



École Polytechnique de Montréal

Département de Génie Informatique et Génie
Logiciel

INF3710

Hiver 2021

TP3 : – Applications et Bases de données

Remis par :

Victor Kim 1954607
Dawut Esse 1956802

À : Rose Hirigoyen

Le 24 Avril 2021

Introduction

Les bases de données se retrouvent partout de nos jours, que ce soit dans les sites web, les logiciels et applications de notre quotidien. L'objectif de ce travail est de concrétiser nos connaissances acquises en BD relationnelle dans un contexte de développement Web. Ainsi, nous avons dû concevoir une application Web permettant à un employé de pouvoir rentrer et afficher des informations sur des examens, des traitements, des factures et des animaux incluant leurs propriétaires. Toutes les fonctionnalités et les détails d'utilisation ont dû être considéré dans le développement de VetoSansFrontieresDB, ce qui veut dire qu'une analyse d'étude de cas s'est fait avec grand détail. Cette dernière s'est faite à travers une modélisation conceptuelle et relationnelle qui nous ont permis de construire la base de données. Nos connaissances en BD relationnelles ont également été testé dans l'implantation d'une liste de requêtes. Du côté Web, la conception, cliente et serveur, s'est effectuée avec Node et Angular.

Normalisation et dépendances fonctionnelles :

DF

Clinique Vétérinaire : noClinique -> rue,ville,province,codePostal,noTelephone,noTelecopieur

Employé : noEmploye -> nom, prenom, adresse, noTelephone, dateDenaissance, sexe,NAS, fonction, salaire

Animal : noAnimal -> nom,type, espece,taille, poids, description, dateDeNaissance, dateInscription, etat, noProprietaire

Examen : noExamen -> date, heure,nomVeterinaire,noAnimal, description

Propriétaire : noPropriétaire -> nom, adresse, noTelephone

Type de traitement : noTypeTraitement -> description,cout

On n'a pas pu normaliser notre BD jusqu'à la forme 3NF par manque de temps. On peut affirmer qu'elle est au moins de UNF.

**Le modèle conceptuel UML :
Voir PDF nommé tp31 pour
évaluer le modèle conceptuel
UML....**

-Trigger :

Pour pouvoir utiliser le trigger, premièrement roulez le trigger2.sql. Par la suite, il faudra d'abord créer un animal, un examen, un traitement et un type de traitement dans data.sql comme vous pourriez le constater avec le test déjà fait. Ensuite, vous supprimez chaque élément et vous faites un Select * From His. Cela devrait afficher les animaux effacés qui sont reliés aux traitements et examens donnés. Le trigger n'a pas été implémenté pour le site, alors ce serait très conseillé de rouler le trigger à part et pas avec le site web vous pourriez quand même utiliser les autres .sql pour le site sauf trigger.

Modèle relationnel

**CliniqueVeterinaire(noClinique, rue, ville,
province, codePostal, noTelephone,
noTelecopieur)**

PK : noClinique

**Employee(noEmploye, nom, prenom,
adresse, noTelephone, dateDeNaissance,
sexe, NAS, fonction, salaire, noClinique)**

PK : noEmploye, noClinique

FK noClinique REFERENCES

Clinique(noClinique)

**Proprietaire(noProprietaire, nom, adresse,
noTelephone, noClinique)**

PK : noProprietaire

FK noClinique REFERENCES

CliniqueVeterinaire(noClinique)

**Animal(noAnimal, nom, type, espece, taille,
poid, description, dateDeNaissance,
dateInscription, etat, noProprietaire)**

PK : noAnimal

FK noProprietaire REFERENCES

Proprietaire(noProprietaire)

**Examen(noExamen, date, heure,
nomVeterinaire, noAnimal, description)**

PK : noExamen

FK noAnimal REFERENCES

Animal(noAnimal)

**TypeTraitement(noTypeTraitement,
description, cout)**

PK : noTypeTraitement

**Traitement(noExamen, dateExamen ,
noAnimal , noTypeTraitement, quantite,
dateDebut, dateFin)**

PK : noExamen

FK noExamen REFERENCES

Examen(noExamen)

FK noProprietaire REFERENCES

Proprietaire(noProprietaire)

FK noAnimal REFERENCES

Animal(noAnimal)

FK noEmploye REFERENCES

Employe(noEmploye)

FK noTypeTraitement REFERENCES

TypeTraitement(noTypeTraitement)

**Facture(noProprietaire, noAnimal,
noEmploye, date, total, paye,
typeDePaiement)**

PK noProprietaire,noAnimal,noEmploye

FK noProprietaire REFERENCES

Proprietaire(noProprietaire)

FK noAnimal REFERENCES

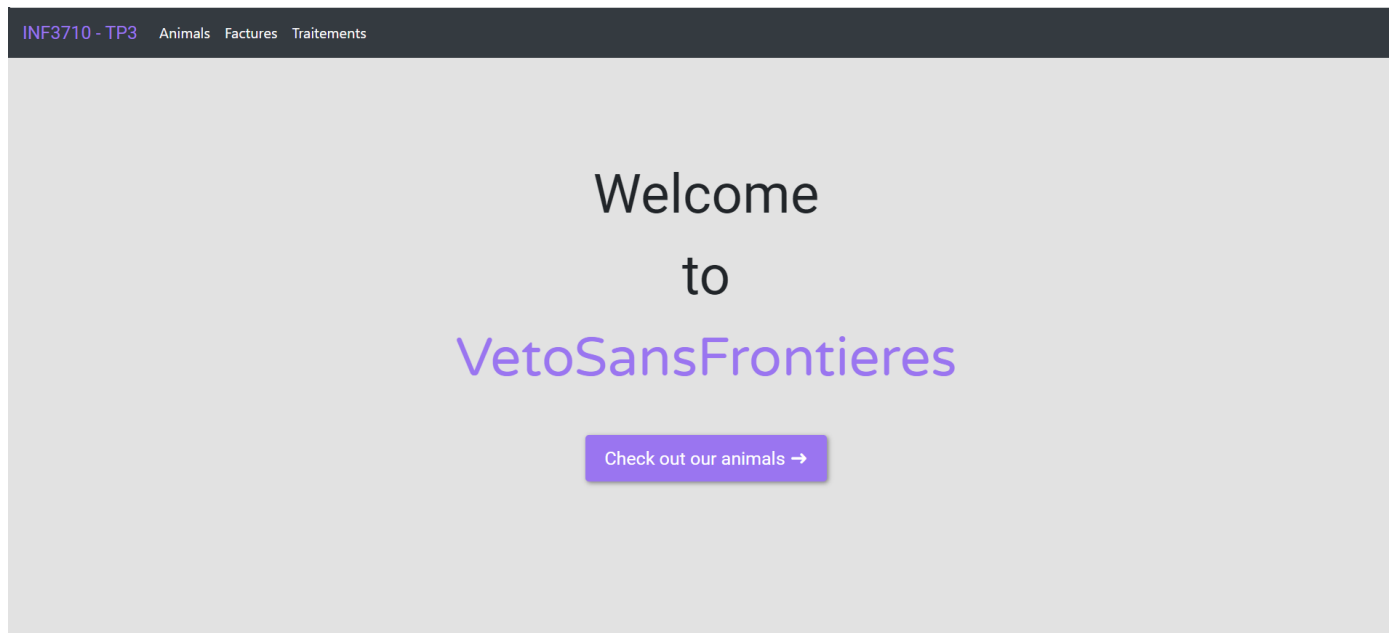
Animal(noAnimal)

FK noEmploye REFERENCES

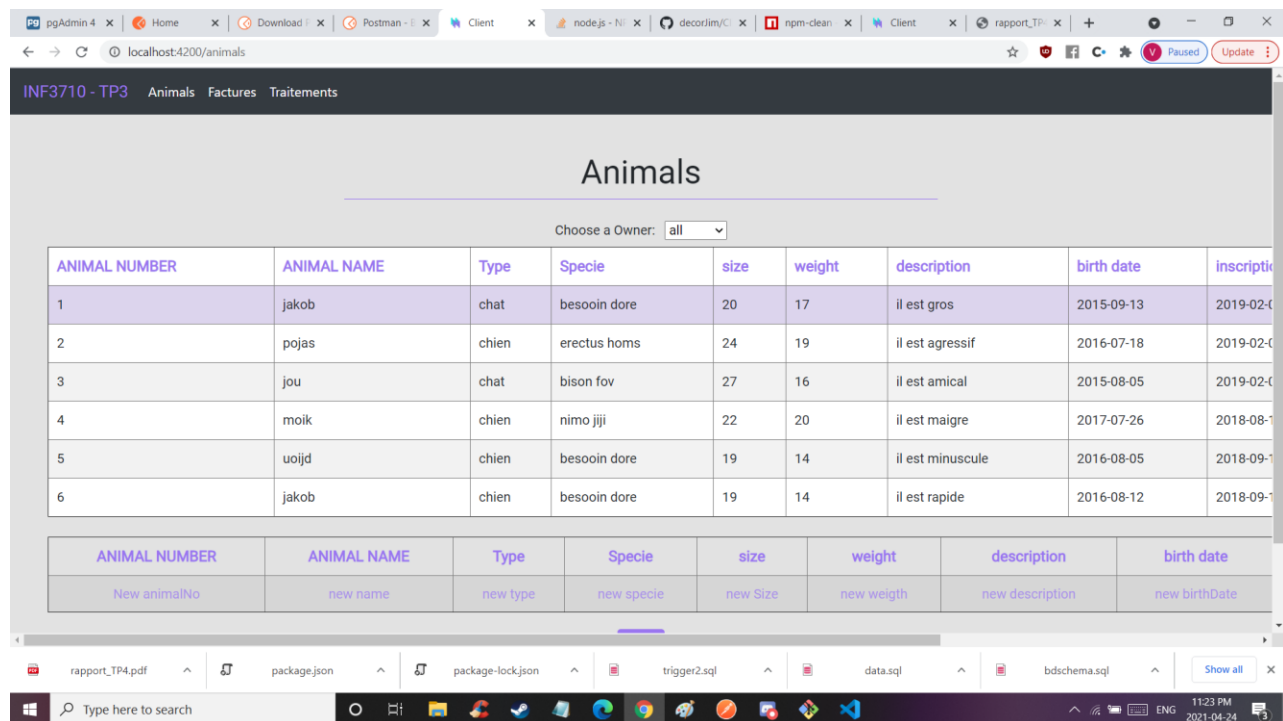
Employe(noEmploye)

**Présentation de l'application
développée :**

Ceci est l'accueil du site web.















Lorsque tu cliques Check out our animals :




On peut choisir un propriétaire et les animaux qui lui sont associés vont apparaître sur le site web

Animals

Choose a Owner:

Type	Specie	size	weight	description	birth date	inscription date	State	Owner	Action
chat	besoin dore	20	17	il est gros	2015-09-13	2019-02-01	vivant	1	 
chien	erectus homs	24	19	il est agressif	2016-07-18	2019-02-02	vivant	1	 
chat	bison fov	27	16	il est amical	2015-08-05	2019-02-01	vivant	2	 
chien	nimo jiji	22	20	il est maigre	2017-07-26	2018-08-19	mort	3	 
chien	besoin dore	19	14	il est minuscule	2016-08-05	2018-09-10	vivant	4	 
chien	besoin dore	19	14	il est rapide	2016-08-12	2018-09-10	vivant	4	 




On peut effacer l'animal ou modifier des informations lui concernant si on regarde la colonne Action.

On peut aussi ajouter un animal en rentrant les informations de celui comme vous pouvez le voir près de l'Icon +.

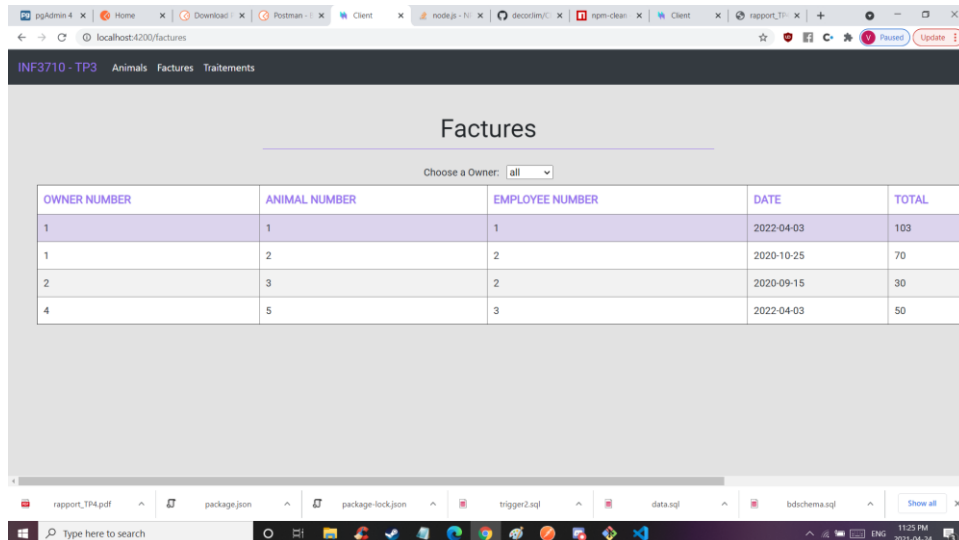
Animals

Choose a Owner:

ANIMAL NUMBER	ANIMAL NAME	Type	Specie	size	weight	description	birth date	inscription date
1	jakob	chat	besoin dore	20	17	il est gros	2015-09-13	2019-02-01
2	pojas	chien	erectus homs	24	19	il est agressif	2016-07-18	2019-02-02
3	jou	chat	bison fov	27	16	il est amical	2015-08-05	2019-02-01
4	moik	chien	nimo jiji	22	20	il est maigre	2017-07-26	2018-08-19
5	uoijd	chien	besoin dore	19	14	il est minuscule	2016-08-05	2018-09-10
6	jakob	chien	besoin dore	19	14	il est rapide	2016-08-12	2018-09-10

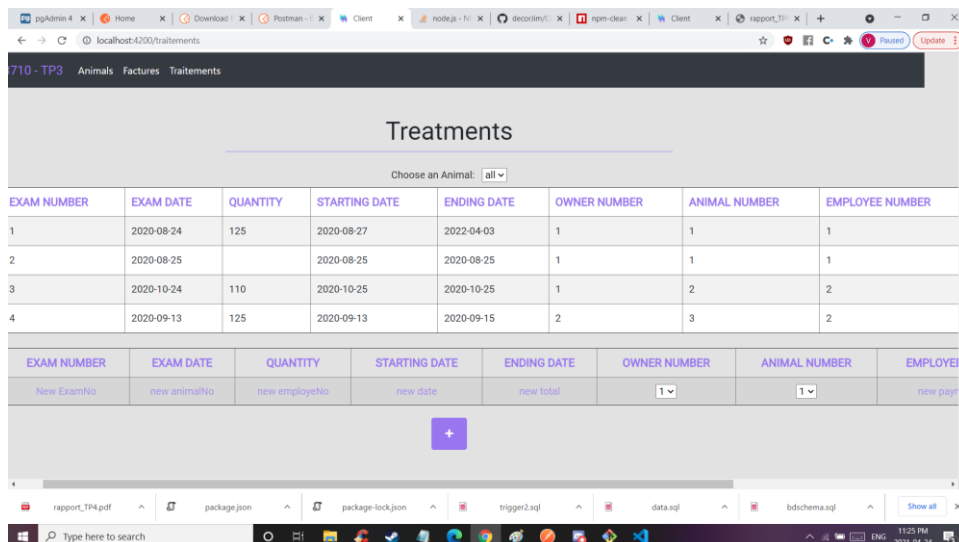


Ceci est la page Facture qui liste toutes les factures associées à un propriétaire.



OWNER NUMBER	ANIMAL NUMBER	EMPLOYEE NUMBER	DATE	TOTAL
1	1	1	2022-04-03	103
1	2	2	2020-10-25	70
2	3	2	2020-09-15	30
4	5	3	2022-04-03	50

Ceci est la page traitement associé à un animal.



EXAM NUMBER	EXAM DATE	QUANTITY	STARTING DATE	ENDING DATE	OWNER NUMBER	ANIMAL NUMBER	EMPLOYEE NUMBER
1	2020-08-24	125	2020-08-27	2022-04-03	1	1	1
2	2020-08-25		2020-08-25	2020-08-25	1	1	1
3	2020-10-24	110	2020-10-25	2020-10-25	1	2	2
4	2020-09-13	125	2020-09-13	2020-09-15	2	3	2

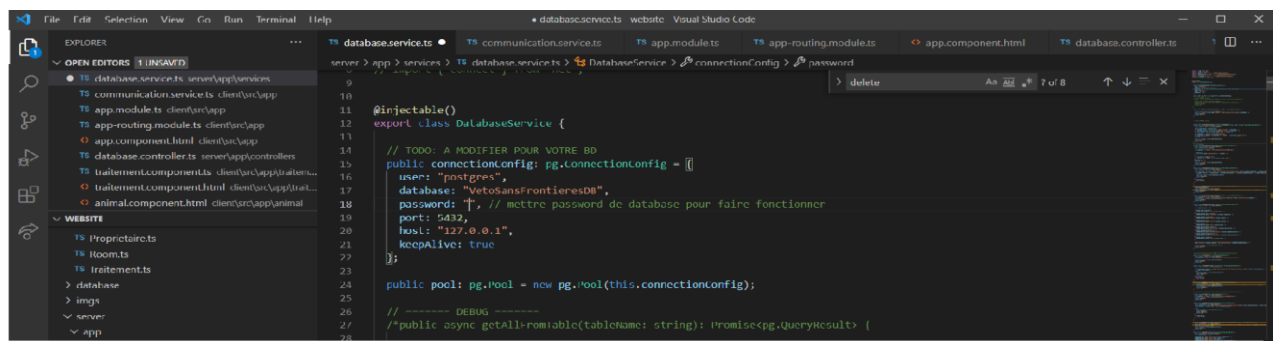
EXAM NUMBER	EXAM DATE	QUANTITY	STARTING DATE	ENDING DATE	OWNER NUMBER	ANIMAL NUMBER	EMPLOYEE NUMBER
New ExamNo	new animalNo	new employeeNo	new date	new total	1	1	new page

Note importante : Pour ajouter un traitement, on doit d'abord créer un examen avant et ensuite un traitement. Vous devriez ajouter un

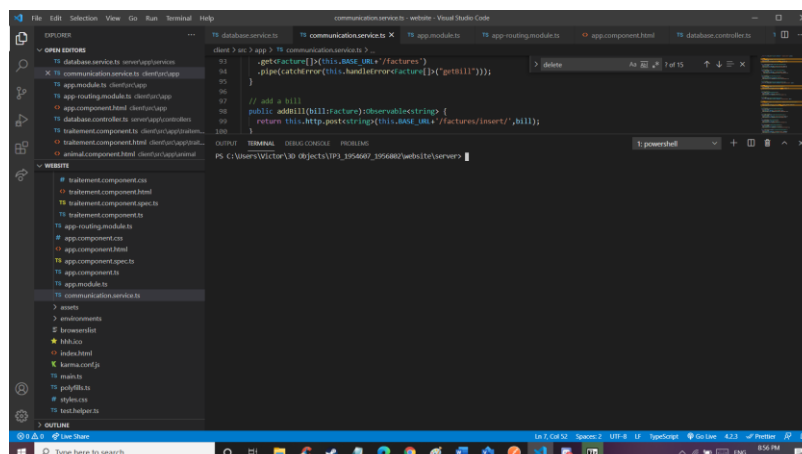
examen dans data.sql et créez ensuite le traitement sur le site.

Guide d'installation et de configuration :

- 1- Premièrement, ouvrez le terminal, allez dans le dossier serveur, faite un npm ci ou npm install .
- 2- Allez sur le fichier database.service.ts présent dans le répertoire /server/app/services.
- 3-Inscrivez un mode de passe dans la base de données qui est à la ligne 18 pour que vous puissiez avoir accès à la BD.



- 4- Ouvrez un nouveau terminal et allez à ce path pour accéder au serveur :
TP3_1954607_1956802\website\server et faites un npm start ici



5- Si tout s'est bien passé, il devrait vous écrire "listening on port 3000". Sinon, vous indiquer une erreur de username/mot de passe incorrect ou de BD inexistante

6- Allez ensuite sur le répertoire du client et exécutez encore, de même que le serveur, un npm ci ou npm install

7- . Quand les modules ont fini de se télécharger, exécutez la commande npm start qui devrait compiler les fichiers et ouvrir une fenêtre avec l'application.

8. Vous pouvez commencer à utiliser l'application.