

**Rédigé par : OUKHENNICHE Abdelkrim**

**Projet CareMe**

**Article N°211**

**Spécifications Techniques**

**Alger le : 08/05/2017**

**Alger le 16/04/2017**

**Article N°**

**Nom livrable**

**Rédaction : Nom rédacteur**

**Rédaction : OUKHENNICHE Abdelkrim**

**Alger le 16/04/2017**

**Article N°201**

**Charte de codage**

# Table des versions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Rédacteur | Description | Date | Validation |
| 1.0 | OUKENICHE Abedelkrim | Création du document | 29/04/2017 | IHADADANE Thinhinane  Taklit Zina |
| 1..1 | OUKENICHE Abedelkrim | Modification de document | 3/05/2017 | TAKLIT Zina  OUKENICHE Abedelkrim |
| 1..2 | OUKENICHE Abedelkrim | Modification de document | 5/05/2017 |  |
| 1.3 | OUKENICHE Abedelkrim | Modifiation de document | 6/05/2017 |  |
| 1.4 | OUKENICHE Abedelkrim | Modification de document et mise page | 8/05/2017 |  |
| 1.5 | TAKLIT Zina | Mise en page | 25/05/2017 | TAKLIT Zina |

Table de matière

[Table des versions 2](#_Toc483511422)

[1. Introduction : 4](#_Toc483511423)

[2. Dictionnaire : 4](#_Toc483511424)

[HTML5 4](#_Toc483511425)

[CSS3 4](#_Toc483511426)

[JAVASCRIPT 4](#_Toc483511427)

[MySQL 4](#_Toc483511428)

[PHP 4](#_Toc483511429)

[3. Spécifications techniques 4](#_Toc483511430)

[3.2 Spécifications techniques coté Client 4](#_Toc483511431)

[3.2 Spécifications techniques coté Expérience Utilisateur 4](#_Toc483511432)

[3.3 Spécifications techniques coté SERVEUR 5](#_Toc483511433)

[3.4 Spécifications techniques coté SECURITE 5](#_Toc483511434)

[3.5 Spécifications techniques coté Performance : 6](#_Toc483511435)

[4. Diagramme de composant 6](#_Toc483511436)

[4.1 Description 6](#_Toc483511437)

[4.2 Diagramme de composant 7](#_Toc483511438)

[5. Diagramme de déploiement 8](#_Toc483511439)

# Introduction :

L’objectif du présent document est de décrire de façon exhaustive les spécifications techniques du projet CareMe.

# Dictionnaire :

La terminologie ainsi que les sigles utilisés dans ce document sont explicité dans ce paragraphe

HTML5**:**(HyperText Markup Language 5) est la dernière révision majeure du HTML (format de données conçu pour représenter les pages web).

CSS3  : Utilisé pour la partie design des pages WEB

JAVASCRIPT : Est un langage de programmation de scripts principalement employé dans les pages web interactives.

MySQL: Un système de gestion de base de données.

PHP : HyperText Preprocessor Langage de programmation web dynamique

# Spécifications techniques

## Spécifications techniques coté Client

1. Le contenu des pages web doit être en HTML5 ;
2. La partie design des pages web doit être en CSS3;
3. La partie dynamique client doit être développée en JavaScript ;
4. Le système doit utiliser la bibliothèque Jquery, comme bibliothèque Js ;
5. Le système doit utiliser la bibliothèque Bootstrap, comme bibliothèque CSS ;
6. Le système doit changer l’url de la page courante quand une requête Ajax est envoyée pour le chargement de la nouvelle page (WEB2.0) ;
7. Les requêtes Ajax doivent être envoyées en utilisant les méthodes offertes par Jquery ($.ajax,$.post,$.get,$.getJSON...) .

## Spécifications techniques coté Expérience Utilisateur

1. Le système doit être en WEB2.0 ;
2. Le système doit changer l’url de la page courante quand une requête Ajax est envoyée pour le chargement de la nouvelle page (WEB2.0) ;
3. Toutes les pages web doivent être responsives;
4. Le système doit détecter si le JavaScript est bloqué dans le navigateur du client (UI);
5. Le système doit être compatible avec presque tous les navigateurs ;
6. La navigation coté web doit être en single page application.

## Spécifications techniques coté SERVEUR

1. Le langage utilisé coté serveur est PHP avec le pattern MVC ;
2. Le système doit utiliser MySQL comme SGBD ;
3. Le système doit utiliser le protocole HTTPS ;
4. Le système doit être développé en MVC ;
5. Le système doit être développé en POO sous PHP;
6. Le système doit envoyer toutes les réponses d’une requête XHR (Ajax) sous le format JSON (application/json) ;
7. Le système doit utiliser l’extension PDO (La classe PDO) de PHP pour se connecter à la base de données ;
8. Le système doit utiliser apache comme serveur http ;
9. Le serveur doit utiliser linux comme système d’exploitation;
10. Le système doit permettre l’exportation des documents sous format, PDF ou PNG (pour les administrateurs ministériels, dossier malade pour le médecin, administrateur de transport...etc);
11. Le système doit détecter le userAgent d’une requête;
12. Le système doit détecter une requête XHR ;
13. Le système doit utiliser les sockets pour les discutions instantanées.

## Spécifications techniques coté SECURITE

1. Le système ne doit pas afficher le langage utilisé ou sa version dans l’entête de la réponse https ;
2. Le système ne doit pas afficher le système d’exploitation utilisé ou sa version dans l’entête de la réponse https ;
3. Le système ne doit pas afficher le serveur utilisé ou sa version dans l’entête de la réponse https ;
4. Le système doit utiliser l’algorithme sha1 (40 octets) comme algorithme de hachage (mot de passe (avec des suffixes) dans la base de données et les cookies) ;
5. Le système ne doit pas utiliser les noms des scripts PHP dans les liens vues par l’utilisateur (.htaccess) ;
6. Le système doit vérifier l’identité d’un utilisateur en combinant les données de ses cookies et les données de sa session;
7. Les cookies ont une durée de vie de 10h réinitialisés lors de l’envoie de chaque requête pour les utilisateurs non administrateurs et de 1h pour les administrateurs ;
8. Les cookies de l’utilisateur doivent être changés lors de chaque connexion;
9. L’identifiant de la session doit être changé à chaque requête ;
10. Toutes les données de la base de donnée seront cryptées en RSA ;
11. Toutes les requêtes SQL seront préparées avec PDO->prepare() pour éviter les injections SQL ;
12. Tous les informations affichable (informations venues d’un utilisateur) pour un utilisateur doivent être filtrées avec htmlspecialchars pour éviter les XSS ;
13. Le serveur doit ignorer et fermer les scripts qui prennent plus que 1s d’exécution pour les comptes non ministériels ;
14. Le système doit accepter que les images sous format (PNG, JPG ou JPEG) dans l’upload des fichiers images ;
15. Le système doit accepter que les images avec une taille maximale de 2Mo dans l’upload des fichiers images ;
16. Le serveur principal doit utiliser l’algorithme round robin comme algorithme d’ordonnancement pour le choix d’un serveur dans la répartition de charge (load balancing) ;
17. Le serveur doit ignorer les requêtes POST avec plus de 5Mo comme taille  (pour les comptes non ministériels) ;
18. Le système doit utiliser la bibliothèque PHP « phpseclib » pour crypter les données ;
19. Le système doit utiliser la carte de GoogleMap comme interface de carte.

Proposition : Le système doit installer un firewall contre les attaques DDOS ;

## Spécifications techniques coté Performance :

1. Le système doit disposer d’une base de données centralisée ;
2. Le système doit supporter un grand nombre de données ;
3. Le système doit gérer l’envoie des rappels et les alertes dans un temps très réduit maximum 2 secondes ;
4. Le système doit utiliser un serveur pour le load balancing et utiliser l’algorithme round robin pour le choix de serveur ;
5. Le système doit être en SPA (single page application) pour charger la vue q’une seule fois ;
6. Tous les fichiers html, css, js, et json doivent être envoyés en version minimale (optimisée).

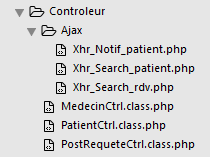
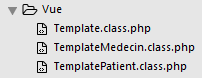
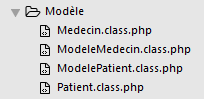
# Diagramme de composant

## Description

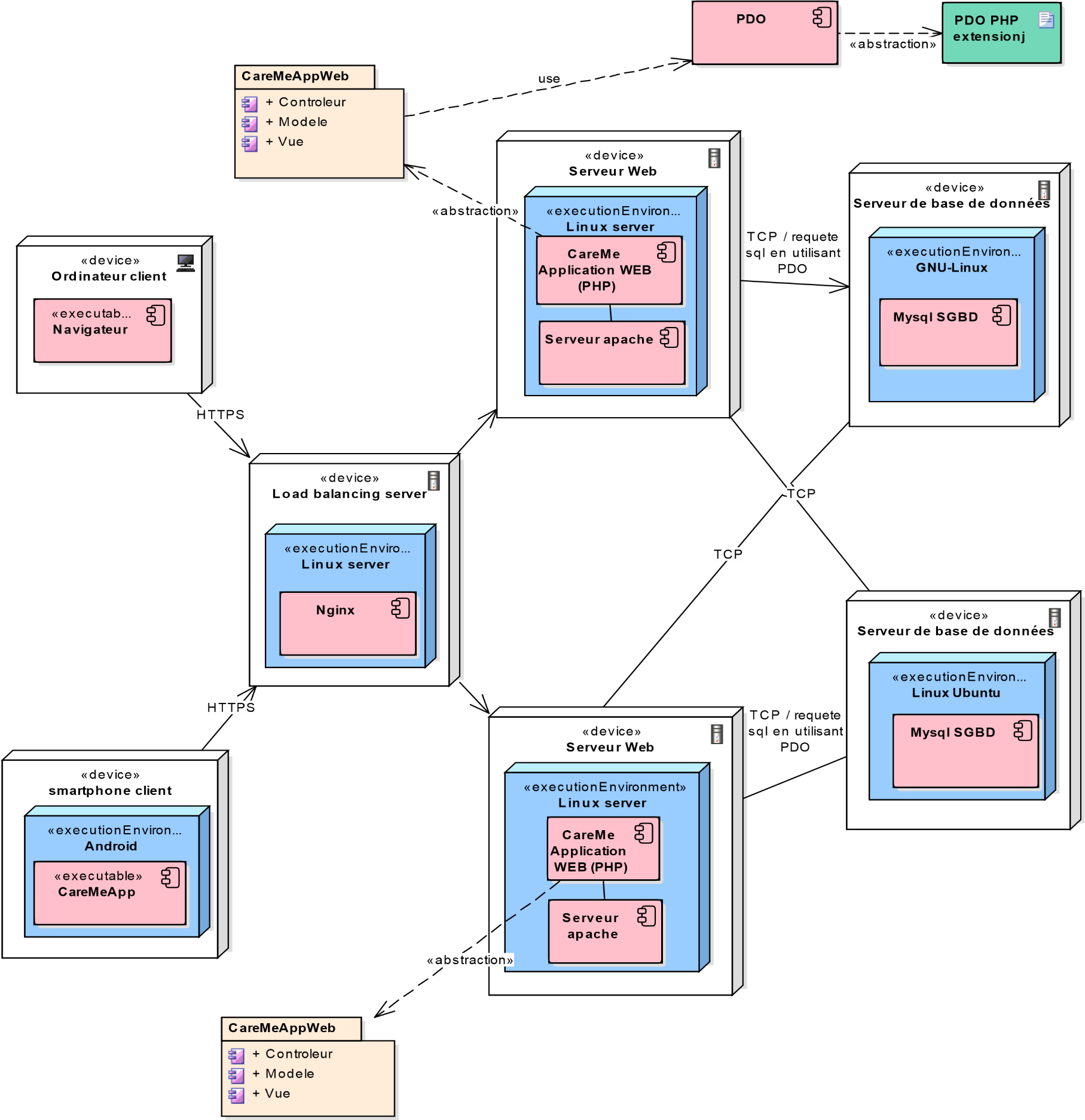
Comme on utilise le pattern MVC sous PHP on a donc découpé la partie programme coté serveur en 3 packages :

* La partie Vue ;
* La partie Modèle ;
* La partie Contrôleur ;

Exemple :



## Diagramme de composant



# C:\Users\magic\Desktop\Component View.pngDiagramme de déploiement