

# Cahier des charges Douceur de Chien

---

## Tables des matières

---

- [Cahier des charges Douceur de Chien](#)
  - [Tables des matières](#)
  - [Objectif du projet](#)
  - [Description détaillée](#)
  - [Liste des fonctionnalités](#)
    - [Fonctionnalités de l'éducateur canin](#)
    - [Fonctionnalités du client](#)
  - [Scénarios d'utilisation](#)
    - [Prise en charge d'un nouveau client depuis l'application](#)
    - [Prise en charge d'un nouveau client par téléphone](#)
    - [Première rencontre physique avec le client](#)
    - [Rendez-vous avec le client](#)
  - [Étude des technologies](#)
    - [Application mobile ou application WEB](#)
    - [Backend](#)
      - [API REST](#)
      - [Génération de PDF](#)
      - [Envoie de mail](#)
    - [Frontend](#)
      - [Bootstrap](#)
      - [Dessin dans un canevas](#)
      - [Calendrier](#)
  - [Modèle de données](#)
    - [API](#)
    - [Diagramme de séquence](#)
  - [Maquettes de l'application](#)
    - [Client](#)
      - [Information personnelles](#)
      - [Calendrier](#)
      - [Contenu séances](#)
    - [Éducateur canin](#)
      - [Affichage de tout les clients](#)
      - [Calendrier](#)
      - [Création de fiche client préliminaire](#)
      - [Affichage fiche client](#)
      - [Validation de fiche client](#)
      - [Contenu séance d'un client](#)
      - [Ajout de document](#)
  - [Planning](#)
  - [Fonctionnalités envisagées mais non retenues](#)

- [Scan Bluetooth des puces sous-cutanées canine RFID](#)
- [Matériel et logiciel](#)
- [Librairies utilisées](#)
- [Dates importantes](#)
- [Livrables](#)

## 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 signé/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.

## 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 planifié et à une liste contenant tous ses actuels clients où 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 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 signé
  - Fiches récapitulatif du cours
  - Autres

L'éducateur doit éditer son horaire afin de permettre au futur client de planifier leurs propres rendez-vous.

L'éducateur canin doit procéder à une création préliminaire d'une fiche cliente, le client concerné sera averti par mail qu'une fiche à été créé avec ses informations, il devra alors valider son compte pour pouvoir y accéder. Lors du premier rendez-vous, l'éducateur canin doit compléter la fiche cliente avec les données qui nécessite 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 personnel de l'éducateur canin du cours sous format texte
- Note personnel de l'éducateur canin du cours sous format graphique
- Note récapitulatif 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 planifier son propre rendez-vous respectant les horaires spécifié par l'éducateur canin depuis le calendrier disponible sur la page d'accueil de l'application, il rentrera lui-même ses propres informations personnelles que l'éducateur canin validera lors de la première rencontre. Le client peut également se connecter à l'application après s'être inscrit pour accéder à ses informations personnelles ainsi que ses documents.

## Liste des fonctionnalités

---

### Fonctionnalités de l'éducateur canin

- Connexion à l'application
- Accès au calendrier de ses rendez-vous
- Édition de l'horaire personnel
- Affichage de tout les clients avec photo du chien et nom de son maitre
- 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 de contenu séance d'un client depuis la recherche spécifique ou le calendrier de rendez-vous

### 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 personnelles
- Accès à ses différents contenus séances

## Scénarios d'utilisation

---

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

1. Le client accède à l'application et sélectionne sur le calendrier de la page d'accueil son propre rendez-vous respectant les horaires que l'éducateur canin aura préalablement configuré.
2. Il devra rentrer différentes informations comme :
  - Son nom
  - Son prénom
  - Son numéro de téléphone
  - Son adresse e-mail
  - Son adresse de domicile
  - La date de naissance du chien
  - La race du chien
  - Le sexe du chien
  - Le nom du chien
  - Son mot de passe

Le client recevra un mail pour valider la création de son compte ainsi que du rendez-vous.

3. L'éducateur canin recevra un mail avec toute les données du rendez-vous qu'il pourra également consulter sur son calendrier personnel.

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

1. Le client appel l'éducateur canin avec son téléphone car il a besoin de ses services.
2. L'éducateur canin va se rendre sur l'application et se connecter avec ses identifiants.
3. Il va se rendre sur l'interface de création d'une nouvelle fiche client.
4. Il va y rentrer les informations personnelles du client communiqué par téléphone :
  - Nom du client
  - Prénom du client
  - Téléphone du client
  - Adresse e-mail du client
  - Adresse du domicile du client
  - Date de naissance du chien
  - Race du chien
  - Sexe du chien
  - Nom du chien
5. Il aura accès à son calendrier personnel afin de visualiser à quelle date il peut se rendre au domicile du client.
6. Le client ainsi que l'éducateur se mettront d'accord sur la date du rendez-vous.
7. L'éducateur canin sélectionnera cette date dans le calendrier.
8. Une fois la fiche client préliminaire avec la date du premier rendez-vous remplis, un mail sera envoyé au client afin qu'il puisse créer son compte dans l'application afin d'avoir accès à différentes fonctionnalités.

## Première rencontre physique avec le client

1. L'éducateur canin se rend au domicile du client à la date spécifié lors de l'appel téléphonique ou du rendez-vous pris depuis l'application.
2. Il va se rendre sur l'application et se connecter avec ses identifiants.

3. Il va rechercher le client grâce à son nom et accéder à sa fiche cliente précédemment créée lors de l'appel téléphonique ou du rendez-vous pris depuis l'application.
4. Il va montrer les données personnelles du client et lui demander une vérification de celle-ci.
5. Si elles sont fausses, modification de celle-ci.
6. Si elles sont correctes, l'éducateur canin devra prendre une photo du chien ainsi que de rentrer manuellement les 15 chiffres du code de la puce sous-cutanée du chien (si existante).
7. Il pourra ensuite sauvegarder cette version final de la fiche client.

## Rendez-vous avec le client

1. L'éducateur canin peut à tout moment lors d'un rendez-vous, accéder à la fiche du client 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 récapitulatif 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 :
  1. Choisir le forfait qu'il désire.
  2. Ajouter sa signature depuis l'application.
  3. Visualiser la version final des conditions d'inscriptions.
  4. Validée s'il est d'accord en cochant une case "Lu et approuvé".
  5. Les conditions d'inscription sous format PDF ainsi qu'une génération automatique d'une facture sous format PDF sera ajouté au dossier partagé du client et envoyé par mail au client.

## Étude des technologies

---

### Application mobile ou application WEB

Après avoir discuté avec le client de mon application, nous avons convenu qu'il lui fallait 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, ceux-là 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 est trop restrictive et pas réellement pratique. 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 affiché de la même manière qu'une application mobile.

## Backend

### API REST

Afin de gérer les différentes données de mon application, j'ai décidé de développer une API respectant l'ensemble de principes architecturaux REST me permettant de rendre mes données plus facilement utilisable lors de futurs projets en lien avec mon travail de diplôme. Pour la réalisation de cette API REST, j'ai été tenté plusieurs choix de framework, 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 tous 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 framework PHP Yii et Laravel à départagé, 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 utile directement intégré au Framework. Malgré toutes ces qualités, le choix d'un Framework PHP quelconque pour mon backend est source de problème et d'incompréhension, de ce fait, je compte développer le backend de mon application en PHP brut.

## Génération de PDF

Afin de pouvoir créer manuellement les différents documents PDF de mon application, j'ai recherché différentes façons de procéder. Après mettre rendu compte que la plupart des outils existants étaient payant (PDFTron, radaeePDF), je me suis dirigé vers une solution gratuite et open-source de génération de PDF en PHP proposé par mon maitre 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](#)

## Envoie de mail

Afin d'envoyer les différents mails aux clients lors de la création du compte ou la création des documents PDF. Je compte utiliser la librairie PHPMailer.

[PHPMailer](#)

## Frontend

### Bootstrap

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

- Angular
- React
- Vue
- Bootstrap

Angular offre beaucoup de fonctionnalité très pratique. 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 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, je me suis dit 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 pur JavaScript. Malgré le fait que Vue ait été implémenté récemment à Laravel, le point faible de Vue pour moi est sa complexité.

Le dernier de ma liste est Bootstrap. Ce framework plus que complet propose un nombre conséquent de composant facilement intégrable 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 framework les plus utilisés. Le point faible de Bootstrap est le fait qu'il soit plus lourd que ces concurrents et de ce fait, moins performant.

Angular et Vue étant plutôt complexe, React ayant une faible documentation, j'ai choisi de me tourner vers la simplicité et non la performance pour la réalisation de mon travail de diplôme. Malgré ses plus faibles performances, et afin de ne pas perdre trop de temps lors de mon développement, je compte utiliser le framework Bootstrap afin de développer ma PWA.

[Bootstrap](#)

## Dessin dans un canevas

Afin de pouvoir signer numériquement les contrats ainsi que 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-Sketchpad](#)

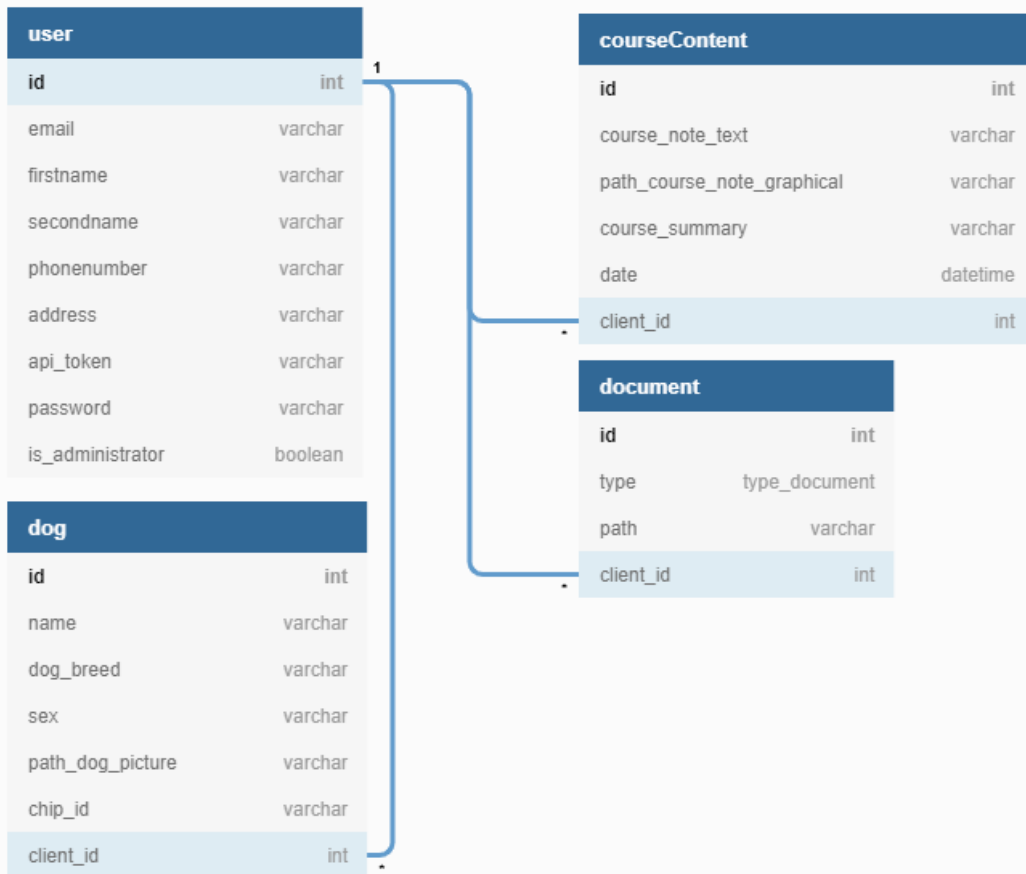
## Calendrier

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 utiliser FullCalendar, car il est très complet et que je l'ai déjà utilisé lors de mon TPI.

[FullCalendar](#)

## Modèle de données

---



user			
Nom	Type	Null	Définition
email	varchar	not null	Adresse email de l'utilisateur.
firstname	varchar	not null	Prénom de l'utilisateur.
secondname	varchar	not null	Nom de l'utilisateur.
phonenumber	varchar	not null	Numéro de téléphone de l'utilisateur.
address	varchar	not null	Adresse du domicile de l'utilisateur.
password	varchar	null	Mot de passe hashé.
api_token	varchar	not null	Token d'authentification de l'utilisateur.
is_administrator	boolean	not null	Booléen définissant le rôle de l'utilisateur.



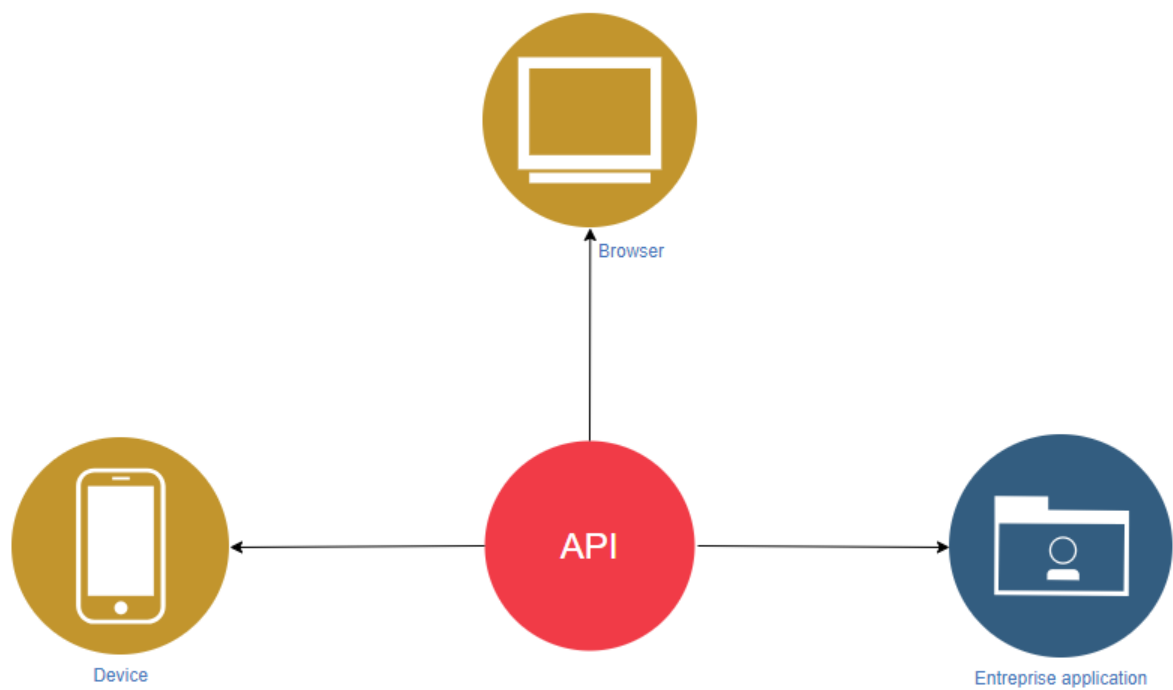
dog			
Nom	Type	Null	Définition
name	varchar	not null	Nom du chien.
dog_breed	varchar	not null	Race du chien.
sex	varchar	not null	Sexe du chien.
path_dog_picture	varchar	null	Chemin d'accès de la photo du chien.
chip_id	varchar	null	Code composé de 15 chiffres (3 pour le pays, 2 pour le type d'animal, 2 pour le fabricant, 8 pour le n° de l'animal).

document			
Nom	Type	Null	Définition
type	type_document	not null	Type de document (conditions d'inscription, poster, résumé du cours canin,ect...) .
path	varchar	not null	Chemin d'accès du document.

courseContent			
Nom	Type	Null	Définition
course_note_text	varchar	null	Note du cours sous format texte.
path_course_note_graphical	varchar	null	Chemin d'accès de note du cours sous format graphique.
course_summary	varchar	null	Note récapitulatif du cours.
date	datetime	not null	Date du cours.

## API

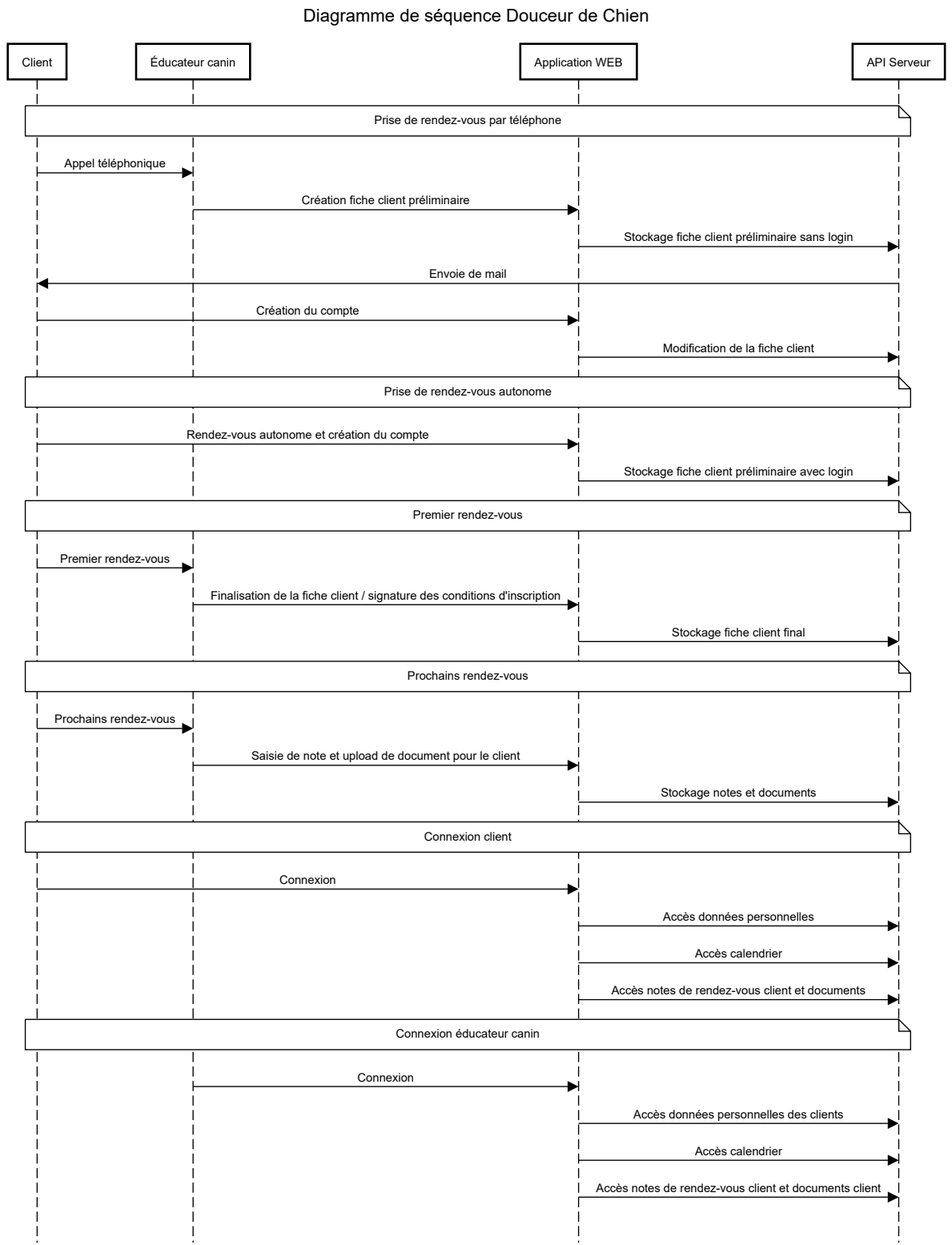
Afin de rendre les données de l'application utilisables sur différents futurs projets pour la société Douceur de Chien, je compte développer ma propre API. Celle-ci me permettra de rendre mes données réutilisables pour une application mobile native ou une application d'entreprise par exemple.



USER			
VERB	URL	Description	Permission
POST	/inscription	Crée un utilisateur.	Utilisateur non authentifié
POST	/login	Permet de fournir à un utilisateur son api_token ainsi que son rôle.	Utilisateur non authentifié
GET	/user	Récupère les données de l'utilisateur authentifié.	Utilisateur authentifié, administrateur
GET	/users	Récupère les données de tout les utilisateurs.	Administrateur
PATCH	/user/{id}	Modifie les données d'un utilisateur.	Administrateur
DELETE	/user/{id}	Supprime les données d'un utilisateur.	Administrateur
DOG			
VERB	URL	Description	Permission
POST	/dog/{idUser}	Crée un chien pour un utilisateur.	Administrateur
GET	/dog	Récupère les données des chiens de l'utilisateur authentifié.	Utilisateur authentifié, administrateur
GET	/dogs	Récupère les données de tout les chiens.	Administrateur
PATCH	/dog/{userId}	Modifie les données d'un chien d'un utilisateur.	Administrateur
DELETE	/dog/{userId}	Supprime les données d'un chien d'un utilisateur.	Administrateur
DOCUMENT			
VERB	URL	Description	Permission
POST	/document/{idUser}	Crée un document pour un utilisateur.	Administrateur
GET	/document	Récupère les documents de l'utilisateur authentifié	Utilisateur authentifié, administrateur

GET	/display-document/{id}	Affiche le document de l'utilisateur authentifié	Utilisateur authentifié, administrateur
DELETE	/document/{id}	Supprime un document	Administrateur

## Diagramme de séquence



## Maquettes de l'application

## Client

### Information personnelles



### Calendrier



Contenu séances



**Éducateur canin**

**Affichage de tout les clients**



Calendrier





**Création de fiche client préliminaire**



Affichage fiche client



Validation de fiche client



Contenue séance d'un client



Ajout de document



## Planning

---

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	30.4	3.5	4.5	5.5	6.5	7.5	10.5	11.5	12.5	19e jour (Ascensi on)	14.5
		1er jour	2e jour	3e jour	4e jour	5e jour	6e jour	7e jour	8e jour	9e jour	10e jour	11e jour	12e jour	13e jour	14e jour	15e jour	16e jour	17e jour	18e jour	20e jour	
API Rest Laravel																					
Création des Modèles																					
User	1	1																			
Dog	1	1																			
CourseContent	1	1																			
Document	1	1																			
Créations des Migrations																					
Users table	1	1																			
Dogs table	1	1																			
CoursesContent table	1	1																			
Documents table	1	1																			
Création des Factories																					
UserFactory	1	1																			
DogFactory	1	1																			
CourseContentFactory	1	1																			
DocumentFactory	1	1																			
Création des Contrôleurs																					
UserController	1	1																			
DogController	1	1																			
CourseContentController	1	1																			
DocumentController	1	1																			
CRUD																					
User	4	2	2																		
Dog	4	2	2																		
CourseContent	4				4																
Document	4				2	2															
Tests Unitaires																					
CRUD User	4					4		4													
CRUD Dog	4							4													
CRUD CourseContent	4							2	2												
CRUD Document	4							4													
Authentication	4								4												
Vues																					
Client																					
Onglet informations personnelles	4								2	2		4									
Onglet calendrier	8										4	4									
Onglet contenu séances	8										2	2	6								
Educateur chien																					
Onglet d'affichage de tous les clients	4													4							
Onglet calendrier	4													2	2						
Onglet de creation de fiche client préliminaire	4															4					
Onglet d'affichage de fiche client	4															4					
Onglet de validation de fiche client	4															2	2				
Onglet des contenus séances d'un client	4															4					
Onglet d'appui de document	4																4				
Fonctionnalités																					
Client																					
Inscription	8																2	4			
Affichage calendrier avec rendez-vous	8																				
Lecture PDF	8																				
Téléchargement PDF	8																				
Educateur chien																					
Affichage calendrier avec rendez-vous	10																				
Envoi de mail pour inscription au client	10																				
Prise de photo du chien	10																				
Dessin de notes personnelles	10																				
Lecture de PDF	8																				
Téléchargement PDF	8																				
Génération de document PDF	8																				
Documentation																					
Rapport	68	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
Workbook	30	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
300																				0	

[illegible]

## Fonctionnalités envisagées mais non retenues

## 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 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, premièrement, le temps de développement est trop conséquent par rapport à la plus-value direct que peut me rajouter cette fonctionnalité. Deuxièmement, ce code est

utilisable uniquement par des vétérinaires et autre autorité compétente. Je compte tout de même garder cette donnée afin de rendre mes données compatibles avec d'autres futurs projets pour la société Douceur de Chien.

## Matériel et logiciel

- Ordinateur Windows 10
- IDE (Visual Studio Code)
- Outil de versioning de code (GIT avec repository en ligne sur GitHub)
- Outil bureautique ([Typora](#))
- Éditeur de diagramme ([Visual Paradigm Online](#) - [sequencediagram](#))
- Éditeur de base de données ([dbdiagram.io](#))
- Éditeur de maquette ([Pencil](#))
- Éditeur de planning (Excel)
- Éditeur de MindMap ([Miro](#))

## Librairies utilisées

- Création de PDF grâce à du HTML ([DOMPDF](#))
- Dessiner dans un canevas ([responsive-sketchpad](#))
- Création de calendrier en JavaScript ([fullcalendar](#))

## Dates importantes

Date	Tâches	nbjours	nbjours effectifs
lundi, 15 mars 2021	version définitive de l'énoncé déposée sur Moodle par le candidat		
mardi, 16 mars 2021	liste du matériel, logiciel, infrastructure nécessaires au diplôme transmise à F. Garcia		
vendredi, 19 mars 2021	Fin 2ème semestre		
mercredi, 31 mars 2021	conseils des maîtres Tech ES		
01 au 09.04.2021	Vacances de Pâques		
lundi, 19 avril 2021	Début du travail de diplôme	1.0	1.0
vendredi, 30 avril 2021	Evaluation Intermédiaire 01	12.0	10.0
jeudi, 13 mai 2021	férié : Ascension	25.0	18.0
vendredi, 14 mai 2021	rendu du rapport intermédiaire + poster	26.0	19.0
vendredi, 14 mai 2021	rendu version intermédiaire du résumé et de l' abstract (pour conseils par l'enseignant d'anglais)	26.0	19.0
lundi, 17 mai 2021	Evaluation Intermédiaire 02	29.0	20.0
jeudi, 20 mai 2021	après-midi, soirée poster : amis, famille, CFC, experts 14h00 : visite des classes de 1ère année (cf. SG) 16h30 : amis, famille, experts, enseignants Tech ES... 18h00 : fin de la soirée poster	32.0	23.0
lundi, 24 mai 2021	férié : Pentecôte	36.0	24.0
lundi, 31 mai 2021	Evaluation Intermédiaire 03	43.0	29.0
vendredi, 11 juin 2021	rendu du travail avant 12:00	54.5	38.0
jeudi, 17 juin 2021	défenses à blanc + harmonisation des notes	60.0	42.0
lundi, 21 juin 2021	défenses de diplôme ( 8 candidats )	64.0	44.0
mardi, 22 juin 2021	défenses de diplôme + repas experts ( 6 candidats )	65.0	45.0

## Livrables

- Code-source
- Planning
- Documentation technique
- Manuel utilisateur (si applicable)
- Journal de travail ou LogBook
- Résumé / Abstract