

# Equida

---

## Document de spécification

# Equipe JAVABIEN

Version : 1.0

Date de dernière mise à jour : 12 novembre 2020

### Auteurs :

**NOËL Clara**  
**CRESTE Jérôme**  
**CHERUEL Quentin**  
**CHAUVIN Tom**

## **Table des matières**

1 – Historique des modifications du document .....	3
2 – Présentation.....	3
2.1 – Application .....	3
2.2 – Documents de référence.....	3
2.3 – Spécifications générales .....	3
2.4 – Contraintes d'évolution .....	3
3 – Spécifications d'interfaces .....	3
3.1 – Structure des pages.....	3
3.2 – Liste des pages.....	4
3.3 – Interfaces avec des fichiers .....	4
4 – Spécifications techniques .....	4
4.1 – Environnement de production .....	4
4.2 – Environnement logiciel.....	4
4.3 – Mise en œuvre.....	4
4.4 – Exigences de programmation .....	5
5 – Modélisation.....	5
5.1 – Modèle relationnel et base de données .....	5
5.2 – Diagramme de classe .....	6
6 – Architecture applicative.....	6

## 1 – Historique des modifications du document

Date	Auteur	Version	Sujet de la modification
01/10/2020	JAVABIEN	1.0	Création
08/10/2020	JAVABIEN	1.1	Présentation du projet
10/11/2020	JAVABIEN	1.2	Spécifications

## 2 – Présentation

### 2.1 – Application

L'application de la société Equida, permet de gérer plusieurs domaines, tel que le planning des ventes aux enchères, la gestion des ventes, en présentant les lieux de vente, ainsi que les informations relatives aux chevaux, toutes ses informations seront disponibles aux utilisateurs.

### 2.2 – Documents de référence

Document	Fonction
Cahier des charges	Description des besoins utilisateurs
Spécifications logicielles	Décrit fonctionnellement et techniquement l'application

### 2.3 – Spécifications générales

Dans la dernière version rendu, supposée finale, l'application permet la consultation des clients, des ventes, de ses catégories de ventes, les chevaux, les lieux de vente et les lots. Nous pouvons aussi prendre connaissance des clients intéressés selon une vente, les courriels et les chevaux mis en vente lors de l'évènement.

### 2.4 – Contraintes d'évolution

L'application ayant évoluée par rapport à la version de départ, il est tout à fait possible de la faire évoluer de nouveau, ce qui est donc anticiper. Il faudra tout de même respecter les contraintes d'intégrité présente entre les relations des tables dans la base de données. Une évolution sur celle-ci est donc très restreinte.

## 3 – Spécifications d'interfaces

### 3.1 – Structure des pages

Toutes les pages de l'application ont en commun le menu de navigation, elles ont le même style (police d'écriture, dimension, code couleur...).

### 3.2 – Liste des pages

Interface	Fichier	Accès	Fonction
Accueil	Index.jsp	URL	Présentation générale de l'entreprise
Vente	ListerLesVentes.jsp	URL	Affichage de toutes les ventes et informations complémentaires
Clients pour une catégorie de vente	ListerLesClients	Récupération id de la vente	Affichage de tous les clients intéressés par une vente
Courriels pour une catégorie de vente	ListerLesCourriels	Récupération id de la vente	Affichage de tous les courriels en relation avec la vente
Chevaux pour une vente	ListerLesChevaux.jsp	Récupération id de la vente	Affichage de tous les chevaux en vente
Catégories de ventes	ListerLesCategories.jsp	URL	Affichage des catégories de vente
Clients	ListerLesClients.jsp	URL	Affichage de tous les clients
Chevaux	ListerTousLesChevaux.jsp	URL	Affichage de tous les chevaux
Lot	ListerLesLots.jsp	URL	Affichage de tous les lots
Lieu	ListerLesLieux.jsp	URL	Affichage de tous les lieux de vente

### 3.3 – Interfaces avec des fichiers

Utilisation d'un fichier au format CSV pour l'import des noms de pays, utilisation d'image pour l'identification des chevaux.

## 4 – Spécifications techniques

### 4.1 – Environnement de production

Pour le bon fonctionnement de l'application, il est indispensable de mettre en place un serveur pour ainsi permettre l'accès à distance ce celle-ci.

### 4.2 – Environnement logiciel

Après la mise en place du serveur, pour le bon fonctionnement de l'application, nous devons disposer d'une base de données (mySQL), d'un serveur web et d'application). L'application étant principalement programmer avec le langage JAVA, il faudra en plus de cela, permettre l'accès à un JDK 7u75 ainsi qu'une création de variable d'environnement.

### 4.3 – Mise en œuvre

L'application est livrée à l'adresse suivante : 172.20.0.241:8080/manager/html

Avec les identifiant suivant : tomcat/tomcat

## 4.4 – Exigences de programmation

L'application comporte 3 langages : **JAVA, HTML, CSS**

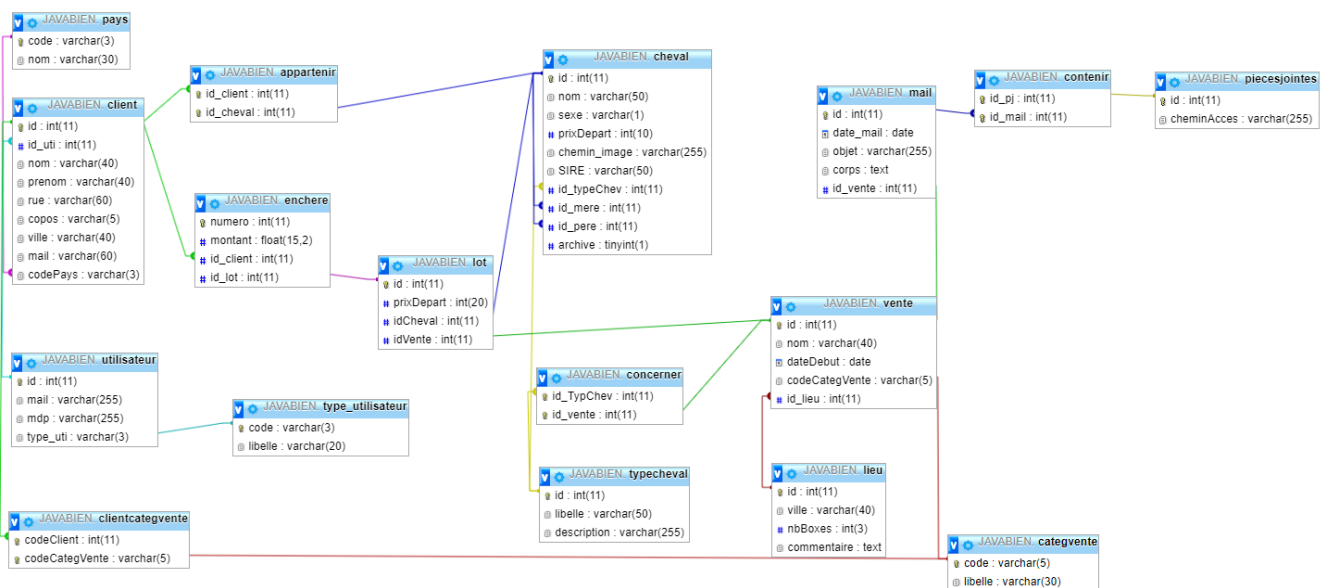
Utilisateur de l'IDE **netBeans 8.2**

La norme **Camel Case** est utilisée

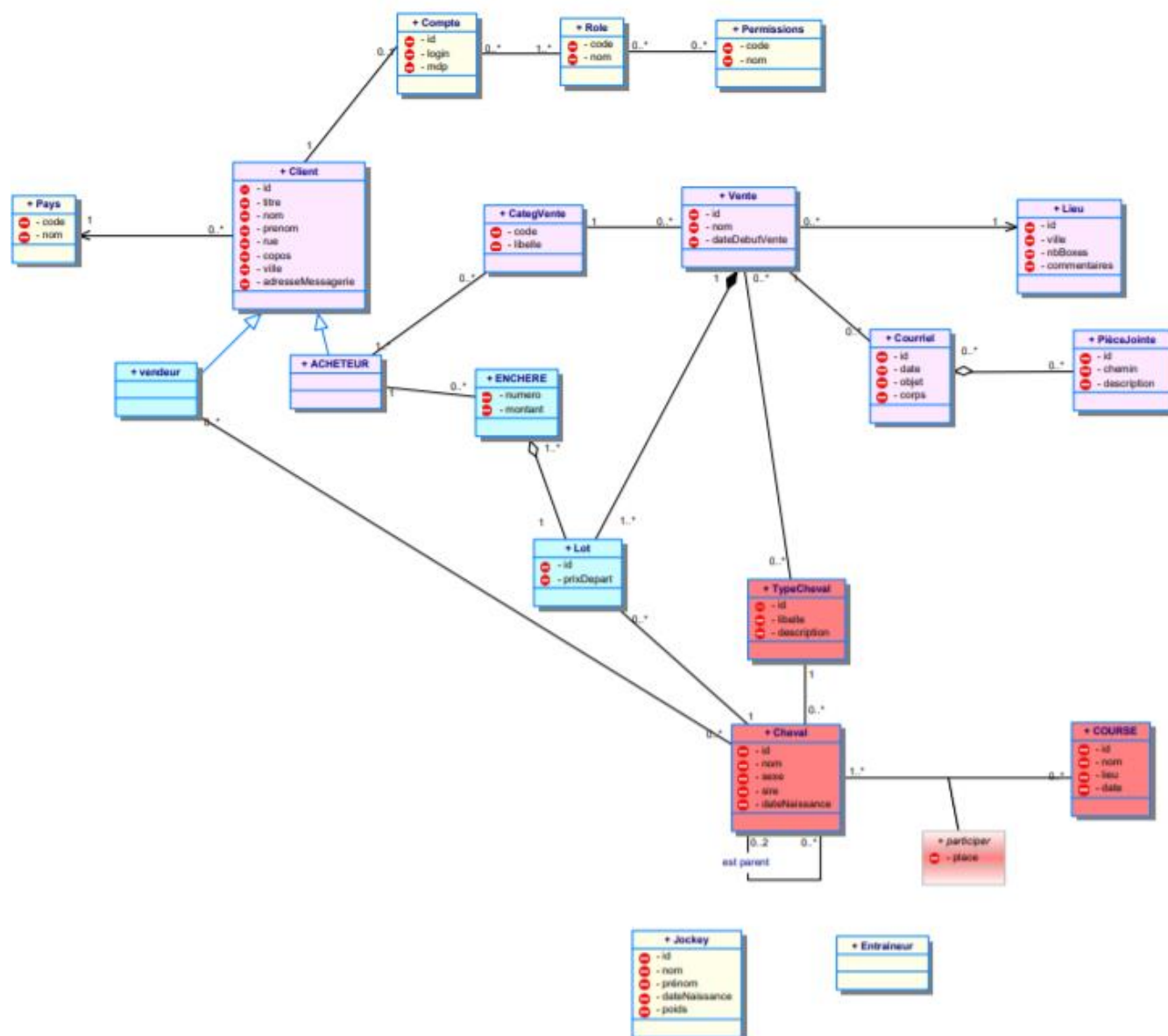
La gestion des versions est gérée grâce à **Git**

## 5 – Modélisation

### 5.1 – Modèle relationnel et base de données



## 5.2 – Diagramme de classe



## 6 – Architecture applicative

Le projet est développé selon la méthode Modèle Vue Contrôleur (MVC), dans la partie modèle, nous avons toutes les classes nécessaires au bon fonctionnement de l'application, dans la partie contrôleur, nous avons les servlets avec les fonctions nécessaires pour toutes les actions mise à disposition pour l'utilisateur et la partie vue afin d'afficher correctement les résultats des actions.