

CENTRE DE FORMATION PROFESSIONNELLE TECHNIQUE

avril - juin 2021

Travail de diplôme



Rapport

Jonathan Borel-Jaquet

# Table des matières

<b>1 Résumé</b>	<b>4</b>
<b>2 Abstract</b>	<b>4</b>
<b>3 Poster</b>	<b>5</b>
<b>4 La société Douceur de Chien</b>	<b>6</b>
4.1 Description . . . . .	6
4.2 Persona . . . . .	7
4.2.1 Identité . . . . .	7
4.2.2 Connaissances en informatique . . . . .	7
4.2.3 Personnalité . . . . .	7
4.2.4 Attentes et connaissances . . . . .	7
4.2.5 Objectifs . . . . .	8
4.2.6 Diplôme . . . . .	8
<b>5 Rappel du cahier des charges</b>	<b>8</b>
5.1 Objectif du projet . . . . .	8
5.2 Description détaillée . . . . .	9
5.3 Liste des fonctionnalités . . . . .	10
5.3.1 Fonctionnalités de l'éducateur canin . . . . .	10
5.3.2 Fonctionnalités du client . . . . .	10
5.4 Scénarios d'utilisation . . . . .	11
5.4.1 Prise en charge d'un nouveau client depuis l'application	11
5.4.2 Prise en charge d'un nouveau client par téléphone . .	11
5.4.3 Première rencontre physique avec le client . . . . .	12
5.4.4 Rendez-vous avec le client . . . . .	12
5.5 Étude des technologies . . . . .	13
5.5.1 Application mobile ou application WEB . . . . .	13
5.5.2 Backend . . . . .	14
5.5.2.1 API REST . . . . .	14
5.5.2.2 Génération de PDF . . . . .	14
5.5.2.3 Envoi de e-mail . . . . .	15
5.5.3 Frontend . . . . .	15
5.5.3.1 Framework . . . . .	15
5.5.3.2 Dessin dans un canevas . . . . .	16
5.5.3.3 Calendrier . . . . .	16
5.6 Diagramme de séquence . . . . .	18

5.7	Mindmap . . . . .	19
5.8	Actions des utilisateurs . . . . .	20
5.9	Fonctionnalités envisagées mais non retenues . . . . .	21
5.9.1	Scan Bluetooth des puces sous-cutanées canine RFID	21
5.10	Planning prévisionnel . . . . .	22
<b>6</b>	<b>Organisation</b>	<b>25</b>
6.1	Analyse concurrentielle . . . . .	25
6.1.1	GESPET . . . . .	25
6.1.2	jegeremonbusiness . . . . .	25
6.1.3	Agenda.ch . . . . .	26
6.2	Environnement de développement . . . . .	26
6.2.1	Laragon . . . . .	26
6.2.2	HeidiSQL . . . . .	27
6.2.3	Visual Studio Code . . . . .	28
6.2.4	Postman . . . . .	29
6.3	Déploiement . . . . .	29
6.4	Gestion de projet . . . . .	29
6.5	Format de documentation . . . . .	32
<b>7</b>	<b>Développement</b>	<b>33</b>
7.1	API REST . . . . .	33
7.1.1	API . . . . .	33
7.1.2	Principes architecturaux REST . . . . .	34
7.1.2.1	HTTP Verbs and Requests . . . . .	34
7.1.2.2	Codes de réponse HTTP . . . . .	35
7.1.2.3	Format de réponse . . . . .	36
7.1.3	Logiciel utilisé . . . . .	36
7.1.3.1	Composer . . . . .	36
7.1.4	Librairies utilisées . . . . .	36
7.1.4.1	PHP dotenv . . . . .	36
7.1.4.2	Dompdf . . . . .	37
7.1.4.3	PHPMailer . . . . .	38
7.1.5	Tests unitaires . . . . .	38
7.1.5.1	Postman . . . . .	39
7.1.6	Outils utilisés . . . . .	39
7.1.6.1	PhpDocumentor . . . . .	39
7.2	PWA . . . . .	40
7.2.1	Vue . . . . .	40
7.2.1.1	Description . . . . .	41
7.2.2	Librairies utilisées . . . . .	41

7.2.2.1	BootstrapVue . . . . .	41
7.2.2.2	Vue Router . . . . .	42
7.2.2.3	Vuex . . . . .	43
7.2.2.4	Axios . . . . .	44
7.2.2.5	Signature Pad . . . . .	44
7.2.2.6	FullCalendar . . . . .	45
7.2.2.7	Moment.js . . . . .	46
7.2.2.8	AlertifyJS . . . . .	46
7.2.2.9	reCAPTCHA . . . . .	47
7.2.3	Tests fonctionnels . . . . .	48
7.2.3.1	Katalon Recorder . . . . .	48
7.2.4	Maquettes effectives . . . . .	49
7.2.4.1	Page d'accueil . . . . .	49
7.2.4.2	Page À propos . . . . .	50
7.2.4.3	Page Agenda . . . . .	50
7.2.4.4	Page Connexion . . . . .	51
7.2.4.5	Page Inscription . . . . .	51
7.2.4.6	Page Administration pour les éducateurs canins . . . . .	52
7.2.4.7	Page des informations du client pour les éducateurs canins . . . . .	52
7.2.4.8	Page des informations du client pour les clients . . . . .	53
7.2.4.9	Page des rendez-vous du client pour les éducateurs canins . . . . .	53
7.2.4.10	Page des rendez-vous du client pour les clients . . . . .	54
7.2.4.11	Page Mon planning pour les éducateurs canins . . . . .	54
7.2.4.12	Page Mes rendez-vous pour les éducateurs canins . . . . .	55
<b>8 Conclusion</b>		<b>56</b>
8.1	Planning prévisionnel et planning effectif . . . . .	56
8.2	Problèmes rencontrés . . . . .	57
8.2.1	Structure de l'API REST . . . . .	57
8.2.2	Service worker de la PWA . . . . .	58
8.2.3	CORS . . . . .	58
8.3	Améliorations envisageables . . . . .	59
8.4	Bilan personnel . . . . .	60
8.5	Remerciements . . . . .	60

# 1 Résumé

Le travail de diplôme réalisé pour la société Douceur de Chien est une PWA (Progressiv Web App), c'est-à-dire, une application disponible de la même manière qu'une application mobile. La société Douceur de Chien est une société d'éducation canine proposant des prestations d'éducation pour les chiens de particuliers. L'idée de ce projet est venue lors d'une rencontre avec l'éducateur canin de la société. En effet, j'ai appris que la prise en charge et la gestion des informations des clients de la société étaient fastidieuses.

Le but du projet est donc de régler cette problématique en centralisant et en facilitant l'accès des données personnelles, des documents signés numériquement et des rendez-vous des clients d'un ou plusieurs éducateurs canins depuis n'importe quel type de périphérique.

# 2 Abstract

The diploma work realized for the company Douceur de Chien is a PWA (Progressiv Web App), a PWA is an application available in the same way as a mobile application. Douceur de Chien is a dog training company offering training services for clients with their dogs. The idea for this project came to me during a meeting with the company dog educator, indeed, I learned that the handling and management of the company's customer information were fastidious.

The goal of the project is to solve this problem by centralizing and facilitating access to the personal data, digitally signed documents and appointments of the clients of one or more dog trainers from any type of device.

### 3 Poster



## 4 La société Douceur de Chien

### 4.1 Description

Douceur de chien est initialement une société d'éducation canine avec à son seul actif, l'éducateur canin Boris Gourdoux. Celui-ci étant diplômé et spécialisé en rééducation et comportement canine, a plus de 300 couples maîtres/chiens à son actif.

L'objectif de Douceur de Chien est de permettre aux maîtres de créer un lien renforcé et durable avec leurs chiens en utilisant uniquement de la patience, de la cohérence et beaucoup de passion. La société porte son nom Douceur de Chien car celle-ci ne propose à ses clients aucun conseil sur l'utilisation de colliers étrangleurs ou autre manière d'en venir à la force pour éduquer les chiens de ces derniers. En effet, la société se base sur une méthode qui respecte les besoins et l'émotionnel du chien en mettant l'accent sur une coopération totale entre ce dernier et son maître.

La société a comme futur objectif la création d'un centre canin permettant à ses clients d'avoir accès à plusieurs services pour leurs chiens : vétérinaire, éducation canine, parc à chiens, boutique canine, etc.

Actuellement, la société compte toujours un seul éducateur canin. Mais dans le futur, celle-ci compte bien en recruter davantage. La problématique actuelle est que la société possède une organisation très fastidieuse, allant jusqu'à stocker les informations et contrats signés et scannés de tous ses clients dans des dossiers sur un Google Drive. Il arrive même que l'éducateur canin oublie ses rendez-vous car ceux-ci sont stockés dans son calendrier privé.

## 4.2 Persona

### 4.2.1 Identité

- Nom : Boris Gourdoux
- Age : 31 ans
- Métier : Educateur et comportementaliste canin

### 4.2.2 Connaissances en informatique

- Utilisation des outils de base Microsoft Office
- Utilisation d'un site internet personnalisé WIX «Douceur de Chien»

### 4.2.3 Personnalité

Monsieur Gourdoux est éducateur canin indépendant. Il rencontre les chiens et leurs maîtres afin de pouvoir les aider à régler certaines difficultés et également à créer une relation de confiance.

Il utilise par ailleurs régulièrement son ordinateur ou son smartphone pour son activité. C'est une personne empathique, autant à l'écoute du chien que du maître. Toutefois, il est très rigoureux dans la partie administrative de son travail. Il aime que les choses soient bien faites. Les supports qu'il utilise doivent être pratiques et intuitifs afin qu'il ne perde pas trop de temps. De plus, il souhaiterait trouver une application pratique et complète qui comporterait tout ce dont il a besoin pour l'organisation de son activité.

### 4.2.4 Attentes et connaissances

Monsieur Gourdoux souhaiterait passer moins de temps à s'occuper de l'administratif et ainsi consacrer plus de temps à ses cours donnés en présentiel afin de pouvoir aider au mieux les chiens et leurs maîtres.

Il s'attend à ce que l'application comporte tout ce dont il a besoin et ceci directement depuis une application, afin qu'il puisse autant travailler depuis la maison qu'à l'extérieur. Monsieur Gourdoux sait déjà très bien gérer son

site internet. Toutefois, il manque des éléments importants pour qu'il puisse mener à bien et sans difficulté son activité.

#### 4.2.5 Objectifs

- Optimiser son temps au niveau administratif et sa relation clients
- Accès facile et rapide à la totalité des informations de ses clients
- Stockage, génération et organisation des différents documents numériques
- Gain de temps lors de la prise de rendez-vous

#### 4.2.6 Diplôme

- Diplôme d'éducateur canin spécialisé en comportement

## 5 Rappel du cahier des charges

### 5.1 Objectif du projet

La PWA (Progressive web app) permettra de faciliter les différentes tâches d'organisation, de prise en charge et de relation d'un client et d'un éducateur canin. L'éducateur canin aura la possibilité de se connecter à l'application afin de pouvoir gérer/visualiser/éditer les différentes informations de ses clients. Il pourra consulter son planning de rendez-vous afin de pouvoir trouver une date adéquate pour un rendez-vous avec un client. L'éducateur canin aura la possibilité de faire signer numériquement les différents contrats lors d'un rendez-vous et d'y stocker avec d'autres documents dans un dossier partagé que le client aura accès depuis son compte.

Le client quant à lui, aura la possibilité de se connecter à l'application afin d'accéder à ses informations personnelles, ses contrats signés, ses documents et ses rendez-vous planifiés avec un éducateur canin. Il pourra également, s'il le souhaite, planifier son propre rendez-vous avec l'éducateur canin depuis le calendrier disponible sur la page d'accueil de l'application.

## 5.2 Description détaillée

La PWA permet à un éducateur canin ou à un client de se connecter à l'application pour avoir accès à différentes fonctionnalités. L'éducateur canin a accès à un calendrier avec tous ses rendez-vous planifiés ainsi qu'à une liste contenant tous ses actuels clients par laquelle il peut, s'il le souhaite, faire une recherche spécifique par nom afin de visualiser ou éditer leurs différentes informations, telles que :

- Les informations personnelles du client
  - Nom
  - Prénom
  - Téléphone
  - Adresse e-mail
  - Adresse du domicile
- Les informations personnelles du/des chiens
  - Date de naissance
  - Race
  - Sexe
  - Nom
  - Photo
  - Numéro de puce sous-cutanée
- Les documents partagés entre le client et l'éducateur canin
  - Conditions d'inscription signées
  - Fiches récapitulatives du cours
  - Autres

L'éducateur canin doit éditer son horaire afin de permettre aux futurs clients de planifier leurs propres rendez-vous.

L'éducateur canin peut procéder à une création préliminaire d'une fiche client. Le client concerné sera averti par e-mail qu'une fiche a été créée avec ses informations. Celui-ci contiendra également un mot de passe généré aléatoirement afin de permettre au client d'accéder à son compte et de, s'il le souhaite, modifier son mot de passe. Lors du premier rendez-vous, l'éducateur canin doit compléter la fiche client avec les données qui nécessitent une rencontre physique.

L'éducateur canin peut à tout moment lors d'un rendez-vous, accéder à la fiche du client afin de pouvoir y rentrer différentes informations :

- Note personnelle de l'éducateur canin du cours sous format texte
- Note personnelle de l'éducateur canin du cours sous format graphique
- Note récapitulative du cours
- Documents à partager

Si le rendez-vous entre l'éducateur canin et le client est le premier, alors l'éducateur doit faire signer depuis l'application les conditions d'inscription au client.

Le client quant à lui, peut une fois connecté, planifier son propre rendez-vous avec un éducateur canin de la société depuis le calendrier disponible sur la page d'accueil de l'application. Le client connecté peut également accéder à ses informations personnelles ainsi que ses documents.

### 5.3 Liste des fonctionnalités

#### 5.3.1 Fonctionnalités de l'éducateur canin

- Connexion à l'application
- Accès au calendrier de ses rendez-vous
- Édition de l'horaire personnel
- Affichage de tous les clients avec photo du chien et nom de son maître
- Recherche spécifique d'un client par nom
- Accès aux informations personnelles d'un client depuis la recherche spécifique ou du calendrier de rendez-vous
- Création préliminaire d'une fiche client
- Accès et création du contenu séance d'un client depuis la recherche spécifique ou le calendrier de rendez-vous
- Prise de rendez-vous

#### 5.3.2 Fonctionnalités du client

- Prise de rendez-vous autonome
- Inscription à l'application
- Connexion à l'application
- Accès au calendrier de ses rendez-vous

- Accès à ses informations personnelles
- Accès à ses documents personnels
- Accès à ses différents contenus séances

## 5.4 Scénarios d'utilisation

### 5.4.1 Prise en charge d'un nouveau client depuis l'application

1. Le client s'inscrit à l'application en rentrant différentes informations comme :
  - Son nom
  - Son prénom
  - Son numéro de téléphone
  - Son adresse e-mail
  - Son adresse de domicile
  - Son mot de passe
2. Le client accède à l'application et choisit dans le calendrier de la page d'accueil son propre rendez-vous.
3. Le client reçoit un e-mail avec les informations de son rendez-vous.
4. L'éducateur canin reçoit un e-mail avec toutes les données du rendez-vous qu'il peut également consulter depuis son calendrier personnel.

### 5.4.2 Prise en charge d'un nouveau client par téléphone

1. Le client appelle l'éducateur canin car il a besoin de ses services.
2. L'éducateur canin se rend sur l'application et se connecte.
3. Il se rend sur l'interface de création d'une nouvelle fiche client.
4. Il insère les informations personnelles du client communiquées par téléphone :
  - Nom du client
  - Prénom du client
  - Numéro de téléphone du client
  - Adresse e-mail du client
  - Adresse de domicile du client
  - La date de naissance du/des chiens

- La race du/des chiens
  - Le sexe du/des chiens
  - Le nom du/des chiens
5. Une fois la fiche client préliminaire créée, un e-mail contenant un mot de passe généré aléatoirement est envoyé au client.
  6. L'éducateur canin aura accès à son calendrier personnel afin de visualiser à quelle date il peut prendre rendez-vous avec le client.
  7. Le client ainsi que l'éducateur se mettront d'accord sur la date du rendez-vous.
  8. L'éducateur canin sélectionnera cette date dans le calendrier.
  9. Une fois le rendez-vous planifié, un e-mail contenant les informations du rendez-vous est envoyé au client.

#### **5.4.3 Première rencontre physique avec le client**

1. Le client se rend au centre d'éducation canine à la date spécifiée lors de l'appel téléphonique ou du rendez-vous pris depuis l'application.
2. L'éducateur canin se rend sur l'application et se connecte.
3. Il recherche le client grâce à son nom et accède à sa fiche client précédemment créée lors de l'appel téléphonique ou du rendez-vous pris depuis l'application.
4. Il montre les données personnelles du client et lui demande une vérification de celles-ci.
5. Si elles sont fausses, il procède aux modifications de celles-ci.
6. Si elles sont correctes, l'éducateur canin peut prendre une photo du chien et rentrer manuellement les 15 chiffres du code de la puce sous-cutanée du chien (si existante).

#### **5.4.4 Rendez-vous avec le client**

1. L'éducateur canin peut à tout moment lors d'un rendez-vous, accéder au contenu de séance afin de pouvoir y ajouter différentes données :
  - Note du cours sous format texte (accessible uniquement par l'éducateur)

- Note du cours sous format graphique (accessible uniquement par l'éducateur)
  - Note récapitulative du cours (accessible par l'éducateur ainsi que le client)
  - Document PDF (accessible par l'éducateur ainsi que le client)
2. Si le rendez-vous est le premier, alors le client doit depuis l'application de l'éducateur :
    - (a) Visualiser les conditions d'inscription.
    - (b) Choisir le forfait qu'il désire.
    - (c) Valider s'il est d'accord en cochant une case "Lu et approuvé".
    - (d) Ajouter sa signature depuis l'application.
    - (e) Les conditions d'inscription sous format PDF ainsi qu'une génération automatique d'une facture sous format PDF seront ajoutées au dossier partagé du client et envoyées par e-mail au client.

## 5.5 Étude des technologies

### 5.5.1 Application mobile ou application WEB

Après avoir discuté avec le client de mon application, nous avons convenu qu'il lui faudrait une application disponible et utilisable sur mobile. Je me suis premièrement tourné vers le développement d'une application mobile avec Cordova. Après discussion avec des professeurs du CFPT, ces derniers ont mis en évidence une certaine problématique : le développement et la mise en place d'une application mobile pour les clients d'un éducateur canin sont trop restrictifs et pas réellement pratiques. Afin de répondre aux attentes de mon application, je me suis orienté sur le développement d'une PWA, c'est-à-dire une application WEB disponible autant sur grand écran que petit écran et affichée de la même manière qu'une application mobile. En effet, le client souhaite accéder à l'application depuis une tablette ou depuis son ordinateur. Une PWA est un site WEB responsive que l'on peut ajouter à son bureau ou à son écran d'accueil de téléphone/tablette afin d'y accéder plus facilement sans passer par un navigateur.

### 5.5.2 Backend

#### 5.5.2.1 API REST

Afin de gérer les différentes données de mon application, j'ai décidé de développer une API respectant l'ensemble des principes architecturaux REST me permettant de rendre mes données plus facilement utilisables lors de futurs projets en lien avec mon travail de diplôme. Pour la réalisation de cette API REST et après quelques recherches, j'ai été tenté par plusieurs choix de frameworks, comme :

- Ruby on Rails
- Phoenix
- Laravel
- Django
- Yii

Ruby on Rails utilisant comme langage Ruby, Phoenix utilisant Elixir et Django utilisant Python, je me suis tout naturellement tourné vers l'utilisation d'un framework PHP. En effet, PHP est un langage beaucoup plus familier pour moi, car je l'ai utilisé tout au long de ma formation. Restant les frameworks PHP Yii et Laravel à départager, j'ai en premier lieu choisi Laravel pour différentes raisons. La première est que je l'ai déjà utilisé lors de ma formation. La deuxième est que Laravel fournit une documentation complète et lisible ainsi qu'une communauté conséquente. La dernière est que Laravel propose un nombre incalculable de fonctionnalités très utiles directement intégrées à celui-ci. Toutefois, malgré toutes les qualités qu'un quelconque Framework peut m'offrir lors du développement de mon API, j'ai décidé de développer mon API en pur PHP afin de rendre mon API maintenable plus facilement. En effet, le développement en PHP pur de mon API permettrait à un autre développeur de reprendre le développement de celle-ci sans contrainte de connaissance dans un framework particulier.

#### 5.5.2.2 Génération de PDF

Afin de pouvoir créer manuellement les différents documents PDF de mon application, j'ai recherché différentes façons de procéder. Après m'être rendu compte que la plupart des outils existants étaient payants (PDFTron, ra-

daeePDF), je me suis dirigé vers une solution gratuite et open-source de génération de PDF en PHP proposé par mon maître de diplôme : TCPDF. Malheureusement, lors de mon POC, je me suis rendu compte que la version actuelle n'était pas maintenue. J'ai donc continué mes recherches jusqu'à trouver une librairie très pratique me permettant de convertir du HTML en PDF : DOMPDF<sup>1</sup>.

### 5.5.2.3 Envoi de e-mail

Afin d'envoyer les différents e-mails aux clients lors de la création du compte ou la création des documents PDF, je compte utiliser PHPMailer<sup>2</sup>.

## 5.5.3 Frontend

### 5.5.3.1 Framework

Afin de développer le frontend de mon application, plus précisément de ma PWA (Progressive web app), j'ai été confronté à différents choix de frameworks, comme :

- Angular
- React
- Vue
- Bootstrap

Angular offre beaucoup de fonctionnalités très pratiques. Il permet de réduire la quantité de code en fournissant des fonctionnalités importantes par défaut, il propose une synchronisation en temps réel entre le modèle et la vue et de ce fait, le rend très performant. Malgré sa documentation très détaillée, Angular est très complexe et demande l'apprentissage de nouveaux concepts comme l'utilisation de TypeScript.

React est connu pour être la concurrence directe d'Angular. En effet, ce framework est connu pour offrir une meilleure courbe d'apprentissage. React se démarque par sa simplicité et son développement rapide permettant un gain

---

1. <https://github.com/dompdf/dompdf>

2. <https://github.com/PHPMailer/PHPMailer>

de temps, il permet également une intégration rapide sur mobile via React Native. Comme inconvénient, on pourrait reprocher à React un manque de documentation officielle.

Ayant partiellement utilisé Vue lors de mon apprentissage, j'en ai déduit qu'il était bon de l'inclure dans ma recherche. En effet, Vue met à disposition de ses utilisateurs une documentation complète et très détaillée ainsi qu'une syntaxe simple et pure JavaScript. Le point faible de Vue pour moi est sa complexité due à sa syntaxe particulière.

Le dernier de ma liste est Bootstrap. Ce framework plus que complet propose un nombre conséquent de composants facilement intégrables ainsi qu'une énorme communauté contribuant au développement d'autres composants. Sa simplicité, sa documentation très complète et son système de grille en fait également l'un des frameworks les plus utilisés.

Lors de ce travail de diplôme, je compte mélanger défi et connaissance. En effet, je compte utiliser Bootstrap pour sa simplicité ainsi que Vue qui implique pour moi l'étude d'une nouvelle technologie et donc d'un défi. Un des éléments qui m'a également motivé à faire ce choix est la découverte de l'existence de BootstrapVue<sup>3</sup>, qui, comme son nom l'indique, permet l'intégration de Bootstrap avec Vue en toute simplicité et de manière compatible.

#### **5.5.3.2 Dessin dans un canevas**

Afin de pouvoir signer numériquement les contrats ainsi que de prendre les notes graphiques de mon application, j'ai choisi un module responsive sans dépendances pour répondre aux attentes de mon application qui devra être fonctionnelle sur mobile : Responsive-Scketchpad<sup>4</sup>.

#### **5.5.3.3 Calendrier**

Afin d'afficher les rendez-vous de l'éducateur canin ainsi que des clients, je compte utiliser un calendrier. Pour respecter la disponibilité de mon application sur mobile, j'ai besoin d'implémenter un calendrier responsive. Afin de ne pas perdre trop de temps lors du développement, je compte

---

3. <https://bootstrap-vue.org/>

4. <https://github.com/tsand/responsive-sketchpad>

utiliser FullCalendar<sup>5</sup> car il est très complet. Malheureusement celui-ci ne permet pas d'affichage responsive.

---

5. <https://fullcalendar.io/>

## 5.6 Diagramme de séquence



FIGURE 1 – Diagramme de séquence prévisionnel

## 5.7 Mindmap

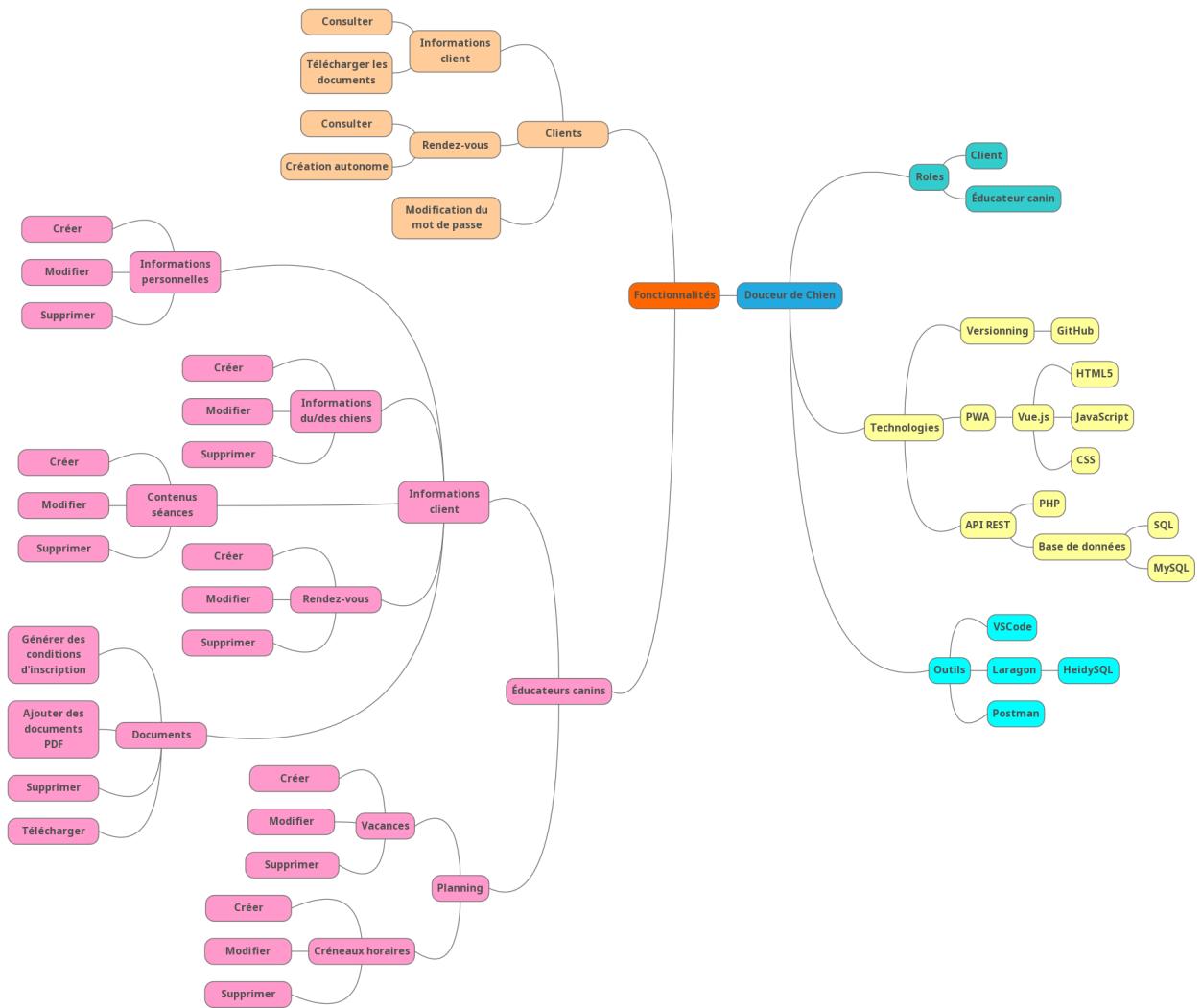


FIGURE 2 – Mindmap du projet Douceur de Chien

## 5.8 Actions des utilisateurs



FIGURE 3 – Diagramme des actions des utilisateurs

## 5.9 Fonctionnalités envisagées mais non retenues

### 5.9.1 Scan Bluetooth des puces sous-cutanées canine RFID

Lors de la réalisation de ce cahier des charges, j'ai envisagé le fait d'introduire la possibilité de scanner le code de la puce sous-cutanée canine directement en Bluetooth grâce à un équipement externe. J'ai décidé de ne pas retenir cette fonctionnalité due à sa grande complexité et à son utilité minime. En effet, d'une part, le temps de développement est trop conséquent par rapport à la plus-value directe que peut me rajouter cette fonctionnalité, d'autre part, ce code est utilisable uniquement par des vétérinaires et autre autorité compétente. Je compte tout de même garder cette donnée afin de la rendre compatible avec d'autres futurs projets pour la société Douceur de Chien.

## 5.10 Planning prévisionnel

Tâches à réaliser	Temps nécessaire en périodes	19.4	20.4	21.4	22.4	23.4	26.4	27.4	28.4	29.4	
		1er jour	2e jour	3e jour	4e jour	5e jour	6e jour	7e jour	8e jour	9e jour	
<b>API Rest</b>											
<b>API routes</b>											
User	4	4									
Dog	4	2	2								
Document	12		4	6	2						
Appoiment	8				4	4					
Planning	16					2	6	6	2		
Absence	4								4		
<b>Vues</b>											
<b>Client</b>											
Onglet rendez-vous autonome	4								4		
Onglet informations personnelles	4								2		
Onglet calendrier	8										
Onglet contenus séances	8										
<b>Éducateur canin</b>											
Onglet d'affichage de tous les clients	4										
Onglet calendrier	4										
Onglet de création de fiche client préliminaire	4										
Onglet d'affichage de fiche client	4										
Onglet de validation de fiche client	4										
Onglet des contenus séances d'un client	4										
Onglet d'ajout de documents	4										
<b>Fonctionnalités</b>											
Progressive web app	8										
Inscription	6										
Connexion	6										
Affichage calendrier avec rendez-vous	8										
Gestion des disponibilités de l'éducateur canin	16										
Lecture PDF	8										
Téléchargement PDF	8										
Génération de document PDF	8										
Prise de photo du chien	16										
Dessin de notes personnelles	10										
<b>Documentation</b>											
Rapport	68	1	1	1	1	1	1	1	1	1	
Workbook	38	1	1	1	1	1	1	1	1	1	
	300	8	8	8	8	8	8	8	8	8	

FIGURE 4 – Planning prévisionnel 1/3

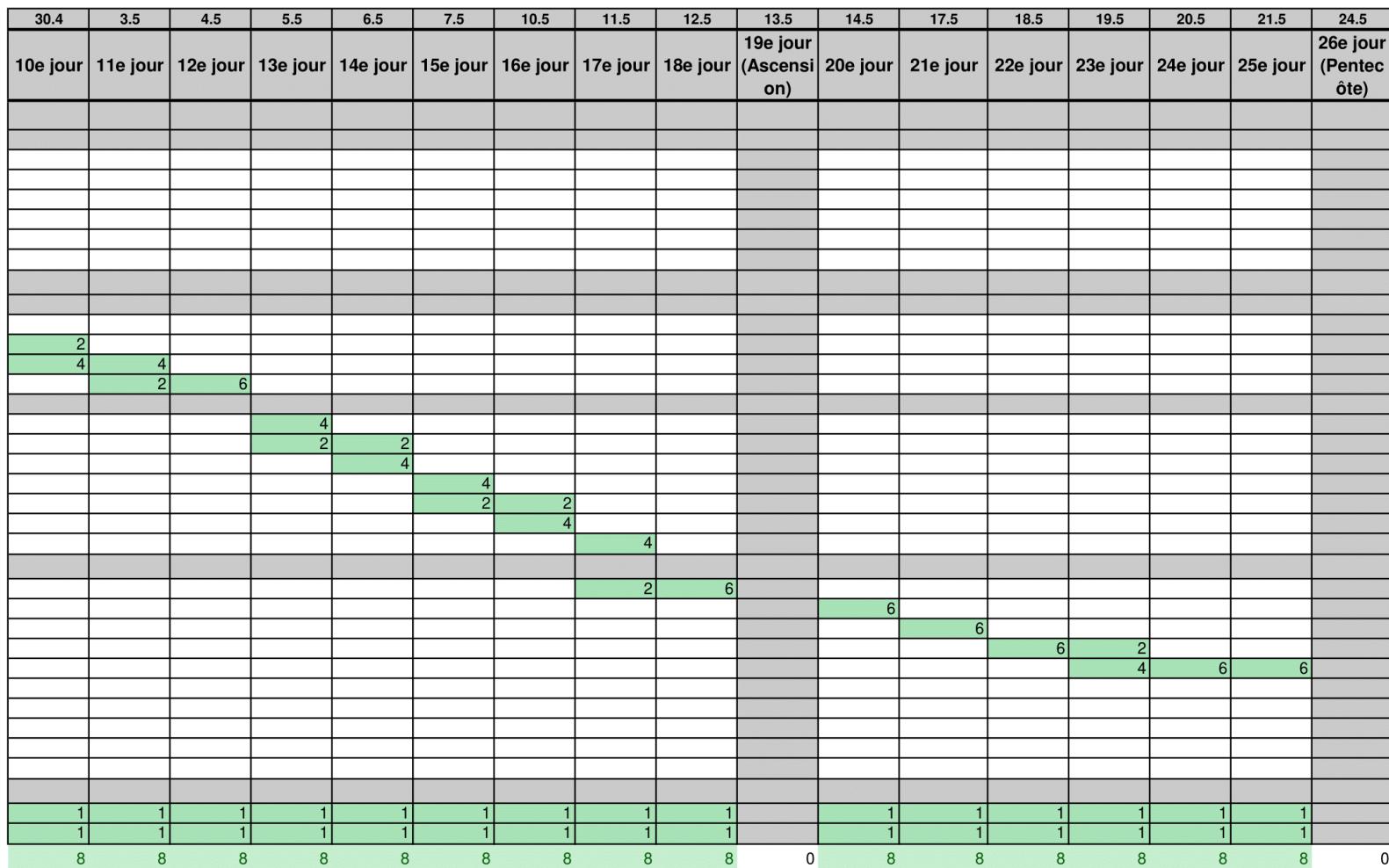


FIGURE 5 – Planning prévisionnel 2/3

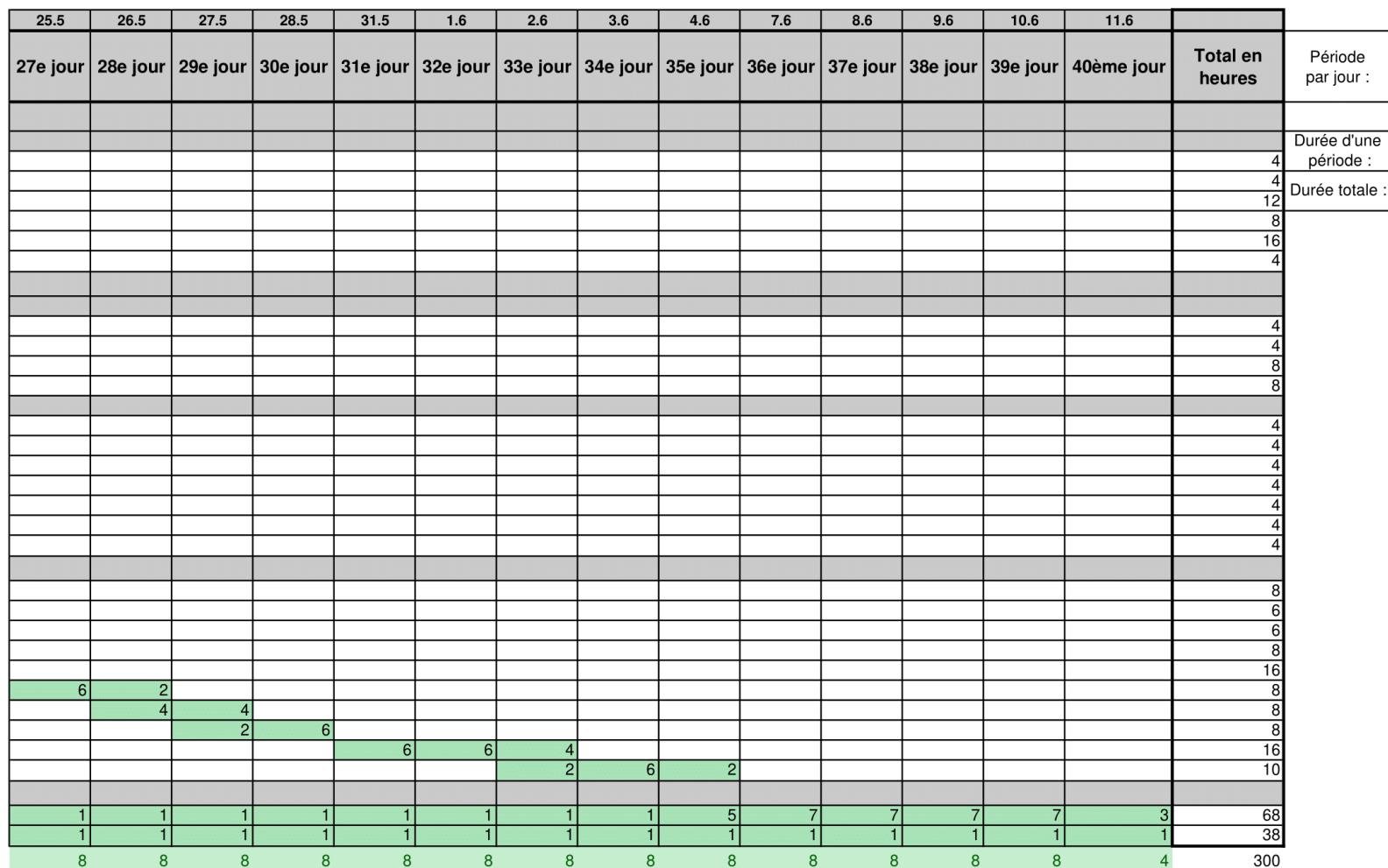


FIGURE 6 – Planning prévisionnel 3/3

## 6 Organisation

### 6.1 Analyse concurrentielle

#### 6.1.1 GESPET

GESPET est une application proposant une tonne de fonctionnalités pour :

- des refuges
- des dresseurs d'animaux
- des transports d'animaux
- des animaleries
- etc.

Après avoir essayé la version de démonstration, j'ai trouvé que l'application GESPET était très complète mais qu'elle partait dans tous les sens. En effet, celle-ci propose une multitude de fonctionnalités pour des éducateurs comme : la gestion de leurs clients, la gestion de leurs rendez-vous, la gestion de leurs factures et revenus. Malgré toutes ces fonctionnalités, l'application ne permet aucune interaction avec les clients des éducateurs, elle permet uniquement de faciliter l'organisation d'un ou plusieurs acteurs du métier en proposant pleins d'outils spécifiques. De plus, l'application étant d'origine espagnole, la traduction française est mauvaise et les différentes possibilités d'aide comme le forum ou les vidéos YouTube sont en espagnol, tout cela ne facilitant pas son utilisation.

#### 6.1.2 jegeremonbusiness

L'application jegeremonbusiness est une application développée par la société "L'artisan web". L'application permet de gérer différentes choses comme :

- les statistiques
- les plannings
- les clients et leurs chiens
- les factures
- etc.

Après avoir essayé la version de démonstration, j'ai trouvé que l'application proposée par "L'artisan web" était simple et efficace. L'interface utilisateur est facile et l'application est responsive, c'est-à-dire utilisable sur petit écran comme grand écran. Toutefois, l'application est difficilement utilisable par plusieurs éducateurs canins. De plus, et similaire à l'application GESPET, cette application permet uniquement d'aider un éducateur canin dans sa gestion de son entreprise et ne propose aucune fonctionnalité aux clients de celui-ci.

### 6.1.3 Agenda.ch

Agenda.ch est une application permettant aux clients d'une société proposant des services de rendez-vous de réserver leurs propres rendez-vous en ligne. Elle permet aux sociétés de configurer tous les paramètres de la réservation en ligne : la présentation, les disponibilités, les durées, etc. Le point fort de agenda.ch est que sa réservation en ligne est très simple et ne nécessite pas la création de compte ou d'appel téléphonique pour la prise de rendez-vous.

Je compte m'inspirer de l'application agenda.ch pour la prise de rendez-vous autonome de mon application. En effet, mon application va permettre aux clients de la société Douceur de Chien de planifier leurs propres rendez-vous avec un des éducateurs canins de la société. Intégrer cette fonctionnalité est possible car les rendez-vous de la société se dérouleront au centre canin de la société et non chez le client.

## 6.2 Environnement de développement

### 6.2.1 Laragon

Laragon est un environnement de développement à lui tout seul. Portable, rapide et puissant pour différents langages comme PHP par exemple, Laragon offre à ses utilisateurs la possibilité de démarrer ses propres services facilement et rapidement afin de lancer un serveur HTTP ou un serveur MySQL et d'utiliser différents outils pré-installés comme HeidiSQL.

Dans le cadre de mon travail de diplôme, j'ai utilisé Laragon pour tous les

éléments que je viens de vous citer. Un serveur HTTP pour tester en local mon API REST développée en PHP et un serveur MySQL pour permettre à celle-ci de communiquer avec une base de données en me servant de l'outil d'administration de base de données HeidiSQL quand il le fallait pour éditer ma base de données ou tester les différentes requêtes SQL de l'API REST.

Il existe plusieurs manières différentes de mettre en place un environnement de développement. Pour ma part, j'ai choisi d'utiliser Laragon car il est simple à mettre en place et que son interface graphique permet de gérer facilement les différents fichiers de configuration d'un environnement de développement.

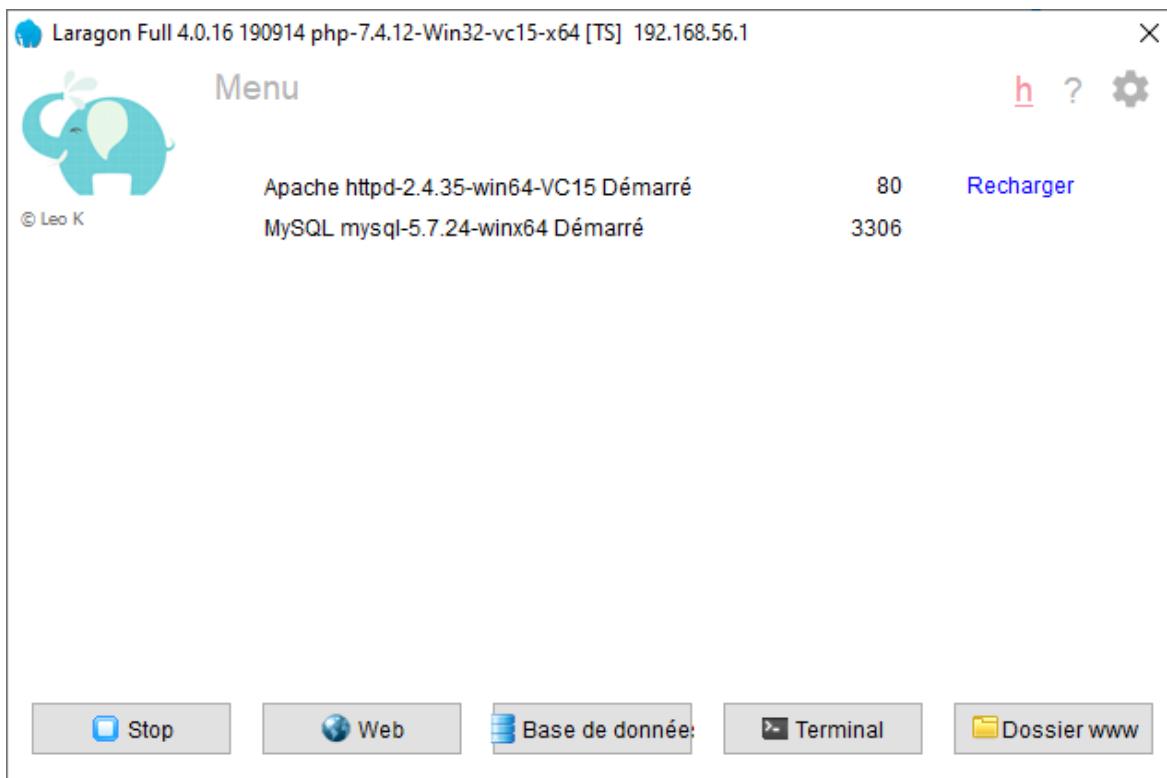


FIGURE 7 – Interface graphique de l'application Laragon

### 6.2.2 HeidiSQL

Comme expliqué auparavant, Laragon propose à ses utilisateurs l'outil HeidiSQL. HeidiSQL est un outil d'administration de base de données possédant un éditeur SQL et un constructeur de requête. Celui-ci a été développé et optimisé pour être justement utilisé avec le SGBDR (Système de Gestion de Base de Données Relationnelles) MySQL. Pour ma part, j'ai utilisé cet

outil pour éditer les tables de ma base de données ainsi que tester mes différentes requêtes SQL. Les requêtes SQL sont une sorte de questions posées à une base de données. Celles-ci peuvent comporter un ou plusieurs critères pour préciser leurs demandes. SQL est le langage de requête que j'ai utilisé. Néanmoins, il en existe bien des autres spécifiques à d'autres structures de base de données.

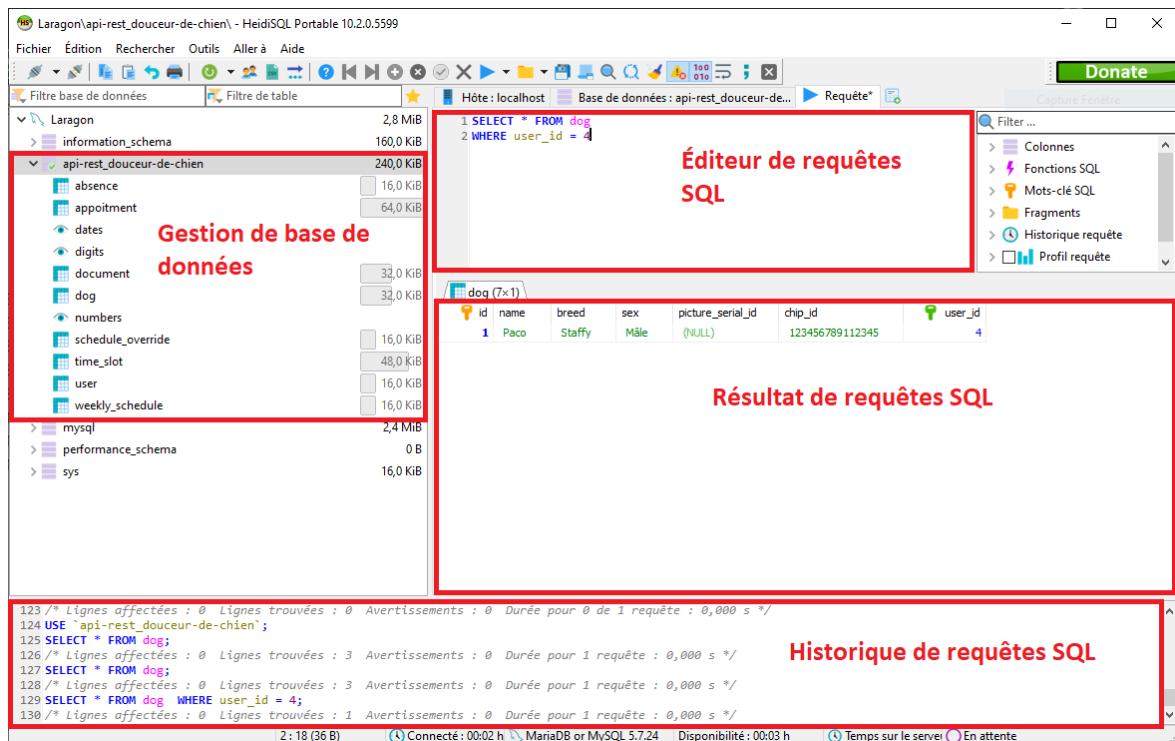


FIGURE 8 – Interface de l'outil d'administration de base de données HeidiSQL

### 6.2.3 Visual Studio Code

Visual Studio Code est un éditeur de code extensible. Comme sa définition l'indique, Visual Studio Code propose à ses utilisateurs la possibilité d'intégrer à celui-ci une multitude d'extensions plus pratiques les unes que les autres. J'utilise Visual Studio Code depuis maintenant quatre ans et je ne comptais pas faire exception lors de ce travail de diplôme. J'ai utilisé Visual Studio Code pour réaliser le frontend ainsi que le backend de mon travail de diplôme. En effet, je me suis servi de plusieurs extensions permettant l'auto-complétion<sup>6</sup> de code PHP, JavaScript, HTML, CSS tout au long de

6. Fonctionnalité informatique permettant à l'utilisateur de limiter la quantité d'informations qu'il saisit avec son clavier

mon travail.

#### 6.2.4 Postman

Postman est une plateforme de collaboration pour le développement d'API. Postman propose plusieurs fonctionnalités à ses utilisateurs afin de les aider dans le développement de leurs API. Dans mon cas, j'ai utilisé Postman pour deux raisons. La première raison est que Postman permet de facilement exécuter des requêtes HTTP. La deuxième est que Postman propose à ses utilisateurs la possibilité de réaliser des tests unitaires et c'est justement ce dont j'avais besoin pour tester les différents cas d'utilisation de mon API REST.

### 6.3 Déploiement

Afin de rendre mes projets accessibles, j'ai fait l'acquisition cette année d'un nom de domaine et d'un hébergement WEB sur Infomaniak. En effet, Infomaniak propose pour les étudiants un hébergement WEB et un hébergement Mail gratuit. Toutefois, le nom de domaine restant à notre charge, je me suis procuré le nom de domaine *boreljaquet.ch*. Une fois l'hébergement WEB et Mail configurés, ainsi que le nom de domaine acheté, j'ai pu mettre en production mes différents projets à des fins de test et de démonstration sur mes différents sous-domaines. Pour mon travail de diplôme j'en ai justement profité pour publier mon API REST et ma PWA afin de tester mon projet lors d'une mise en production.

### 6.4 Gestion de projet

Pour planifier les différentes tâches de mon travail de diplôme, j'ai décidé d'utiliser l'outil de gestion de projet en ligne Trello<sup>7</sup>. Normalement, Trello est un outil très pratique en travail de groupe. En effet, il permet de créer différentes tâches en les catégorisant et en leur assignant un membre de l'équipe.

---

7. <https://trello.com/>

Dans ma formation, j'ai souvent été confronté à utiliser Trello dans des projets de groupe et plus précisément de la méthode Kanban. Dans ces versions les plus simplistes, Kanban est une méthode de gestion de projet se reposant sur trois états de tâches distinctes :

1. To Do
  - Les tâches qu'il faut réaliser
2. In Progress
  - Les tâches qui sont en train d'être réalisées
3. Done
  - Les tâches qui sont terminées

Malgré le fait que je sois seul à réaliser mon travail de diplôme, j'ai tout de même utilisé Trello avec la méthode Kanban afin de découper mon travail en différentes tâches distinctes. Ce découpage m'a permis de visualiser plus clairement le déroulement de mon projet afin de ne pas me perdre dans les tâches à réaliser.



FIGURE 9 – Catégories des tâches Trello

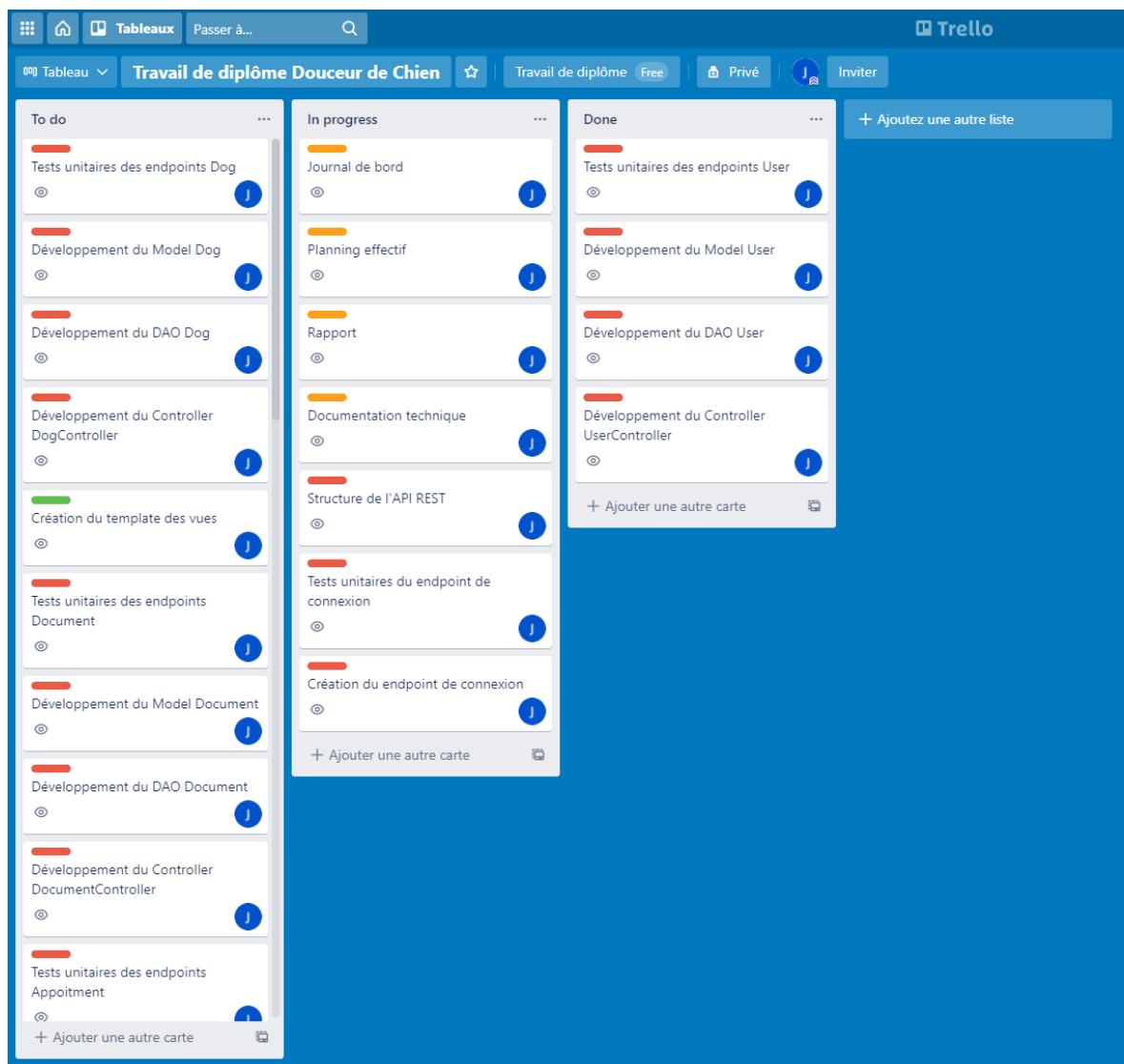


FIGURE 10 – Trello du projet utilisant la méthode Kanban

## 6.5 Format de documentation

Le format de documentation est toujours un point flou et complexe de tout travail de diplôme. Pour régler ce point au plus vite, je suis tout de suite allé voir mon maître de diplôme M. Mathieu. Un échange a eu lieu par rapport à l'élaboration de la documentation. Nous avons convenu que la documentation de mon travail de diplôme devra être découpé en trois parties différentes.

La première est le rapport, document que je symbolisera théorique. On y retrouve les informations non-techniques du travail de diplôme. Celui-ci pourrait être donné à n'importe qui afin qu'il puisse comprendre les motivations, les démarches et les autres éléments nécessaires qui m'ont permis de réaliser mon projet de diplôme.

La deuxième partie est la documentation technique. Contrairement au rapport, cette documentation a comme objectif de permettre à un autre développeur en informatique de pouvoir comprendre les différents outils et éléments techniques que j'ai réalisés et utilisés lors de mon travail de diplôme pour lui faciliter, si nécessaire, la récupération du projet.

La dernière partie est le journal de bord. Celui-ci contient le déroulement de mon travail de diplôme. Il permet de se rendre compte des différentes réflexions et développements que j'ai réalisés tout au long de mon projet jour après jour.

## 7 Développement

### 7.1 API REST

#### 7.1.1 API

Lors de ce travail de diplôme, j'ai réalisé une API REST. Avant de commencer l'explication de qu'est ce qu'une API REST, commençons par expliquer qu'est ce qu'une API en tant que telle. L'acronyme API de son appellation française *Interface de programme d'application* est un ensemble de règles permettant à différents programmes de communiquer entre eux. L'objectif d'une API est de permettre à différentes applications la possibilité de créer, de récupérer, de modifier et d'insérer des données dans un but commun.

Dans mon cas, j'ai décidé de réaliser une API afin de centraliser les différentes données de la société dans le but de les rendre réutilisables plus facilement pour d'autres applications futures de la société Douceur de Chien, comme une application d'entreprise ou une application mobile par exemple.

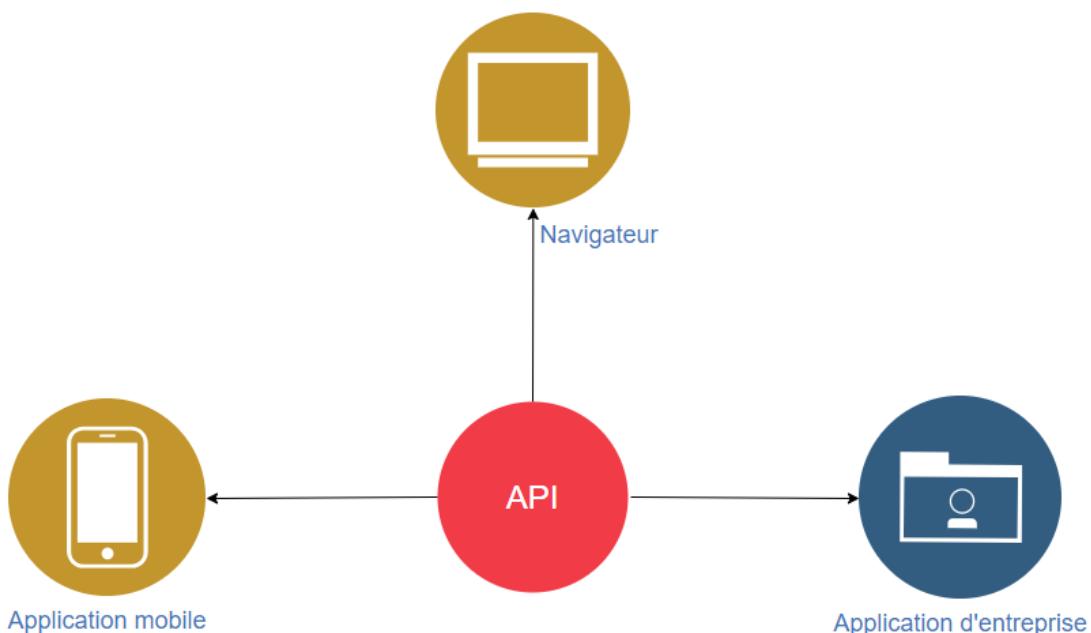


FIGURE 11 – Représentation de l'utilité d'une API

## 7.1.2 Principes architecturaux REST

### 7.1.2.1 HTTP Verbs and Requests

Une API REST est une API respectant les principes architecturaux REST signifiant *REpresentational State Transfer*. Son principe est de faire appel à des requêtes HTTP<sup>8</sup> pour obtenir, modifier/remplacer, créer et supprimer des données. Ces différents types de requêtes HTTP sont identifiés par ce que l'on appelle des *HTTP Verbs*, ceux que j'ai utilisés sont :

- POST
- GET
- PATCH
- DELETE

Pour faire une requête à une API REST, il va falloir utiliser ce qu'on appelle des *endpoints*. Les *endpoints* sont des URL permettant à l'API d'identifier la ressource qu'elle doit créer (POST), récupérer (GET), modifier (PATCH) ou supprimer (DELETE). Par exemple :

- Pour créer un nouvel utilisateur :
  - L'endpoint sera *api/v1/users* avec le *HTTP Verb* POST
- Pour récupérer les informations du chien avec l'identifiant numéro 3 :
  - L'endpoint sera *api/v1/dogs/3* avec le *HTTP Verb* GET
- Pour modifier les informations de l'utilisateur avec l'identifiant numéro 12 :
  - L'endpoint sera *api/v1/users/12* avec le *HTTP Verb* PATCH
- Pour supprimer les informations du chien avec l'identifiant numéro 7 :
  - L'endpoint sera *api/v1/dogs/7* avec le *HTTP Verb* DELETE

---

8. Hypertext Transfer Protocol (HTTP) est un protocole de communication client-serveur.

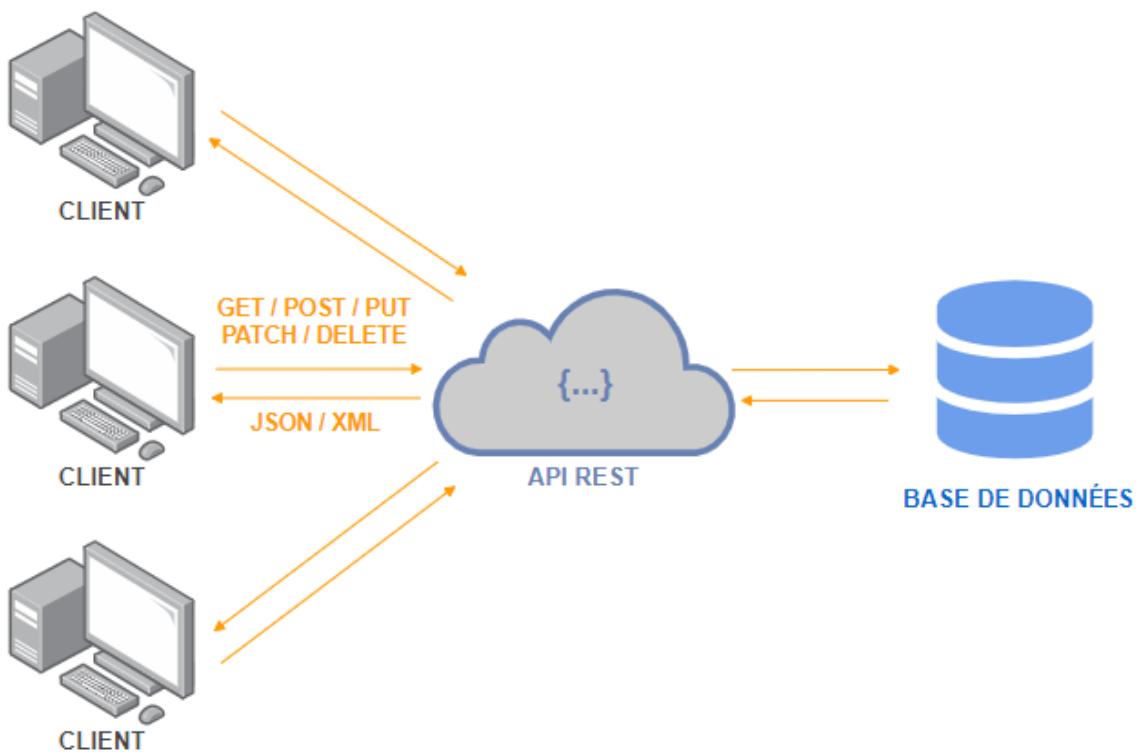


FIGURE 12 – Représentation de communication d'une API REST

### 7.1.2.2 Codes de réponse HTTP

Une API REST doit retourner différents types de code de réponse HTTP. Également appelés codes d'état, les codes HTTP permettent de signaler si une requête HTTP a été exécutée avec succès ou non. Il existe une multitude de codes de réponses regroupées en cinq classes :

- Les réponses informatives allant du code 100 jusqu'au 199
- Les réponses de succès allant du code 200 jusqu'au 299
- Les redirections allant du code 300 jusqu'au 399
- Les erreurs du client allant du code 400 jusqu'au 499
- Les erreurs du serveur allant du code 500 jusqu'au 599

Malgré le fait qu'il existe une multitude de codes de réponses HTTP, l'API REST que j'ai développée lors de ce travail de diplôme en retourne réellement que quelques unes différentes. Parmi celles-ci se trouvent les codes :

- 200 OK
- 201 Created
- 400 Bad Request
- 401 Unauthorized

- 403 Forbidden
- 404 Not Found
- 415 Unsupported Media Type

### 7.1.2.3 Format de réponse

Lorsqu'une API REST doit retourner la représentation d'une ressource, celle-ci peut retourner cette représentation en différents formats. Dans le cadre de mon travail de diplôme, j'ai utilisé le format de réponse JSON pour la simple et bonne raison que j'ai utilisé le langage JavaScript pour le frontend de mon application, que je le trouve davantage compréhensible, simple à lire et que je suis plus à l'aise avec l'utilisation de celui-ci.

## 7.1.3 Logiciel utilisé

### 7.1.3.1 Composer

Composer est un logiciel permettant de gérer les différentes dépendances PHP d'un projet. Il permet à ses utilisateurs de gérer et intégrer des librairies dans leurs projets basés sur le langage PHP en écrivant des lignes de commande depuis une interface de ligne de commande. En effet, Composer est un outil très pratique pour installer, désinstaller ou mettre à jour ses dépendances PHP.

J'ai utilisé Composer pour utiliser et gérer les différentes librairies que j'ai utilisées lors de la réalisation de mon API REST en PHP comme : Dompdf et PHPMailer.

### 7.1.4 Librairies utilisées

#### 7.1.4.1 PHP dotenv

PHP dotenv est une librairie permettant de générer et charger des variables d'environnement. PHP dotenv est un moyen simple de charger des données de configuration personnalisées dans une application PHP. Ces variables d'environnement contiennent les informations sensibles permettant le bon

fonctionnement de mon API REST. L'utilisation de PHP dotenv dans mon API REST permet de configurer rapidement les données de connexion à la base de données, les informations de connexion au serveur SMTP ainsi que d'y renseigner la ou les clefs d'API personnelles utilisées.

#### 7.1.4.2 Dompdf

Dompdf est une librairie permettant la conversion de fichier HTML avec du CSS en PDF. Cette librairie m'a permis de créer les conditions d'inscription de la société depuis un modèle créé en HTML avec du CSS. Son fonctionnement est de lire le rendu final d'une page HTML avec son CSS afin de convertir celui-ci en document PDF. Dans mon cas, j'ai créé une page HTML qui se modifie en fonction des informations du client comme son nom, sa signature, le numéro de forfait qu'il a choisi et la date du jour actuelle. Une fois que ce modèle HTML avec son CSS pour le rendre plus joli et contenant les différentes données des clients ajoutées, j'ai utilisé Dompdf pour convertir ce rendu final en document PDF.

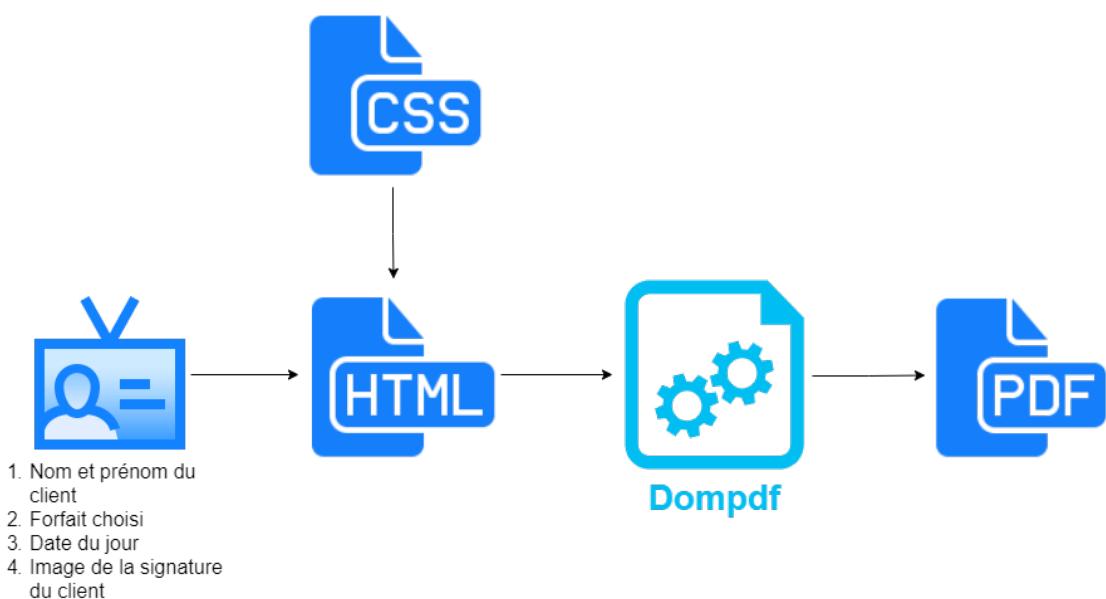


FIGURE 13 – Représentation de la création de document avec Dompdf

J'ai utilisé Dompdf pour la génération des conditions d'inscription après avoir cherché plusieurs alternatives. En effet, j'avais commencé mes recherches avec des outils permettant de modifier directement un PDF afin

d'ajouter la signature du client sur une condition d'inscription déjà créée en format PDF. Lors de mes recherches, j'ai trouvé plusieurs outils permettant la modification de PDF, mais, malheureusement, tous ces outils étaient payants. Je me suis donc orienté sur une alternative en imaginant une manière de créer un PDF from scratch. J'ai donc effectué des recherches et trouvé les librairies PHP comme TCPDF ou FPDF. La réalisation des conditions d'inscription from scratch étant assez fastidieuses avec ces outils et ceux-ci n'étant pas maintenus, j'ai réfléchi à une façon plus simple et plus pratique. C'est alors que je me suis demandé s'il était possible de convertir du HTML et CSS en PDF. En effet, je me suis dit que l'élaboration et l'organisation d'un rendu HTML et CSS était si pratique que s'il existerait une façon de convertir ce résultat en PDF, mon problème serait réglé. C'est alors que j'ai effectué des recherches et j'ai découvert l'existence de la librairie Dompdf.

#### 7.1.4.3 PHPMailer

PHPMailer est une librairie logicielle d'envoi d'e-mails en PHP. Cette librairie m'a permis d'envoyer les différents e-mails lors d'événements comme : la création de compte, la création et l'ajout de document PDF et la prise de rendez-vous. Envoyer un e-mail en code natif exigeant de très bonnes connaissances en normes SMTP<sup>9</sup>, en format d'e-mail et en vulnérabilités d'injection de spammer, PHPMailer permet de faciliter toutes ces interactions en proposant une solution d'envoi d'e-mail facile en PHP. J'ai choisi d'utiliser PHPMailer car celui-ci est le plus populaire au monde pour l'envoi d'e-mail depuis PHP et qu'il propose beaucoup de fonctionnalités permettant de ne pas perdre trop de temps sur son développement.

#### 7.1.5 Tests unitaires

Afin de tester le bon fonctionnement de mon API REST, j'ai effectué ce que l'on appelle des tests unitaires. Le test unitaire est une procédure permettant de vérifier le bon fonctionnement d'une partie précise d'un logiciel. Pour ma part, il m'a permis de vérifier les différents cas d'utilisations et d'exceptions de mes différents endpoints. Pour donner un exemple, lorsque

---

9. Simple Mail Transfer Protocol est un protocole de communication utilisé pour transférer le courrier électronique vers les serveurs de messagerie électronique

nous souhaitons ajouter un chien pour un utilisateur alors que nous n'avons pas la permission de le faire, j'ai réalisé un test unitaire vérifiant bien que lorsque cela se produit, l'API REST retourne bien une réponse HTTP avec comme code **403 Forbidden** ainsi que le message d'erreur **Vous n'avez pas les permissions**.

#### 7.1.5.1 Postman

Pour réaliser ces tests unitaires, je me suis servi de l'application Postman qui comme je vous l'ai expliqué auparavant, permet de réaliser des tests unitaires. Avec l'outil Postman, j'ai réalisé plus de cent tests unitaires qui m'ont permis de vérifier le bon fonctionnement de mon API REST. Malgré ce nombre conséquent, je ne pourrai pas m'avancer sur le fait que mon API REST est 100% fiable et sécurisée. En effet, je suppose qu'il reste tout de même quelques cas précis qui ne sont pas gérés dans le cas où quelqu'un voudrait volontairement nuire au bon fonctionnement de celle-ci.

Postman permet également de générer une documentation contenant l'intégralité des informations de toutes les requêtes que j'ai développées. La documentation générée par Postman est très pratique car elle permet également de fournir un exemple de requête pour un langage choisi.

Cette documentation est disponible depuis l'adresse suivante : <https://documenter.getpostman.com/view/9321886/TzJuAd3h>

#### 7.1.6 Outils utilisés

##### 7.1.6.1 PhpDocumentor

PhpDocumentor est un outil de génération de documentation écrit en PHP permettant d'analyser le code source d'un projet PHP afin d'en générer une documentation au format HTML et PDF entre autres. En effet, lors de la réalisation de mon API REST, j'ai respecté la structure de documentation de cet outil afin de pouvoir générer une documentation au format HTML permettant d'accéder aux descriptions de mes différentes classes et ainsi que leurs méthodes.

J'ai hébergé cette documentation sur mon hébergement WEB infomaniak.

Cette documentation est disponible depuis l'adresse suivante : <https://api-rest-douceur-de-chien.boreljaquet.ch/docs/>

## 7.2 PWA

Une PWA de son nom *Progressive Web App* est une application WEB constituée de page WEB qui peut apparaître de la même manière qu'une application mobile traditionnelle sans engendrer les contraintes de celle-ci, comme sa soumission aux App-Stores, sa comptabilité autant sur IOS que sur Android ainsi que son utilisation importante de mémoire. En effet, les éducateurs canins de la société comptent utiliser leur tablette ou leur téléphone afin d'accéder à l'application et y insérer les différentes informations des clients. De ce fait, il fallait que mon application soit accessible autant sur grand écran que petit écran.

### 7.2.1 Vue

Lors de l'élaboration de mon cahier des charges, je ne comptais pas utiliser de framework particulier afin de réaliser le frontend de mon travail de diplôme. En effet, je comptais uniquement utiliser Bootstrap qui bien que celui-ci soit tout de même un framework, n'implique pas vraiment de réelle condition ou restriction à son développement comparé à d'autres frameworks comme Vue ou React qui ceux-ci, demandent de respecter une certaine syntaxe plutôt stricte. C'est vers la fin du développement de mon API REST que j'ai commencé à réfléchir au développement de mon frontend. J'ai profité de ce travail de diplôme pour découvrir de nouvelles technologies ainsi que de sortir de ma zone de confort en utilisant un framework que je n'avais jamais utilisé auparavant, du moins, jamais approfondi. Pour réaliser le frontend de mon travail de diplôme, j'ai donc utilisé le framework Vue. J'ai choisi d'utiliser Vue car je l'avais déjà partiellement utilisé lors de ma 1ère année de technicien, qui n'oblige pas l'utilisation d'une extension syntaxique de JavaScript comme le fait React par exemple et que celui-ci est connu pour proposer une courbe d'apprentissage progressive.

### 7.2.1.1 Description

Comme ses confrères React et Angular, Vue propose une approche orientée composant, permettant de réaliser des applications WEB monopages. Une application WEB monopage est une application WEB accessible via une seule page WEB évitant le chargement d'une nouvelle page à chaque action afin de fluidifier l'interaction des utilisateurs. En effet, Vue permet l'élaboration de composants qui existent indépendamment les uns des autres permettant de découper et structurer son application. La principale caractéristique d'un composant Vue est que celui-ci est découpé en 3 parties distinctes :

- Le template qui est principalement le code HTML
- Le script qui est le code JavaScript
- Le style qui est le code CSS

## 7.2.2 Librairies utilisées

### 7.2.2.1 BootstrapVue

Lors du développement de la toute première page de mon application, je me suis tout naturellement dirigé vers la page officielle de Bootstrap afin de l'intégrer et l'utiliser dans mon projet Vue. C'est alors que j'ai découvert qu'il existait un plugin Vue permettant de faciliter l'intégration et l'utilisation de Bootstrap dans un projet Vue s'intitulant BootstrapVue. J'ai utilisé BootstrapVue et plus précisément Bootstrap afin de développer mon application de manière responsive, c'est-à-dire de manière à être aussi bien affiché sur grand écran que sur petit écran. En effet, Bootstrap permet d'utiliser un système de grille permettant d'organiser les différents éléments d'une page WEB afin de permettre à ceux-ci d'être facilement lisibles sur petit écran.

### 7.2.2.2 Vue Router

Vue Router est le routeur officiel de Vue. Il permet la création et l'utilisation d'application WEB monopage aussi appelée SPA pour *Single Page Application*. En effet, mon application WEB monopage permet de toujours garder la même barre de navigation ainsi que le même pied de page pour toutes les pages de mon application, c'est-à-dire que uniquement le contenu principal de la page se verra être modifié sans recharger la page dans son intégralité.



FIGURE 14 – Représentation du fonctionnement de l'application WEB monopage

Pour résumer, Vue router s'occupe de choisir quel composant sera affiché comme contenu principal ainsi que de gérer les différentes redirections qui doivent être réalisées comme sur les boutons de ma barre de navigation par exemple.

### 7.2.2.3 Vuex

Vuex est un gestionnaire d'état et une librairie pour les applications Vue. Un gestionnaire d'état permet de stocker et manipuler des données centralisées afin que celles-ci soient accessibles par l'intégralité des composants Vue d'une application dans le but de dicter son fonctionnement général. Dans mon cas, l'état que j'avais besoin de centraliser dans mon application était l'authentification, c'est-à-dire que j'avais besoin de savoir quel utilisateur était authentifié et quel type de rôle il avait. Lors de la connexion de mon application, mon gestionnaire d'état va stocker l'api token ainsi que le rôle de l'utilisateur afin que ces données soient accessibles pour l'intégralité de mes composants dans le but d'afficher les bons éléments ainsi que de gérer les différents droits et accès de mon application.

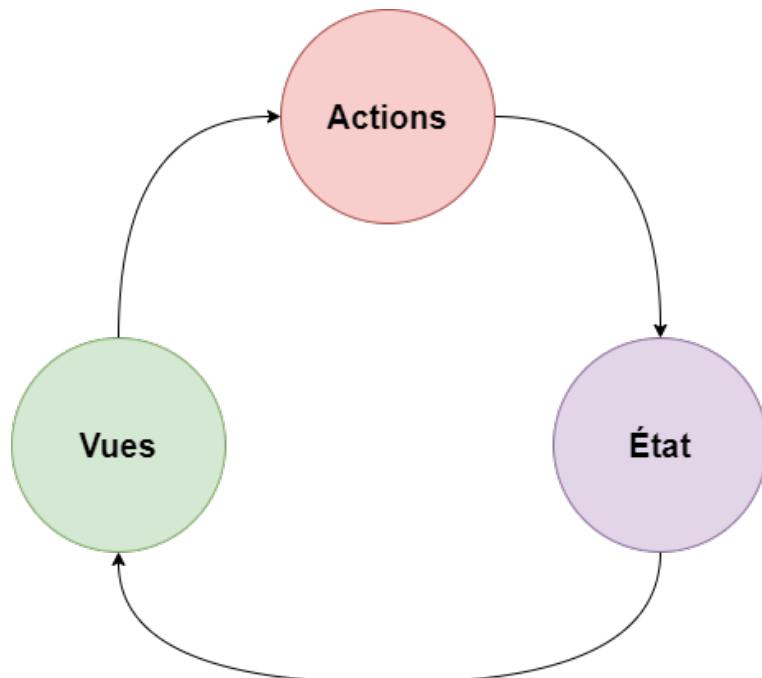


FIGURE 15 – Représentation du fonctionnement du gestionnaire d'état

Dans mon cas, les actions de mon gestionnaire d'état sont l'inscription, la connexion ainsi que la déconnexion. Ces actions vont modifier l'état de l'application, c'est-à-dire déterminer si l'utilisateur est connecté et si celui-ci est un client ou un éducateur canin. L'état quant à lui, va permettre à l'application de savoir quand et comment afficher les différentes fonctionnalités pour chaque utilisateur.

#### 7.2.2.4 Axios

Axios est une librairie JavaScript permettant d'exécuter des requêtes HTTP asynchrones en JavaScript. La caractéristique principale d'Axios est que cette librairie est basée sur les promesses. Les promesses sont une représentation de l'état d'une opération asynchrone. En effet, une requête HTTP réalisée avec Axios va retourner de manière asynchrone, une promesse que nous allons par la suite pouvoir utiliser. Par exemple, lors de la création d'une condition d'inscription pour un client, je souhaitais ne pas bloquer l'utilisateur durant la génération de celle-ci. Pour ce faire, j'ai donc effectué une requête HTTP asynchrone avec Axios afin que celle-ci renvoie une promesse qui serait résolue une fois les conditions d'inscription créées afin de permettre à l'utilisateur de ne pas être bloqué. Dans le cas contraire, la promesse peut également retourner une erreur si quelque chose c'est mal passé. J'ai choisi d'utiliser des promesses avec Axios et non des fonctions de callback avec Ajax ou XMLHttpRequest car à mon goût, celles-ci rendent le code difficile à lire et de ce fait, difficile à maintenir.

#### 7.2.2.5 Signature Pad

Lors de l'étude des technologies précédant le début de mon travail de diplôme, je m'étais renseigné sur une librairie permettant de dessiner des courbes dans une certaine zone. J'avais donc trouvé la librairie Responsive Sketchpad. Comme son nom l'indique, Responsive Sketchpad est une librairie fournissant une zone de dessin s'adaptant à l'affichage et ne nécessitant aucune dépendance. Je me suis donc tout naturellement dirigé vers l'implémentation de cette option dans mon application afin de répondre aux critères de celle-ci. En effet, mon application nécessite une zone de dessin afin de permettre aux clients de signer leurs conditions d'inscription ainsi que de permettre aux éducateurs canins de prendre des notes graphiques. C'est lors de la réalisation de la fonctionnalité de prise de notes graphiques, que je me suis rendu compte que la librairie Responsive Sketchpad ne permettait pas de charger une image existante sans passer par un certain format JSON représentant les positions des points dessinés. C'est alors que je me suis renseigné et que j'ai trouvé la librairie Signature Pad permettant de charger une image existante en base64<sup>10</sup>.

---

10. Forme de codage d'une information en utilisant 64 caractères

Signature Pad est une librairie JavaScript permettant de dessiner des courbes en reproduisant la typographie d'une plume afin d'imiter une signature. Je me suis renseigné et j'ai appris l'existence d'un composant Vue développé par la communauté permettant de facilement intégrer Signature Pad.



FIGURE 16 – Exemple de signature réalisé avec Signature Pad

#### 7.2.2.6 FullCalendar

FullCalendar est une librairie JavaScript très complète permettant d'afficher des calendriers de différentes manières : affichage mensuel, affichage hebdomadaire, affichage journalier, etc. FullCalendar permet également de charger ces calendriers avec des données afin d'y ajouter des événements, le tout en proposant une tonne de fonctionnalités, toutes plus pratiques les unes que les autres. FullCalendar m'a permis d'afficher les plannings des éducateurs canins de la société ainsi que leurs rendez-vous. De plus, FullCalendar propose une intégration Vue facilitant beaucoup son utilisation avec celui-ci.

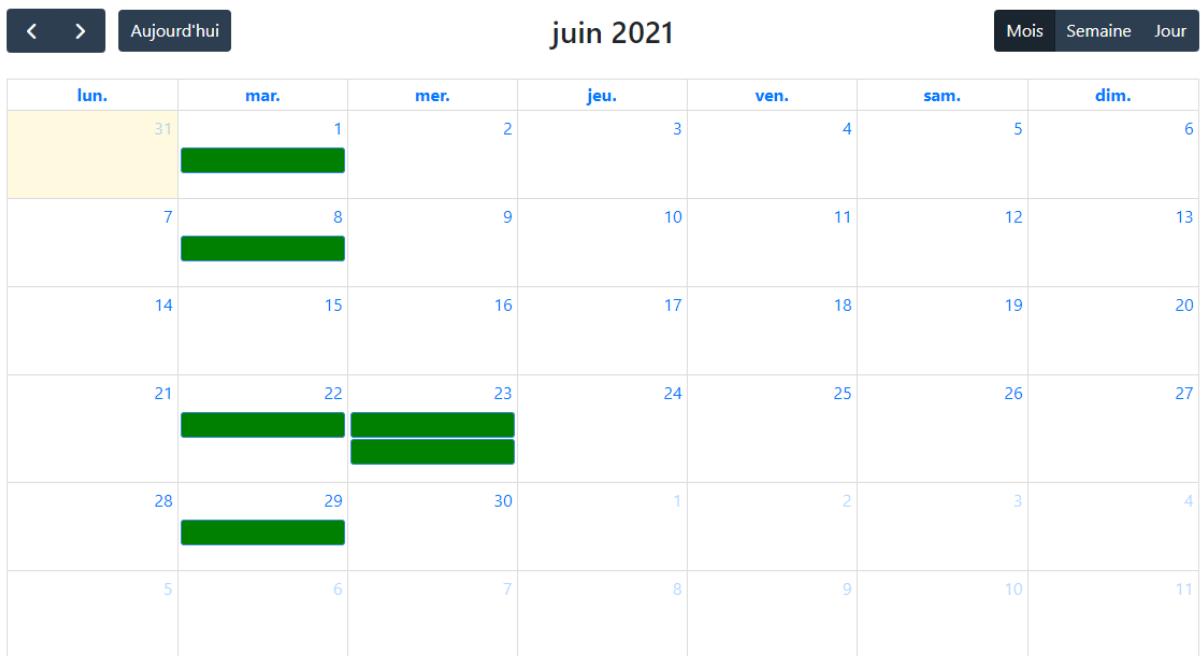


FIGURE 17 – Exemple de calendrier affiché avec FullCalendar

### 7.2.2.7 Moment.js

Moment.js est une librairie JavaScript permettant de manipuler des dates et heures en JavaScript de manière rapide et efficace. En effet, la manipulation des dates est souvent fastidieuse et de ce fait, Moment.js propose plusieurs fonctionnalités permettant de pallier à ce problème. Dans mon cas, j'ai utilisé Moment.js uniquement dans le but de formater les différentes dates de mon application afin de faciliter leurs utilisations.

### 7.2.2.8 AlertifyJS

AlertifyJS est une librairie JavaScript permettant d'afficher des boîtes de dialogue et des notifications en JavaScript. J'ai utilisé un composant développé par la communauté afin d'utiliser AlertifyJS dans le but d'avertir l'utilisateur lors du succès ou l'échec de ses actions de manière esthétique et non bloquante en affichant une notification. AlertifyJS m'a permis de ne pas perdre trop de temps sur l'affichage des notifications de mon application.

### 7.2.2.9 reCAPTCHA

reCAPTCHA est le système de détection automatisée d'utilisateurs appartenant à Google. J'ai utilisé reCAPTCHA afin d'empêcher qu'un robot spam ma fonctionnalité d'inscription dans le but de surcharger ma base de données. Pour ce faire, je me suis renseigné et j'ai trouvé un composant Vue développé par la communauté permettant d'intégrer reCAPTCHA. Je me suis ensuite rendu sur la documentation officielle afin de me créer un compte ainsi que de me documenter par rapport à ce service proposé par Google.

Pour fonctionner, Google fournit deux clefs reCAPTCHA. La première est la clef pour l'intégration côté client et la seconde est la clef secrète pour l'intégration côté serveur afin de permettre la communication entre le côté client et le service reCAPTCHA. L'intégration côté client reCAPTCHA va fournir un jeton de réponse utilisateur qui doit être soumis par le service reCAPTCHA afin de valider ou non l'action. Pour ce faire, j'ai envoyé ce jeton lors de l'endpoint d'inscription de mon API REST afin de soumettre et vérifier la validité de ce jeton. Si le jeton est validé par le service reCAPTCHA, alors l'inscription s'effectue. Dans le cas contraire, une erreur est retournée.

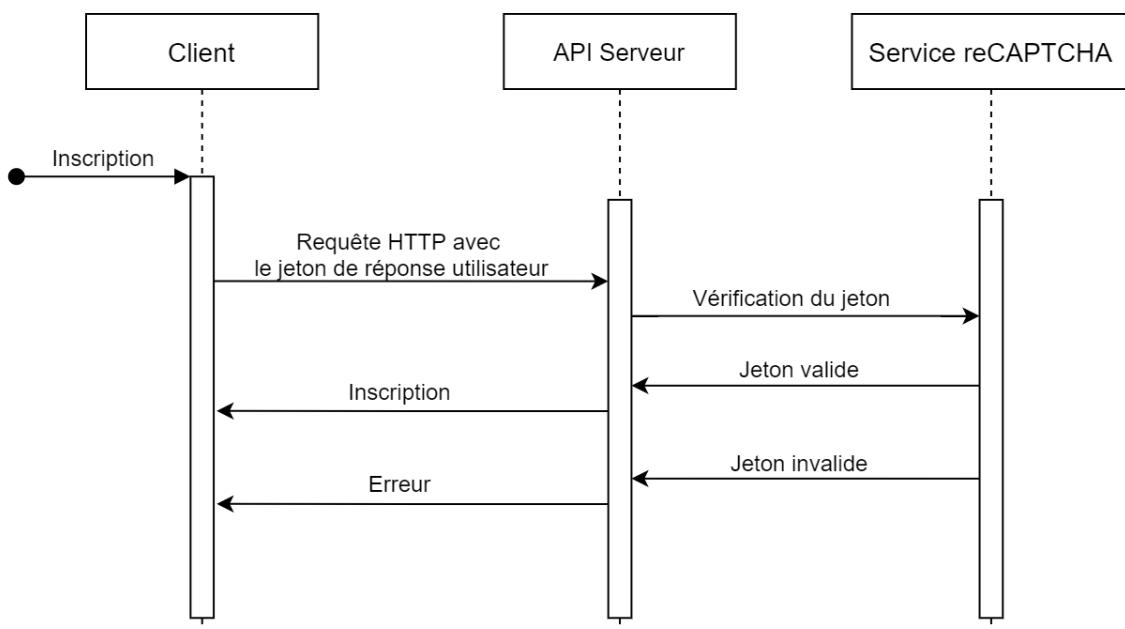


FIGURE 18 – Diagramme de séquence reCAPTCHA

### 7.2.3 Tests fonctionnels

Afin de tester le bon fonctionnement de ma PWA, j'ai effectué des scénarios de tests sur celle-ci. Comparés aux tests sur mon API REST, les tests que j'ai réalisés, m'ont principalement permis de vérifier que à chaque modification de mon application WEB, il n'y aurait aucun effet de bord apportant des bugs. En effet, j'ai réalisé des scénarios de tests avec Katalon Recorder qui m'ont permis de vérifier quand je le souhaitais, la plupart des scénarios d'utilisation de ma PWA de manière rapide.

#### 7.2.3.1 Katalon Recorder

Pour réaliser ces tests fonctionnels, je me suis servi de l'application Katalon Recorder proposé par Katalon Studio lui-même construit sur les cadres d'automatisation open-source Selenium. Proposé par mon maître de diplôme M. Mathieu, Katalon Recorder est une extension de navigateur disponible sur Google Chrome et Mozilla Firefox permettant d'enregistrer des actions et interactions utilisateurs afin de générer des scripts de test. Katalon Recorder m'a permis de tester les principales fonctionnalités de ma PWA comme la prise de rendez-vous pour un client ou la prise de notes graphiques pour l'éducateur canin par exemple. L'extension Katalon Recorder proposant une interface graphique et une prise en main très pratique, m'a permis de ne pas perdre trop de temps sur la réalisation de la plupart des scénarios de tests de ma PWA.

## 7.2.4 Maquettes effectives

### 7.2.4.1 Page d'accueil

The image displays two side-by-side screenshots of a website for 'Douceur de chien'.  
The left screenshot shows the homepage. It features a large blue header bar with the logo 'Douceur de chien' and navigation links 'Accueil', 'À propos', and 'Agenda'. On the right is a 'Connexion' button. Below the header is a large circular image of a smiling grey dog. To the left of the dog is the text 'Douceur de chien' and 'Educateur & comportementaliste canin'. Below this are links for 'Vous voulez prendre rendez-vous?' and 'Accéder aux calendriers des éducateurs canins', along with a 'Calendriers' button.  
The right screenshot shows a mobile version of the same website. It has a similar layout with the 'Douceur de chien' logo at the top, followed by the text 'Douceur de chien' and 'Educateur & comportementaliste canin'. It includes the same links for appointments and calendars, and a 'Calendriers' button. A three-line menu icon is visible in the top right corner.

FIGURE 19 – Page d'accueil

### 7.2.4.2 Page À propos



**Bonjour, je suis Boris Gourdoux**

+ de 300 couples Maître/chien à mon actif

Depuis ma toute petite enfance, ma relation avec le chien a été salvatrice. Elle m'a permis de dépasser bien des difficultés et de trouver une véritable complicité. Diplômé et spécialisé en rééducation et comportement, vous bénéficiez de toute mon expertise et expérience dans l'éducation canine, fruit de nombreuses formations dans ce qui se fait de mieux dans le domaine. L'objectif chez Douceur De Chien est de vous permettre de créer un lien renforcé et durable avec votre compagnon à 4 pattes et ceci en utilisant uniquement de la patience, de la cohérence et beaucoup de passion. Ainsi chez Douceur De Chien vous n'obtiendrez aucun conseil sur l'utilisation de colliers étrangleur ou même de la force pour éduquer votre chien.



**Une méthode moderne**

Je base ma méthode en respectant les besoins et l'émotionnel du chien, c'est pourquoi je met l'accent sur une coopération totale de ce dernier en lui permettant de faire ces propres choix et ainsi VOUS permettre d'obtenir de votre chien, écoute et attention en toutes circonstances



Depuis ma toute petite enfance, ma relation avec le chien a été salvatrice. Elle m'a permis de dépasser bien des difficultés et de trouver une véritable complicité. Diplômé et spécialisé en rééducation et comportement, vous bénéficiez de toute mon expertise et expérience dans l'éducation canine, fruit de nombreuses formations dans ce qui se fait de mieux dans le domaine. L'objectif chez Douceur De Chien est de vous permettre de créer un lien renforcé et durable avec votre compagnon à 4 pattes et ceci en utilisant uniquement de la patience, de la cohérence et beaucoup de passion. Ainsi chez Douceur De Chien vous n'obtiendrez aucun conseil sur l'utilisation de colliers étrangleur ou même de la force pour éduquer votre chien.

FIGURE 20 – Page À propos

### 7.2.4.3 Page Agenda



**Agenda des éducateurs canins de la société**

Rendez-vous libre

juin 2021

lun.	mar.	mer.	jeu.	ven.	sam.	dim.
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4



**Agenda des éducateurs canins de la société**

Rendez-vous libre

juin 2021

lun.	mar.	mer.	jeu.	ven.	sam.	dim.
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4

FIGURE 21 – Page Agenda

#### 7.2.4.4 Page Connexion

The figure displays two side-by-side screenshots of the Douceur de Chien website's login page. Both screenshots have a blue header bar at the top containing the logo (a dog and person icon), the words "Accueil", "À propos", "Agenda", and a "Connexion" button. The left screenshot shows a "Se connecter" section with fields for "Adresse email" (Email address) and "Mot de passe" (Password), and a "Connexion" (Login) button. Below this, there is a link "Nouveau sur ce site ? S'inscrire". The right screenshot shows a similar layout but lacks the "Nouveau sur ce site ? S'inscrire" link.

**Se connecter**

Nouveau sur ce site ? [S'inscrire](#)

Adresse email:

Mot de passe :

[Connexion](#)

**Se connecter**

Nouveau sur ce site ? [S'inscrire](#)

Adresse email:

Mot de passe :

[Connexion](#)

**Contact**

Téléphone  
**06.24.00.33.68**

Email  
[contact@douceurdechien.com](mailto:contact@douceurdechien.com)

Zone d'intervention  
Montréal et alentours

©2021 Douceur de chien - [politique de confidentialité](#) - [conditions d'utilisation](#)

**Contact**

Téléphone  
**06.24.00.33.68**

Email

FIGURE 22 – Page Connexion

#### 7.2.4.5 Page Inscription

The figure displays two side-by-side screenshots of the Douceur de Chien website's inscription page. Both screenshots have a blue header bar at the top containing the logo (a dog and person icon), the words "Accueil", "À propos", "Agenda", and a "Connexion" button. The left screenshot shows a "S'inscrire" section with fields for "Adresse e-mail", "Prénom", "Nom de famille", "Numéro de téléphone", "Adresse", "Mot de passe", "Confirmation du mot de passe", and "Captcha". Below these fields are two checkboxes: "Je suis majeur" and "Je accepte les conditions générales d'utilisation". The right screenshot shows a similar layout but lacks the checkboxes at the bottom.

**S'inscrire**

Déjà membre ? [Se connecter](#)

Adresse e-mail :

Prénom :

Nom de famille :

Numéro de téléphone :

Adresse :

Mot de passe :

Confirmation du mot de passe :

Captcha :

Le code ci-dessus n'est pas valable

**S'inscrire**

Déjà membre ? [Se connecter](#)

Adresse e-mail :

Prénom :

Nom de famille :

Numéro de téléphone :

Adresse :

Mot de passe :

FIGURE 23 – Page Inscription

### 7.2.4.6 Page Administration pour les éducateurs canins

Nom	Prenom	Chiens	Action
Alfari	Alana	Afficher	
Borel-Jaquet	Jonathan	Afficher	
Borel-Jaquet	Thierry	Afficher	
Burger	Flo	Afficher	
Copter	Client	Afficher	
Donatallo	Daniel	Afficher	
Dubois	Eric	Afficher	
Lapon	Alexis	Afficher	

FIGURE 24 – Page Administration pour les éducateurs canins

### 7.2.4.7 Page des informations du client pour les éducateurs canins

Nom complet	Alana Alfari
Adresse e-mail	Alfari@eduge.ch
Numéro de téléphone	0761167812
Adresse de domicile	Route de Frontenex 100 1208 Genève

Nom	Jaya
Race	Rhodesian Ridgeback

FIGURE 25 – Page des informations du client pour les éducateurs canins

### 7.2.4.8 Page des informations du client pour les clients

FIGURE 26 – Page des informations du client pour les clients

### 7.2.4.9 Page des rendez-vous du client pour les éducateurs canins

FIGURE 27 – Page des rendez-vous du client pour les éducateurs canins

### 7.2.4.10 Page des rendez-vous du client pour les clients

FIGURE 28 – Page des rendez-vous du client pour les clients

### 7.2.4.11 Page Mon planning pour les éducateurs canins

FIGURE 29 – Page Mon planning pour les éducateurs canins

### 7.2.4.12 Page Mes rendez-vous pour les éducateurs canins

**Mes rendez-vous**

Planifier un rendez-vous

juin 2021

lun.	mar.	mer.	jeu.	ven.	sam.	dim.
31		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11						

**Contact**

Téléphone      Email      Zone d'intervention

FIGURE 30 – Page Mes rendez-vous pour les éducateurs canins

## 8 Conclusion

## 8.1 Planning prévisionnel et planning effectif

Lors de l'élaboration du cahier des charges de mon travail de diplôme, j'ai réalisé un planning prévisionnel afin de réfléchir aux différentes étapes que j'allais devoir réaliser. Bien évidemment, il est impossible de prédire à la perfection le déroulement d'un tel travail qui plus est, réalisé seul. Toutefois, ce qui est intéressant avec la réalisation d'un planning prévisionnel, c'est que celui-ci permet d'être comparé avec un planning effectif. C'est justement ce que j'ai réalisé durant l'intégralité de mon travail de diplôme. En effet, à chaque fin de journée, j'ai complété mon planning effectif afin d'y noter les tâches que j'avais réalisées dans la journée.

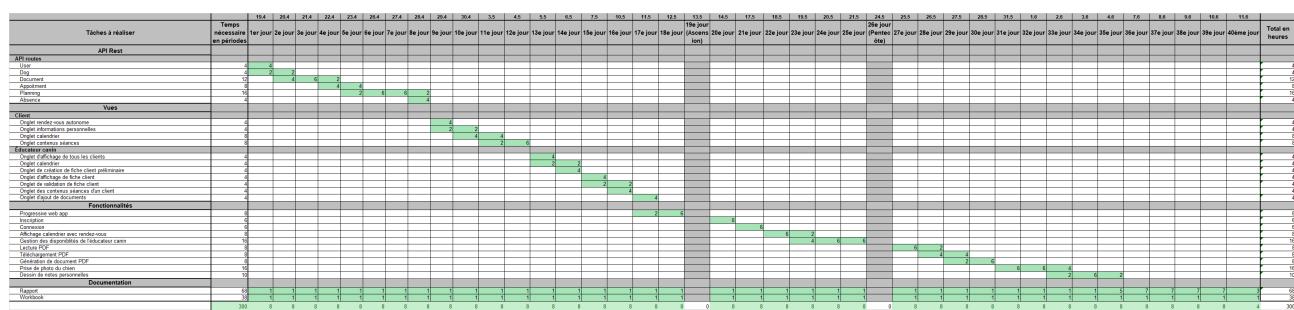


FIGURE 31 – Planning prévisionnel du travail de diplôme

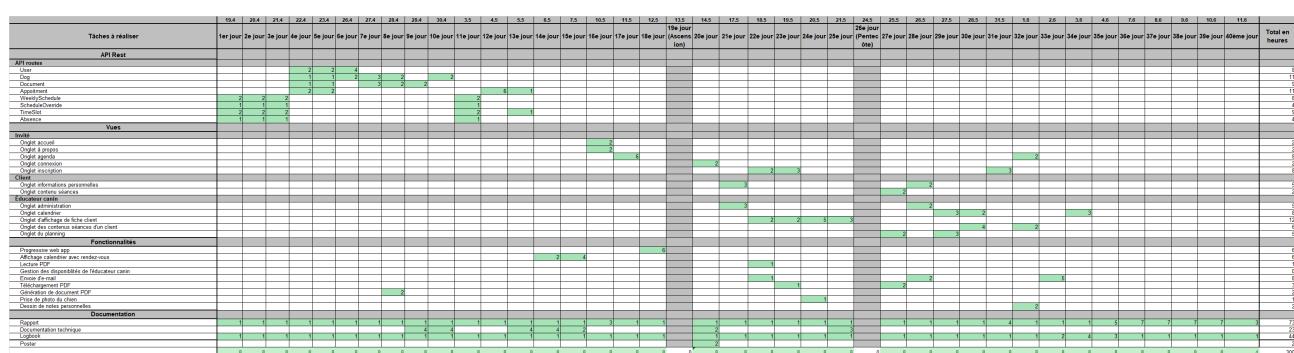


FIGURE 32 – Planning effectif du travail de diplôme

Comme on peut le voir, il y a plus de tâches dans le planning effectif que dans le planning prévisionnel. En effet, lors de mon développement, j'ai ajouté ou remplacé des fonctionnalités et des éléments auxquels je n'avais pas pensé lors de la réalisation de mon cahier des charges. Par exemple, lors du développement de mon API REST, je me suis rendu compte que la fonctionnalité d'édition de planning pour les éducateurs canins n'était pas assez complète. J'ai donc réfléchi et j'ai rajouté des tables supplémentaires dans ma base de données qui ont de ce fait, demandé plus de temps de développement lors de la réalisation de mon API REST.

## 8.2 Problèmes rencontrés

### 8.2.1 Structure de l'API REST

Au début de mon travail de diplôme et plus précisément lors de la réalisation de mon API REST, j'ai été confronté à un problème de conception avec celle-ci. En effet, étant de nature anxieux, j'ai commencé son développement de manière trop précipité afin de pouvoir le plus rapidement possible passer au développement de ma PWA. Après la deuxième semaine de mon travail de diplôme, je me suis retrouvé avec un code pas structuré et difficilement maintenable. Me rendant compte de l'erreur de conception que j'avais réalisé, j'ai alors communiqué au plus vite avec mon professeur de diplôme M. Mathieu afin de lui demander son avis. M. Mathieu m'a alors été d'une grande aide en me proposant de découper mon API REST de manière objet. Trouvant cette approche très correcte et l'ayant déjà pratiquée lors des précédentes années, l'approche objet était comme une évidence pour moi. Malheureusement, je n'y avais pas pensé lors de mon développement.

Le développement de mon API REST étant dorénavant terminé, je suis très satisfait de sa conception qui est maintenant très compréhensible et de ce fait, maintenable. En effet, j'ai pu être certain de cette information, car l'un de mes camarades de classe ayant remarqué mon implication et souhaitant développer sa propre API, m'a demandé, s'il pouvait s'inspirer de ma structure. Après que celui-ci l'ait intégrée, il m'a confirmé que son utilisation et sa maintenance étaient très pratiques.

### 8.2.2 Service worker de la PWA

Lors de l'élaboration de ma PWA, j'ai été confronté à une complexité que je n'imaginais pas aussi importante. En effet, la réalisation d'une PWA de manière optimale n'est vraiment pas évidente. En effet, une PWA est une application disponible autant sur petit écran que sur grand écran mais qui doit également permettre de rendre celle-ci utilisable avec peu, voire pas de signal Internet. Ce fonctionnement est possible grâce à un service worker.

Un service worker est un système venant se placer entre l'application et le réseau Internet. Celui-ci permet de mettre en cache les données essentielles de l'application afin que celle-ci soit utilisable sans accès à Internet. De plus, le service worker doit être capable de modifier les données en cache avec de nouvelles si celles-ci sont obsolètes. J'ai alors passé une journée entière à me renseigner afin d'implémenter cette fonctionnalité. Malheureusement, malgré la totalité de la journée passée dessus, le résultat n'était pas concluant et m'avait fait perdre beaucoup de temps. De ce fait, j'en ai parlé avec mon professeur de diplôme M. Mathieu et nous avons convenu ensemble qu'il ne fallait pas perdre trop de temps avec le développement de cette fonctionnalité.

### 8.2.3 CORS

Lors de l'élaboration de ma PWA et plus précisément des appels aux différents endpoints effectués dans celle-ci, j'ai appris l'existence d'une fonctionnalité CORS que je ne connaissais pas. CORS de son nom *Cross-origin ressource sharing* est un mécanisme qui permet à des ressources d'être récupérées par un autre nom de domaine que leurs origines.

Lors de la réalisation de ma PWA, j'ai rencontré un problème avec CORS. En effet, je me suis rendu compte que quelques unes de mes requêtes HTTP envoyées depuis ma PWA à l'API REST généraient d'abord une requête avec comme méthode de requête HTTP OPTIONS et que celles-ci retournaient une erreur. Je me suis donc renseigné et j'ai appris que pour les requêtes HTTP dites non simples (avec un en-tête "Authorization" par exemple), le navigateur enverrait d'abord une requête de "contrôle en amont" afin de déterminer si les en-têtes de la requête qui vont être envoyés correspondent bien avec les en-têtes du serveur. Par exemple, si un en-tête des en-têtes utilisés dans la requête ne figure pas dans la liste des en-têtes

autorisés par le serveur, le navigateur refusera d'envoyer la requête.

Pour régler ce problème, j'en ai parlé avec mon professeur de diplôme M. Mathieu avec qui nous avons convenu d'une solution de contournement qui consiste à retourner un code de succès lors de la réception de requête avec la méthode de requête HTTP OPTIONS.

### 8.3 Améliorations envisageables

Une des fonctionnalités importante que je compte inclure après le travail de diplôme est la vérification d'adresse e-mail. En effet, je souhaiterai faire en sorte que lors de l'inscription d'un client de manière autonome, celui-ci se verrait recevoir un e-mail afin de valider sa création dans le but de vérifier que son adresse e-mail soit correct. Quant aux comptes n'ayant pas été validés après deux jours, ceux-ci se verraient être supprimés. Toutefois, cette fonctionnalité nécessitera des changements dans ma base de données ainsi que dans mon API REST.

Comme expliqué dans les problèmes rencontrés, je n'ai pas eu le temps de réaliser la totalité des fonctionnalités nécessaires à l'utilisation de l'application de manière hors ligne. Une amélioration possible serait d'approfondir cette fonctionnalité afin de rendre l'application plus accessible.

La dernière fonctionnalité que je trouve importante d'inclure est l'amélioration de performance des requêtes SQL de ma base de données. Comme par exemple, la génération de planning d'un des éducateurs de la société. En effet, cette requête nécessitant l'implication de nombreuses tables de ma base de données, je me suis rendu compte après avoir effectué des tests que celle-ci n'était pas utilisable avec beaucoup de données. Le problème venant de sa conception initiale, il faudrait que je reprenne son fonctionnement depuis le début en imaginant peut-être un autre scénario d'utilisation.

## 8.4 Bilan personnel

À l'heure du bilan, je dirais que mon travail de diplôme a été une expérience très enrichissante, autant de manière technique que de manière professionnelle. En effet, c'était la première fois que je réalisais un travail de cette envergure, allant d'un développement backend et frontend jusqu'à la réalisation de tests ainsi que de la documentation de manière constante.

Ce travail m'a également permis de travailler sur moi-même. En effet, mon principal défaut est que je suis défaitiste et de ce fait, j'ai tendance à me surmener par peur de l'échec. Des personnes pourraient voir cela comme une qualité. Ces personnes ont raison dans le cas où le degré de surmenage n'est pas élevé. Dans mon cas, je suis passé par des moments de doute et de stress allant jusqu'à impacter mon sommeil.

Toutefois, je ne me suis pas arrêté sur ces détails et j'ai su surmonter ces épreuves grâce à l'assistance de ma classe, de mes professeurs et de mes proches qui ont su me motiver, m'aider et me soutenir. Ce travail m'a permis de me rendre compte de mes capacités.

Malgré cette appréciation que l'on peut trouver négative, je suis très fier et satisfait du travail que j'ai réalisé lors de ce travail de diplôme. Ce travail m'a permis d'avoir davantage confiance en moi et en mes capacités.

## 8.5 Remerciements

Je tiens à remercier l'intégralité des élèves de ma classe de technicien T.IS-E2A qui dans l'ensemble m'ont conseillé et aidé lors de mes différents problèmes et réflexions.

Je tiens également à remercier les différents professeurs qui ont eu un regard attentif et des conseils pertinents lors de la réalisation de mon travail de diplôme.

Pour finir, je tiens à remercier spécialement mon professeur de diplôme M. Mathieu qui a su me soutenir à la fois professionnellement avec ses larges connaissances en programmation, en situation professionnelle et à la fois moralement dans mes différents moments de doute ou de stress. En effet, M. Mathieu m'a toujours conseillé de prendre une journée par semaine dans

laquelle il fallait que je ne pense pas à mon travail de diplôme loin des écrans. De plus, par son aide professionnelle, M. Mathieu est un professeur avec qui nous pouvons se sentir à l'aise. Cette qualité m'a permis de le voir comme un soutien et non comme la personne souhaitant l'obtention ou non de mon diplôme de technicien.

## Table des figures

1	Diagramme de séquence prévisionnel . . . . .	18
2	Mindmap du projet Douceur de Chien . . . . .	19
3	Diagramme des actions des utilisateurs . . . . .	20
4	Planning prévisionnel 1/3 . . . . .	22
5	Planning prévisionnel 2/3 . . . . .	23
6	Planning prévisionnel 3/3 . . . . .	24
7	Interface graphique de l'application Laragon . . . . .	27
8	Interface de l'outil d'administration de base de données HeidiSQL . . . . .	28
9	Catégories des tâches Trello . . . . .	30
10	Trello du projet utilisant la méthode Kanban . . . . .	31
11	Représentation de l'utilité d'une API . . . . .	33
12	Représentation de communication d'une API REST . . . . .	35
13	Représentation de la création de document avec Dompdf . . . . .	37
14	Représentation du fonctionnement de l'application WEB monopage . . . . .	42
15	Représentation du fonctionnement du gestionnaire d'état . . . . .	43
16	Exemple de signature réalisé avec Signature Pad . . . . .	45
17	Exemple de calendrier affiché avec FullCalendar . . . . .	46
18	Diagramme de séquence reCAPTCHA . . . . .	47
19	Page d'accueil . . . . .	49
20	Page À propos . . . . .	50
21	Page Agenda . . . . .	50
22	Page Connexion . . . . .	51
23	Page Inscription . . . . .	51
24	Page Administration pour les éducateurs canins . . . . .	52
25	Page des informations du client pour les éducateurs canins . . . . .	52
26	Page des informations du client pour les clients . . . . .	53
27	Page des rendez-vous du client pour les éducateurs canins . . . . .	53
28	Page des rendez-vous du client pour les clients . . . . .	54
29	Page Mon planning pour les éducateurs canins . . . . .	54
30	Page Mes rendez-vous pour les éducateurs canins . . . . .	55
31	Planning prévisionnel du travail de diplôme . . . . .	56
32	Planning effectif du travail de diplôme . . . . .	56