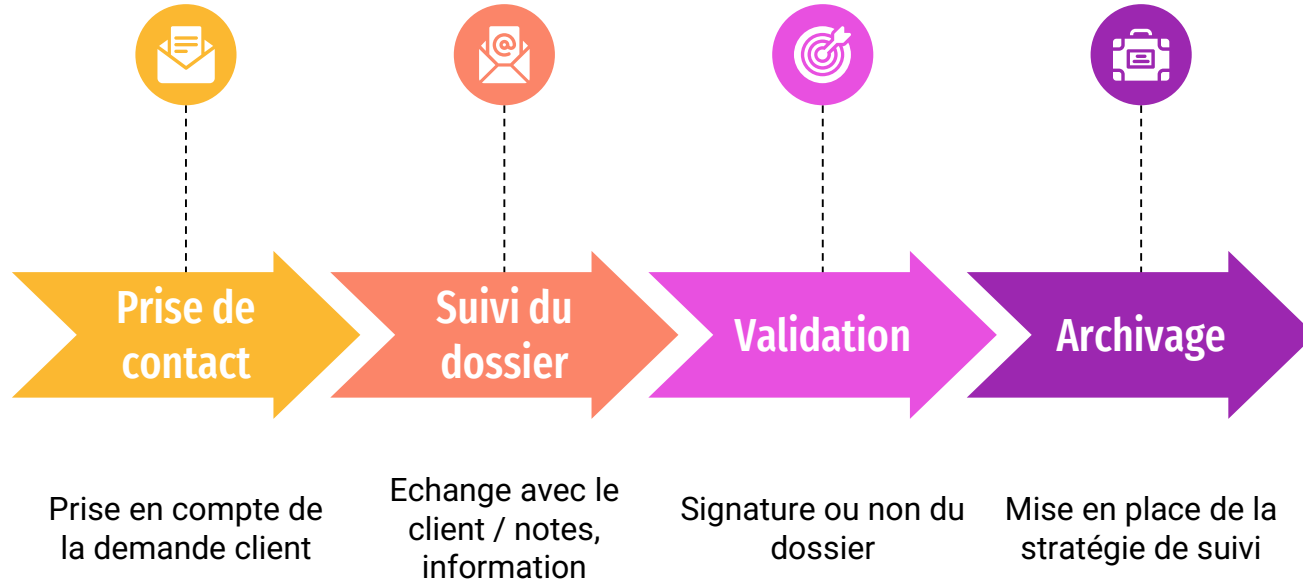
# Difficultés

- Transversalité des savoirs
- Besoins des acquis
- Mise en place de l'environnement de travail qui peut être chronophage

# Notre projet : un outil de GRC

La gestion de la relation client, ou gestion des relations avec la clientèle, est l'ensemble des outils et techniques destinés à tenir compte des souhaits et des attentes des clients et des prospects, afin de les satisfaire et de les fidéliser en leur offrant ou proposant des services.

# User story



# User story

Prise de contact



## formulaire

Créer un utilisateur  
Via un formulaire

WEAKNESSES



## Suivi du dossier

Côté entreprise :  
Intervenir sur le profil  
utilisateur.  
Côté user : suivre son  
dossier

OPPORTUNITIES



## Validation

Faire l'action de  
validation ou pas

THREATS

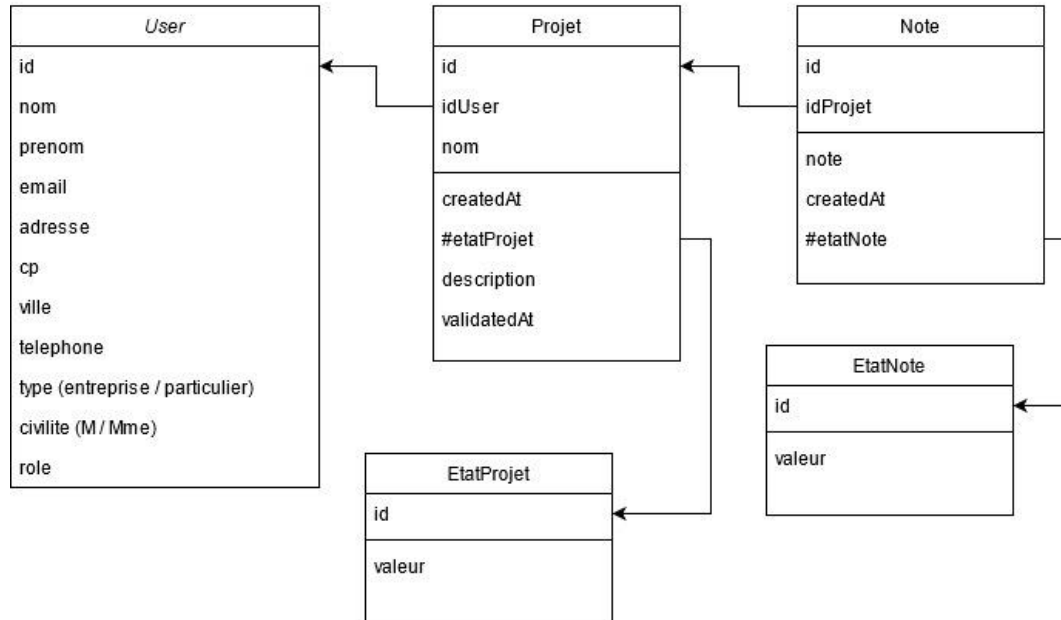


## Archivage

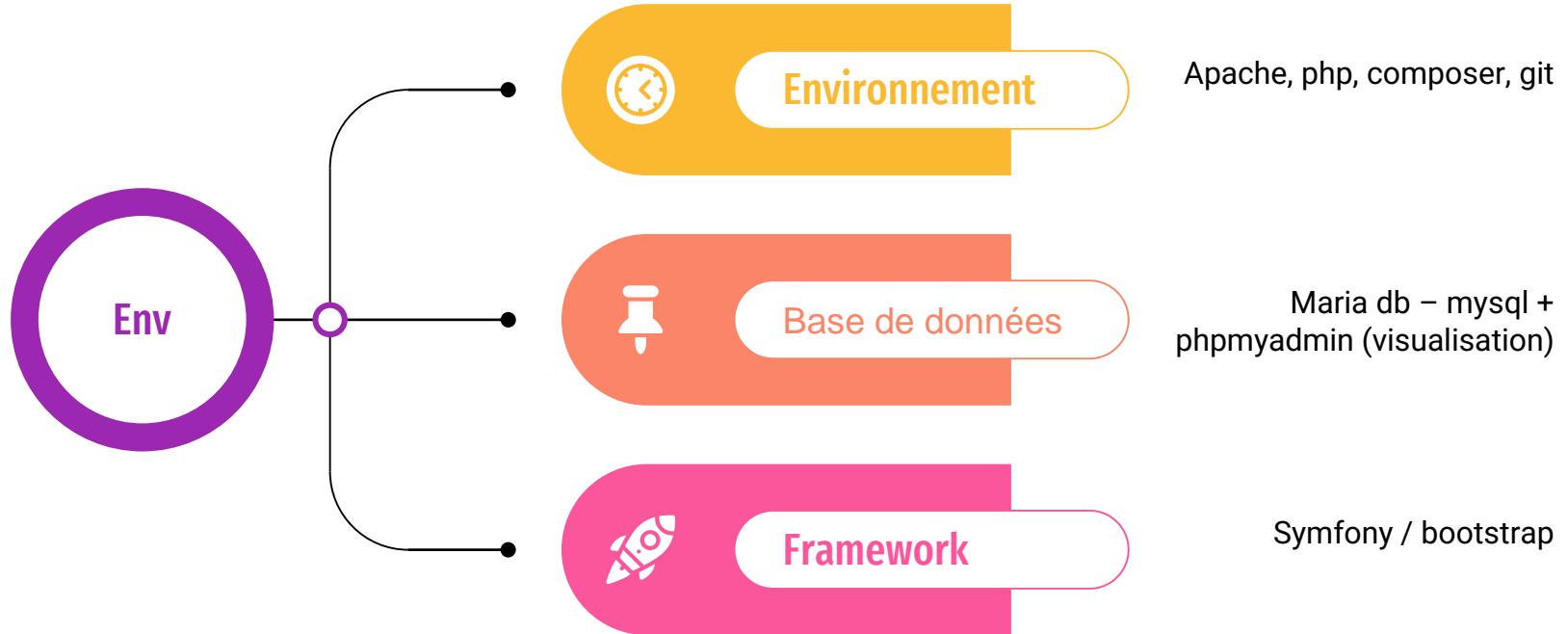
Quoi faire ?  
Stratégie  
commerciale, Api, ...

# Modèle du projet

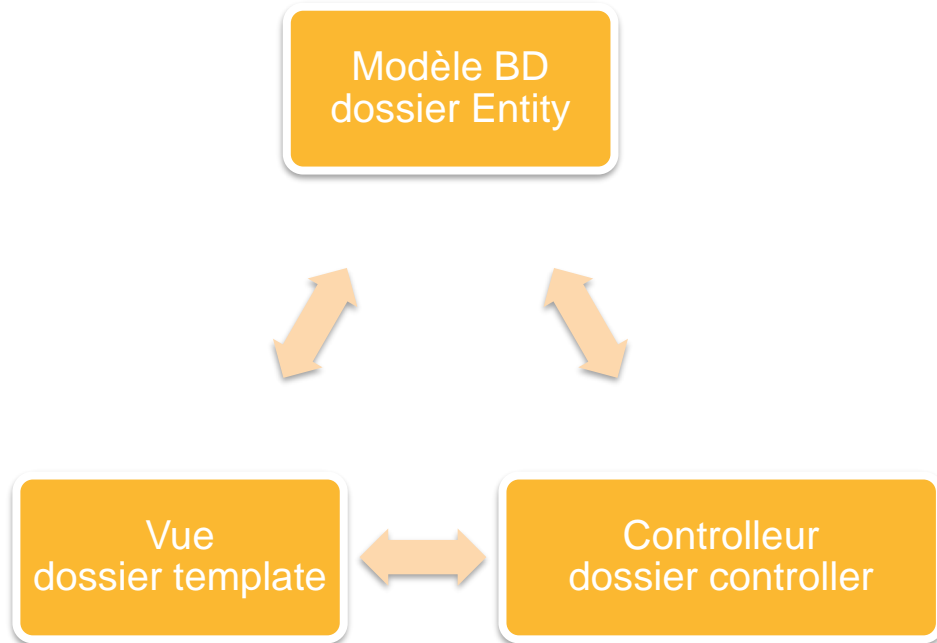
## Modèle R209



# Environnement Technique



# Modèle MVC



# Démarrage projet symfony

Symfony new grc --full

Création du projet



Symfony console make:user

Créer la table



Liens avec la BD



- Avoir un user qui a les droits sur une table grc
- - modifier le fichier .env

Symfony console  
make:registration-form

La page de register



L'authentification



Symfony console make:auth



# Arborescence

src / :

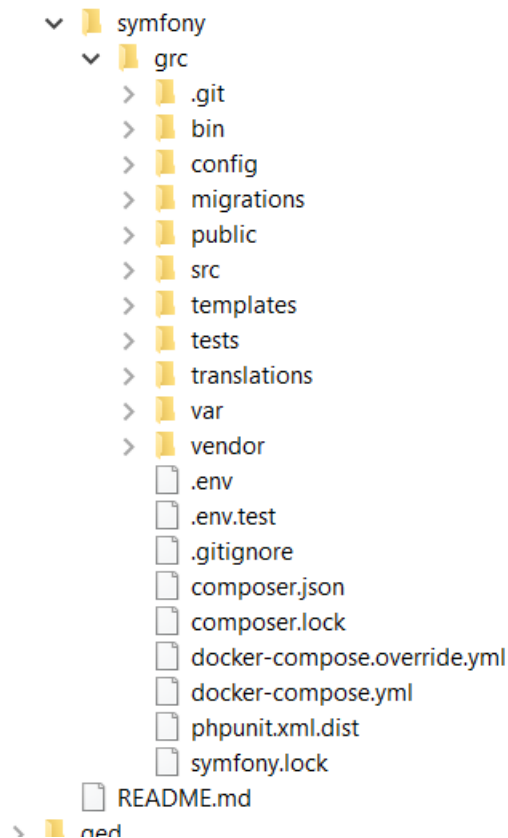
- Controller            tous les controleurs
- Entity                toutes les tables sous la forme Nom.php
- Repository            un fichier par table pour créer des requêtes

templates/ :

- Liste des dossiers (un par controller) -> fichier twig (front)
- Fichier base.html.twig qui va contenir notre template et le menu

.env :

- Fichier qui permet d'identifier le user et la base de données



# Support pour ce module



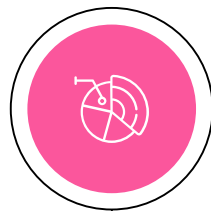
Documentation officielle

Symfony, bootstrap ...



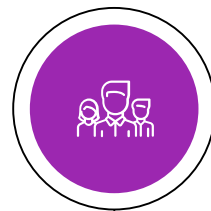
Code sur git

Code sur github



Wiki sur git

Wiki d'aide sur git



google

...

# Pour la bonne réussite de ce module

## Les prérequis

Avoir les bases  
Ne pas prendre du retard



Comprendre le  
fonctionnement de  
symfony

Routage,

## Env de développement stable

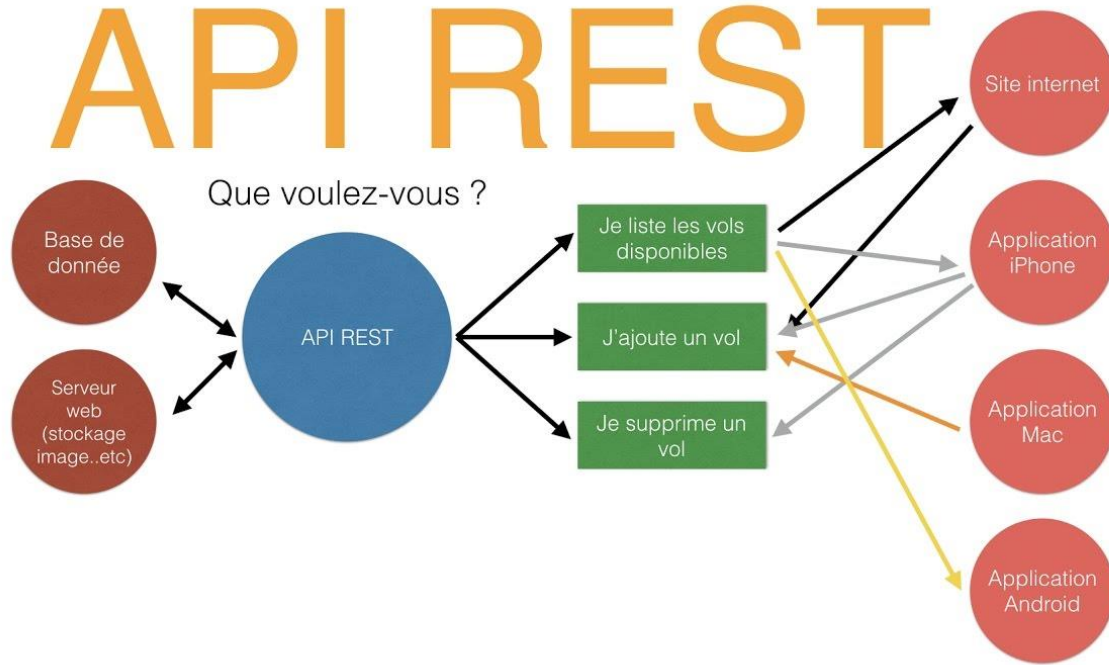
Env portable  
Facilement  
reprogrammable



## Modèle – base de données

Créer le modèle  
rapidement,  
correctement et  
définitivement

# Créer une API



# La sérialisation

## SÉRIALISATION DES DONNÉES

### VARIABLE

```
-----  
$posts = [...]  
$post = new Post()
```



### REPRESENTATION

```
-----  
N'importe quel format de  
représentation textuelle  
(JSON, XML, CSV,  
YML ...)
```

# Normalisation / Sérialisation

- La normalisation permet de transformer des objets en tableau associatif
- La sérialisation permet de transformer des tableaux associatifs en chaîne de caractère (XML, json, csv, ...)

# Que souhaitons nous

Nous souhaitons que lorsque un navigateur ou un code quelconque se rend à une adresse précise, la plateforme va chercher des informations à la base de données qui lui renvoie des données interprétables et compréhensibles facilement.

Obtenir un format json pour remplacer nos objets php qui arrivent sous forme de tableaux associatifs.

## **Idée générale :**

Lorsque l'on reçoit une requête HTTP , on répond par du texte. (json, csv, xml, ...)

# Exemple : liste des users de mon application

L'adresse : <http://51.222.75.212:8005/manager/api> renvoie la liste des user de mon application

```
{
  "0": {
    "id": 1,
    "email": "david@david.fr",
    "userIdentifier": "david@david.fr",
    "roles": {
      "0": "ROLE_USER",
      "password": "$2y$13$DUBfx9yXaTxr2Ozf2zIW..p4TRYztFuZFojTomyQVrAv0AGRPKs4y"
    }
  },
  "1": {
    "id": 2,
    "email": "david2@david.fr",
    "userIdentifier": "david2@david.fr",
    "roles": {
      "0": "ROLE_USER",
      "password": "$2y$13$xF23ZTAekX3BDQT2sENLFO0rshhpF4zxcAGK3upag9rA5.2ir5qa"
    }
  }
}
```



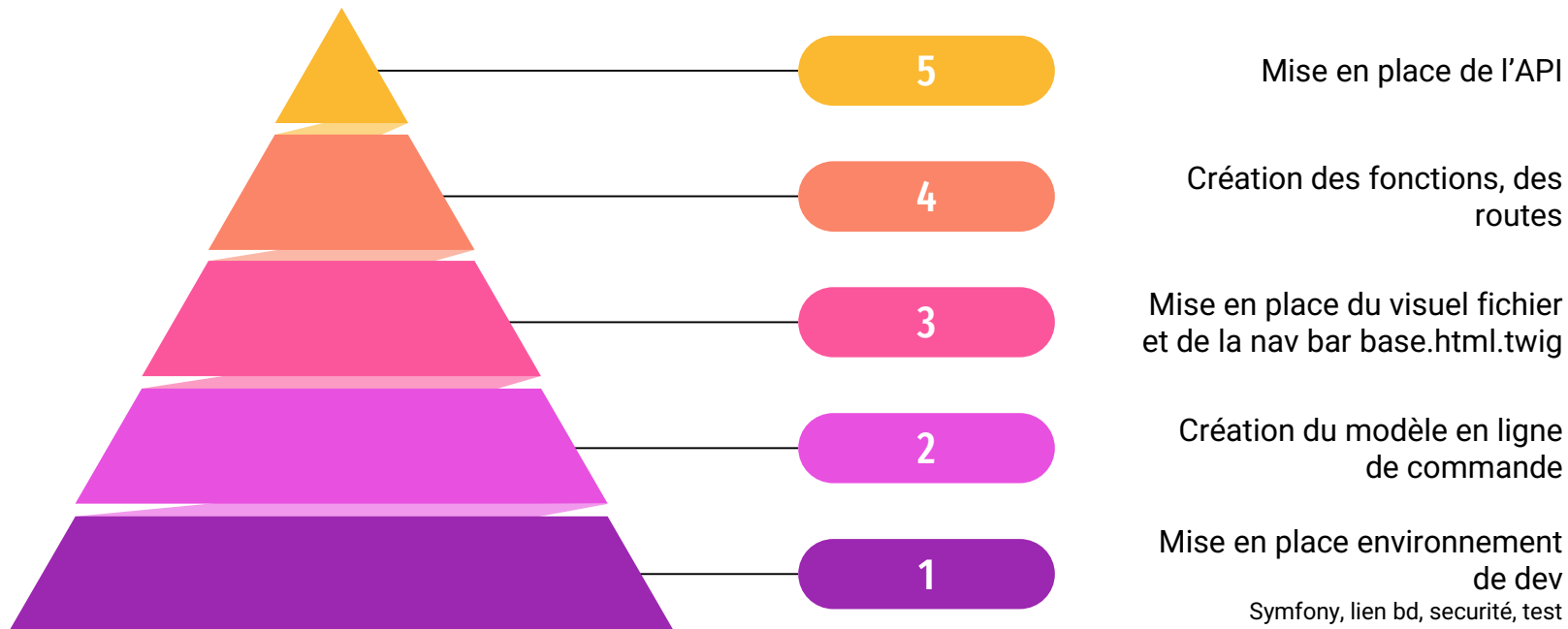
# Exemple : liste des users de mon application

## Code de la route /manager/api

```
1 <?php
2
3 namespace App\Controller;
4
5 use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
6 use Symfony\Component\HttpFoundation\Response;
7 use Symfony\Component\Routing\Annotation\Route;
8 use Sensio\Bundle\FrameworkExtraBundle\Configuration\Method;
9 use App\Entity\User;
10 use App\Repository\UserRepository;
11 use Symfony\Component\Serializer\SerializerInterface;
12 use Symfony\Component\Serializer\Serializer;
13
14
15 class ManagerController extends AbstractController
16 {
17
```

```
/**
 * @Route("/manager/api", name="app_manager_api")
 * @Method({"GET", "POST"})
 */
public function api(UserRepository $userRepo): Response
{
    $recupUserObject = $userRepo->findAll();
    $response = $this->json($recupUserObject, 200, []);
    return $response;
}
```

# Feuille de route



# Liste des commandes

## Liste des commandes de départ

Symfony new grc -full  
Modification du fichier.env  
Symfony console doctrine:database:create  
Symfony console make:user  
Symfony console make:auth  
Symfony console make:registration-form  
Symfony console serve -d  
Test d'accès via un navigateur

## Liste des commandes

Symfony console make:controller  
Symfony console make:entity  
Symfony console make:migration  
Symfony console doctrine:migrations:migrate