

UTBM - Département Informatique

**Projet – BD40**

**Sujet n.3 : Cave à vin**

*Automne 2018 – 10/10/2018*

**Groupe 3**

Anaïs JARNO

Haoyun LIAO

Weitao YAN

**TABLE DES MATIERES**

[1 Historique du document 1](#_Toc531787628)

[2 Description du sujet de projet 2](#_Toc531787629)

[2.1 Contexte du projet 2](#_Toc531787630)

[2.2 Domaine étudié 2](#_Toc531787631)

[2.3 Fonctionnement du système 3](#_Toc531787632)

[2.4 Acteurs et flux 4](#_Toc531787633)

[2.5 Périmètre du système 4](#_Toc531787634)

[2.6 Extensions possibles 4](#_Toc531787635)

[3 Dossier de conception du projet 5](#_Toc531787636)

[3.1 Modélisation conceptuelle 5](#_Toc531787637)

[3.1.1 Dictionnaire de données 5](#_Toc531787638)

[3.1.2 Modèle entité association 7](#_Toc531787639)

[3.1.3 Modèles de communication et de traitements 11](#_Toc531787640)

[4 La modélisation logique 12](#_Toc531787641)

[5 La modélisation de l’IHM 15](#_Toc531787642)

[6 Sources 21](#_Toc531787643)

# Historique du document

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Auteur** | **Modifications du document** |
| Etape 1 | 09/10/2018 | Anaïs J. | Création du rapport, page de garde, table des matières et mise en page |
|  | 13/10/2018 | Anaïs J. | Création d’un schéma « résumé » pour la partie explicative du projet |
|  | 21/10/2018 | Weitao | Rédaction du contenu 2.3 et 2.4 |
|  | 21/10/2018 | Haoyun | Rédaction du contenu 2.1 et 2.2 |
|  | 21/10/2018 | Anaïs J. | Création du MCD (entité-relation) |
|  | 21/10/2018 | Weitao | Création du modèle de communication (Diagramme de flux) |
|  | 21/10/2018 | Haoyun et Anaïs | Modifications et correction des parties 2.1, 2.2 |
|  | 22/10/2018 | Weitao | Modification du contenu 2.3 et 2.4 et rédaction du 2.5 |
|  | 22/10/2018 | Anaïs J. | Ajout de la partie 2.6 et modification du MCD |
|  | 22/10/2018 | Weitao | Création du MCT et modification du MCC |
|  | 22/10/2018 | Haoyun | Création du dictionnaire |
|  | 26/10/2018 | Anaïs J. | Correction du MCD |
|  | 05/11/2018 | Weitao | Modification du MCT et MCC |
|  | 22/10/2018 | Haoyun | Création du dictionnaire |
|  | 07/11/2018 | Anaïs J. | Dernières modifications du MCD, du dictionnaire et des parties |
|  | 23/11/2018 | Anaïs J. | Correction MCD, orientation des pages et tri du dictionnaire |
|  | 04/11/2018 | Weitao | Ajout de la partie Normalisation |
|  | 04/11/2018 | Anaïs J. | Ajout et rédaction de la partie IHM |

# Description du sujet de projet

## Contexte du projet

Nous sommes des étudiants en première année d’informatique à l’UTBM. Dans le cadre de l’UV de BD40 concernant la conception de systèmes d’informations, nous avons à réaliser un projet pour mettre en application les connaissances acquises durant les cours et les séances de travaux pratiques. Les membres de notre groupe étaient très intéressés et curieux de découvrir le sujet du vin. Le groupe étant composé de deux chinois, nous pensions que ce serait une bonne occasion pour approfondir un sujet très lié à la culture française.

C’est ainsi que nous avons donc choisi le sujet de la cave à vin.

## Domaine étudié

Nous avons choisi de restreindre le domaine à une utilisation personnelle.

Le principe est de créer une application de base de données permettant la gestion de cave à vin personnelle. Elle devra ainsi permettre à son utilisateur de visualiser le contenu de sa cave, les informations concernant ses vins et bouteilles, et d’ajouter ou de retirer des bouteilles de sa cave.

Nous étudions donc les diverses fonctions nécessaires à l’application, telles que la conservation du vin, l’enregistrement de la date d'achat, de la meilleure date de consommation (l’apogée) etc.

Nous avons alors réalisé un schéma servant à résumer les caractéristiques principales de notre gestion de cave :

Une image contenant texte, carte

Description générée avec un niveau de confiance très élevé

## Fonctionnement du système

Ce système d’information s’adresse à la gestion d’une cave à vin personnelle. On distingue le vin et la bouteille. Les termes ‘*fournisseur’* et ‘*producteur’* indiqueront le même acteur.

Des fournisseurs vont produire des vins de différentes régions et ayant d’autres caractéristiques comme la couleur, le cépage, l’appellation, etc.

Le client peut acheter plusieurs bouteilles à la fois, ou une seule. La cave à vin sert à stocker des bouteilles achetées par le consommateur. Les bouteilles sont stockées dans la cave jusqu’au jour où le consommateur veut la déguster.

Dès que la bouteille est sortie de la cave pour être dégustée, celle-ci est retirée de la liste des bouteilles, ou sa quantité est réduite. Une date d’entrée est également notée dès que la bouteille entre en stock.

On a également une fiche de dégustation qui permet d’enregistrer toutes les informations concernant la dégustation pour que le moment venu le consommateur puisse choisir le vin le plus adapté à son désir, à son repas...

## Acteurs et flux

|  |  |  |  |
| --- | --- | --- | --- |
| Les acteurs externes | Les acteurs internes | Le flux primaire | Les flux secondaires |
| Producteur de vin | * Le consommateur * La cave | Les entrées-sorties de bouteilles dans la cave | * Le stockage du vin (information enregistrée) * La dégustation du vin * L’achat du vin |

## Périmètre du système

Notre système est très simple. C’est une cave à vin personnelle. C’est-à-dire la cave sert à stocker des bouteilles pour un foyer ou pour une personne.

De temps en temps, l’utilisateur achètera une bouteille ou plusieurs bouteilles, qu’il fera alors entrer dans sa cave.

Puis, il invitera des amis, des parents ou des collègues chez lui et ils pourront alors déguster l’un des vins contenus dans la cave. Par la suite, il pourra alors remplir une fiche de dégustation. Cela permet de noter les avis après dégustation pour que l’utilisateur puisse choisir la prochaine fois son vin préféré, ou le plus adapté à la situation.

## Extensions possibles

Les extensions possibles du projet, c’est-à-dire les options que l’on ferait dans un contexte plus professionnel et non scolaire, pourrait être du type :

* Système de facture lié à l’enregistrement d’une arrivée de stock
* Revente de bouteilles
* Commande directe depuis l’application pour un vin spécifique vers le producteur concerné et gestion de la livraison

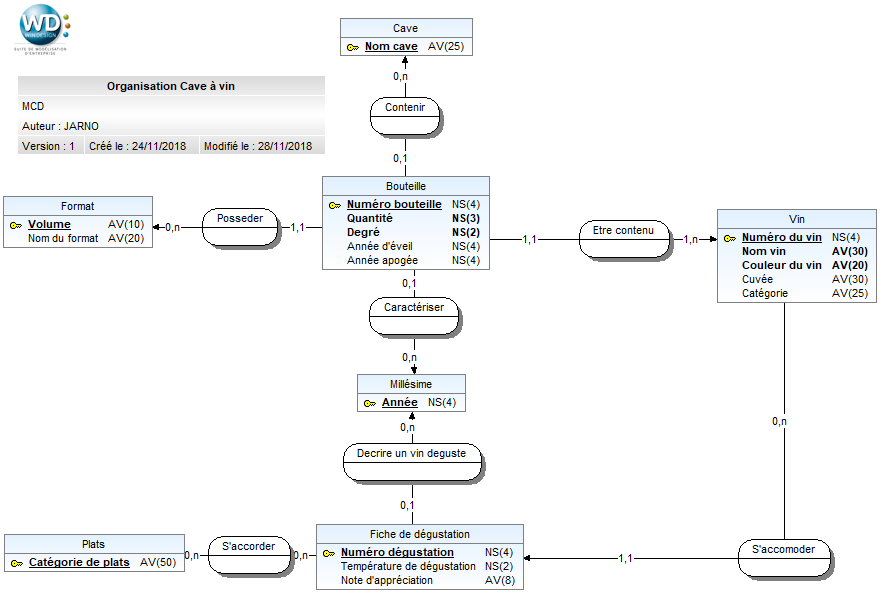
# Dossier de conception du projet

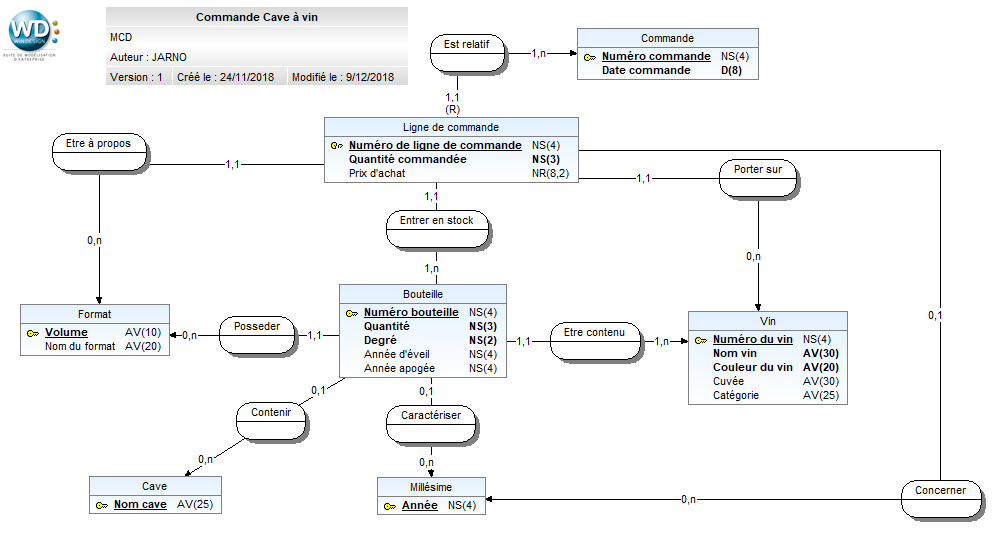
## Modélisation conceptuelle

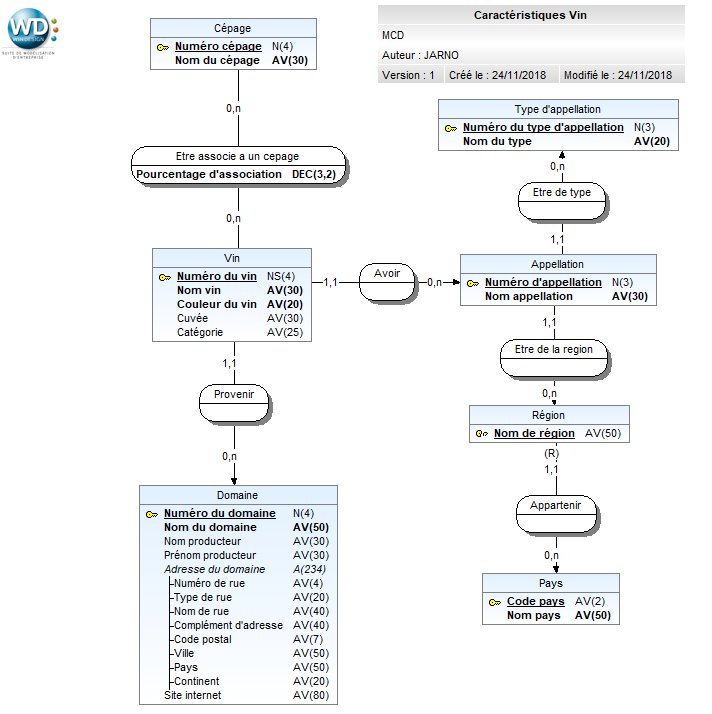
### Dictionnaire de données

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nom Conceptuel | Nom Logique ou Alias | Type  (E, Ca, Co) | Nature + Longueur | Type  Win’Design | Identifiant |
| Adresse du domaine | ADR\_DOMAINE | Co | AN234 | A(234) |  |
| Numéro de rue | NUM\_RUE\_DOMAINE | E | N4 | AV(4) |  |
| Type de rue | TYPE\_RUE\_DOMAINE | E | A20 | AV(20) |  |
| Nom de rue | NOM\_RUE\_DOMAINE | E | A40 | AV(40) |  |
| Complément d’adresse | COMPL\_ADR\_DOMAINE | E | A40 | AV(40) |  |
| Ville | VILLE\_DOMAINE | E | A50 | AV(50) |  |
| Code postal | CP\_DOMAINE | E | A7 | AV(7) |  |
| Pays | PAYS\_DOMAINE | E | A50 | AV(50) |  |
| Continent | CONTINENT\_DOMAINE | E | A20 | AV(20) |  |
| Année | ANNEE\_MILLESIME | E | N4 | NS(4) | OUI |
| Année apogée | ANNEE\_APOGEE | E | N4 | NS(4) |  |
| Année d’éveil | ANNEE\_EVEIL | E | N4 | NS(4) |  |
| Catégorie | CATEG\_VIN | E | A25 | AV(25) |  |
| Catégorie de plats | CATEGORIE\_PLATS | E | A50 | AV(50) |  |
| Code pays | CODE\_PAYS | E | A2 | AV(2) | OUI |
| Couleur du vin | COULEUR\_VIN | E | A20 | AV(20) |  |
| Cuvée | CUVEE\_VIN | E | A30 | AV(30) |  |
| Date commande | DATE\_COMMANDE | E | AN8 | D(8) |  |
| Degré | DEGRE\_BOUTEILLE | E | N2 | NS(2) |  |
| Nom appellation | NOM\_APPELLATION | E | A30 | AV(30) |  |
| Nom cave | NOM\_CAVE | E | A25 | AV(25) | OUI |
| Nom de région | NOM\_REGION | E | A50 | AV(50) | OUI |
| Nom du cépage | NOM\_CEPAGE | E | A30 | AV(30) |  |
| Nom du domaine | NOM\_DOMAINE | E | A50 | AV(50) |  |
| Nom du format | NOM\_FORMAT | E | A20 | AV(20) |  |
| Nom du type | NOM\_TYPEAPP | E | A20 | AV(20) |  |
| Nom pays | NOM\_PAYS | E | A50 | AV(50) |  |
| Nom producteur | NOM\_PROD | E | A30 | AV(30) |  |
| Nom vin | NOM\_VIN | E | A30 | AV(30) |  |
| Note d’appréciation | NOTE\_DEGUST | E | A8 | AV(8) |  |
| Numéro bouteille | NUM\_BOUTEILLE | E | N4 | NS(4) | OUI |
| Numéro cépage | NUM\_CEPAGE | E | N4 | N(4) | OUI |
| Numéro commande | NUM\_COMMANDE | E | N4 | NS(4) | OUI |
| Numéro d’appellation | NUM\_APPELLATION | E | N3 | N(3) | OUI |
| Numéro dégustation | NUM\_DEGUST | E | N4 | NS(4) | OUI |
| Numéro de ligne de commande | NUMERO\_LIGNE\_DE\_COMMANDE | E | N4 | NS(4) | OUI |
| Numéro du domaine | NUM\_DOMAINE | E | N4 | N(4) | OUI |
| Numéro du type d’appellation | NUM\_TYPEAPP | E | N3 | N(3) | OUI |
| Numéro du vin | NUM\_VIN | E | N4 | NS(4) | OUI |
| Pourcentage d’association | POURCENTAGE\_CEPAGE | E | AN(3,2) | DEC(3,2) |  |
| Prénom producteur | PRENOM\_PROD | E | A30 | AV(30) |  |
| Prix d’achat | PRIX\_ACHAT | E | AN(8,2) | DEC(8,2) |  |
| Quantité | QUANTITE\_BOUTEILLE | E | N3 | NS(3) |  |
| Quantité commandée | QUANTITE\_COMMANDE | E | N3 | NS(3) |  |
| Site internet | SITE\_PROD | E | A80 | AV(80) |  |
| Température de dégustation | TEMP\_DEGUST | E | N2 | NS(2) |  |
| Volume | VOLUME\_FORMAT | E | A10 | AV(10) | OUI |

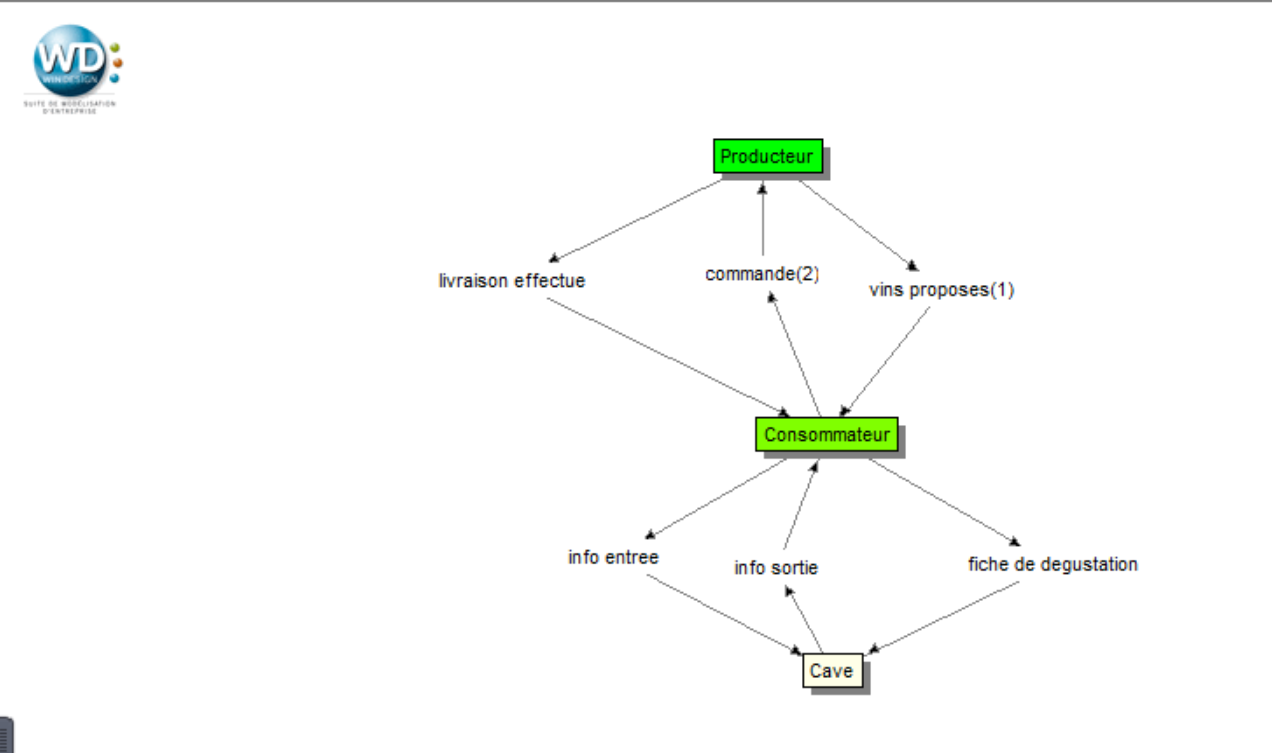
### Modèle entité association

**Décomposition du MCD global en sous-modèles :**





### Modèles de communication et de traitements



# La modélisation logique

**MLR-Sous modèle-Organisation**

**1FN : Oui**

*Justification* : Tous les attributs sont fonctionnellement dépendants de la clé.

**2FN : Oui**

*Justification* : Toutes les relations sauf ‘*S’ACCORDER*’ n’ont qu’une seule clé. Donc naturellement tout attribut non clé ne dépend pas d’une partie de cette clé. C’est-à-dire qu’ils sont en dépendance fonctionnelle élémentaire par rapport à la clé. Quant à ‘*S’ACCORDER*’, elle est en 2FN étant donné qu’elle n’a pas d’attribut non clé.

**3FN : Oui**

*Justification* :

*BOUTEILLE :* Tous les attributs non clé sont indépendants des autres.

*VIN :* Tous les attributs non clé sont indépendants des autres. Pour *NUM\_DOMAINE* et *NUM\_APPELLATION*, vu qu’ils se réfèrent à des entités différentes n’ayant pas d’associations entre elles. Ils sont ainsi indépendants.

*FICHE\_DEGUST* : Tous les attributs non clé sont indépendants des autres. Pour *NUM\_VIN* et *ANNEE\_MILLESIME*, vu que **MILLESIME** caractérise la bouteille, ils sont ainsi indépendants.

**Conclusion** : Ce sous-modèle est en 3FN, il est ainsi normalisé.

**MLR-Sous modèle-Commande**

**1FN : Oui**

*Justification* : Tous les attributs sont fonctionnellement dépendants de la clé.

**2FN : Oui**

*Justification* : Toutes les relations sauf ‘*LIGNE\_DE\_COMMANDE*’ et ‘*ENTRER EN STOCK*’ n’ont qu’une seule clé. Donc naturellement tout attribut non clé ne dépend pas d’une partie de cette clé. C’est-à-dire qu’ils sont en dépendance fonctionnelle élémentaire par rapport à la clé. Quant à ‘*ENTRER EN STOCK*’, elle est en 2FN étant donné qu’elle n’a pas d’attribut non clé. Quant à ‘*LIGNE\_DE\_COMMANDE*’, vu que l’identifiant de l’entité *LIGNE\_DE\_COMMANDE* est relatif, autrement dit, tout attribut non clé de *LIGNE\_DE\_COMMANDE* sont en dépendance fonctionnelle élémentaire par rapport à ces deux clés. Elle est ainsi en 2FN.

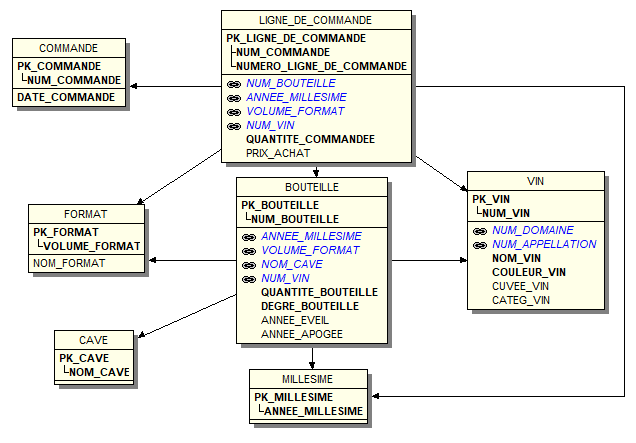
**3FN : Oui**

*Justification* :

*BOUTEILLE***:** Tous les attributs non clé sont indépendants des autres.

*VIN* : Tous les attributs non clé sont indépendants des autres. Pour **NUM\_DOMAINE** et **NUM\_APPELLATION**, vu qu’ils se réfèrent à des entités différentes n’ayant pas d’associations entre elles. Ils sont ainsi indépendants.

*LIGNE\_DE\_COMMANDE :*Tous les attributs non clé sont indépendants des autres.

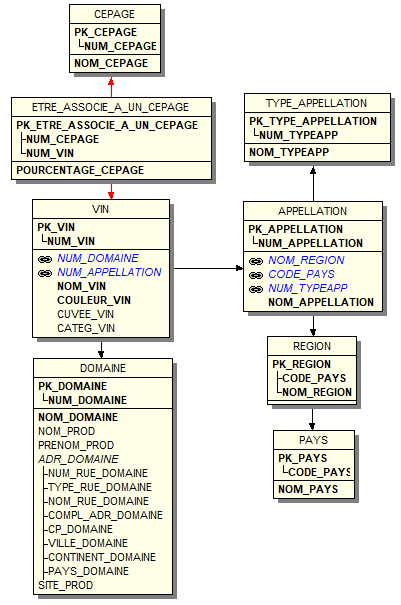
**Conclusion** : Ce sous-modèle est en 3FN, il est ainsi normalisé.

**MLR-Sous modèle-Commande**

**1FN : Oui**

*Justification* : Tous les attributs sont fonctionnellement dépendants de la clé. Dans *DOMAINE*, même s’il y a des propriétés composées, **ADR\_DOMAINE** ne fait pas partie des attributs non clé. De plus, chaque valeur de la clé (**NUM\_DOMAINE**) ne correspond qu’à une seule valeur des attributs composés. Elle est ainsi en 1FN.

**2FN : Oui**

*Justification* : Toutes les relations sauf ‘*ETRE\_ASSOCIE\_A\_UN\_CEPAGE*’ et ‘*REGION*’ n’ont qu’une seule clé. Donc naturellement tout attribut non clé ne dépend pas d’une partie de cette clé. C’est-à-dire qu’ils sont en dépendance fonctionnelle élémentaire par rapport à la clé. Quant à ‘*REGION*’, elle est en 2FN étant donné qu’elle n’a pas d’attribut non clé. Quant à ‘*ETRE\_ASSOCIE\_A\_UN\_CEPAGE*’, vu que **POURCENTAGE\_CEPAGE** pour un vin donné est déterminé en même temps par **NUM\_CEPAGE** et **NUM\_VIN**, il est en dépendance fonctionnelle élémentaire par rapport à ces deux clés. La relation est ainsi en 2FN.

**3FN : Oui**

*Justification* :

*VIN :* Tous les attributs non clé sont indépendants des autres. Pour **NUM\_DOMAINE** et **NUM\_APPELLATION**, vu qu’ils se réfèrent à des entités différentes n’ayant pas d’associations entre elles. Ils sont ainsi indépendants.

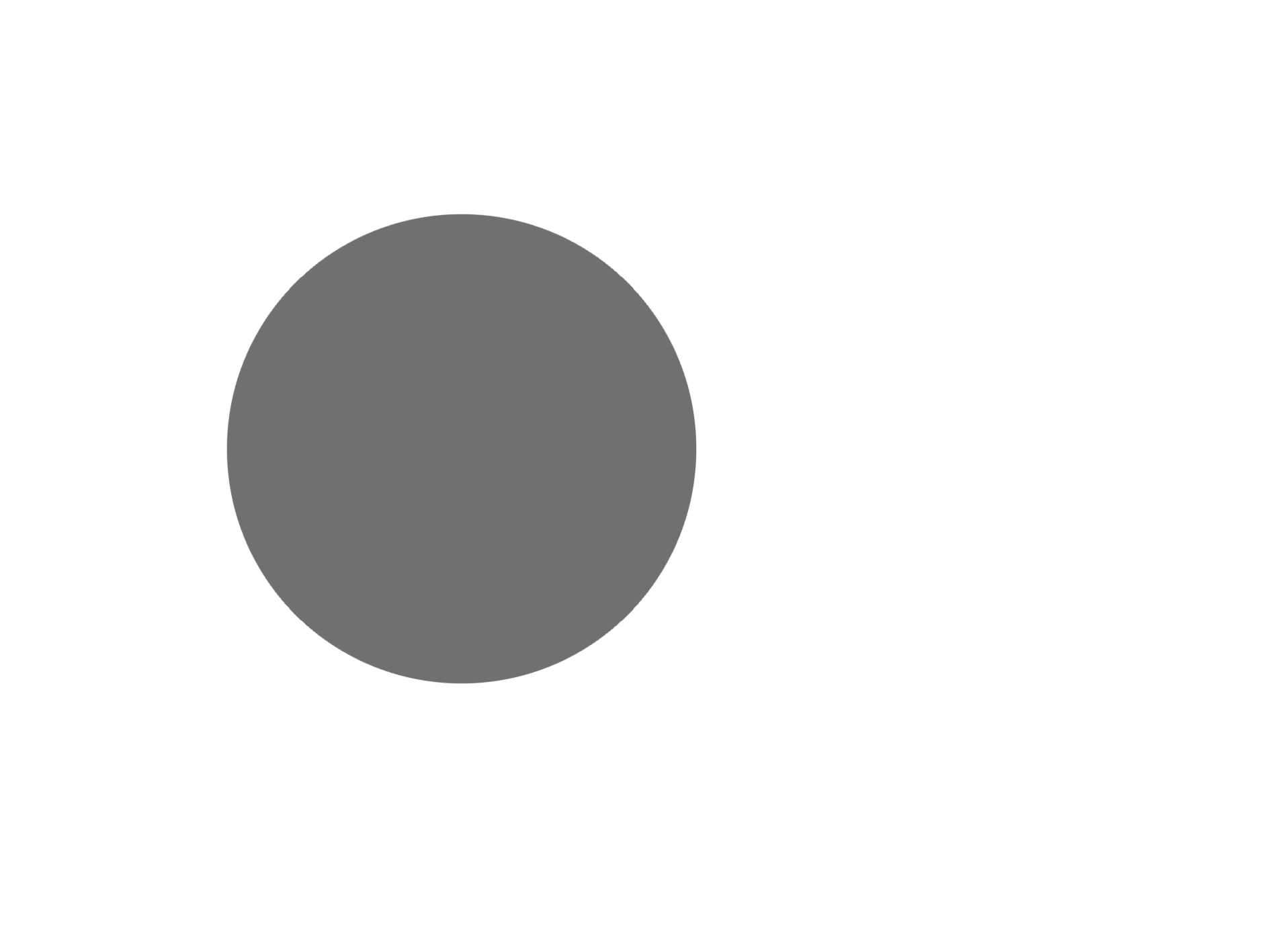
*APPELLATION :* Tous les attributs non clé sont indépendants des autres.

*DOMAINE* *:* Tous les attributs non clé sont indépendants des autres. Pour l’adresse, il existe des pays qui se situe dans deux continents. Donc **PAYS\_DOMAINE** et **CONTINENT\_DOMAINE** sont indépendants. Même dans un pays, il se peut qu’une ville a un homonyme, donc **VILLE\_DOMAINE** et **CP\_DOMAINE** sont aussi indépendants.

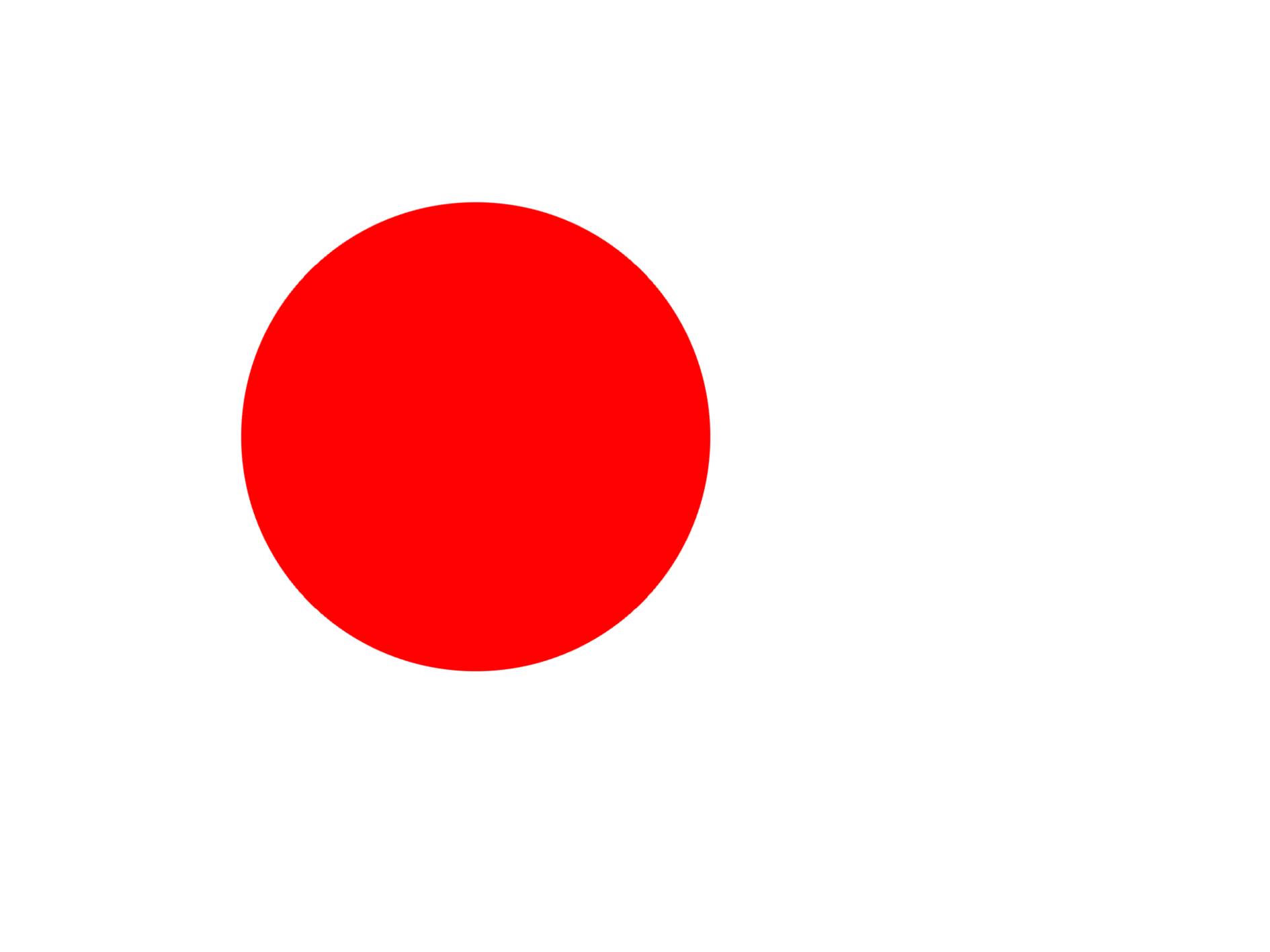
**Conclusion** : Toutes les relations dans MLR sont en 3FN, donc le MLR global est normalisé.

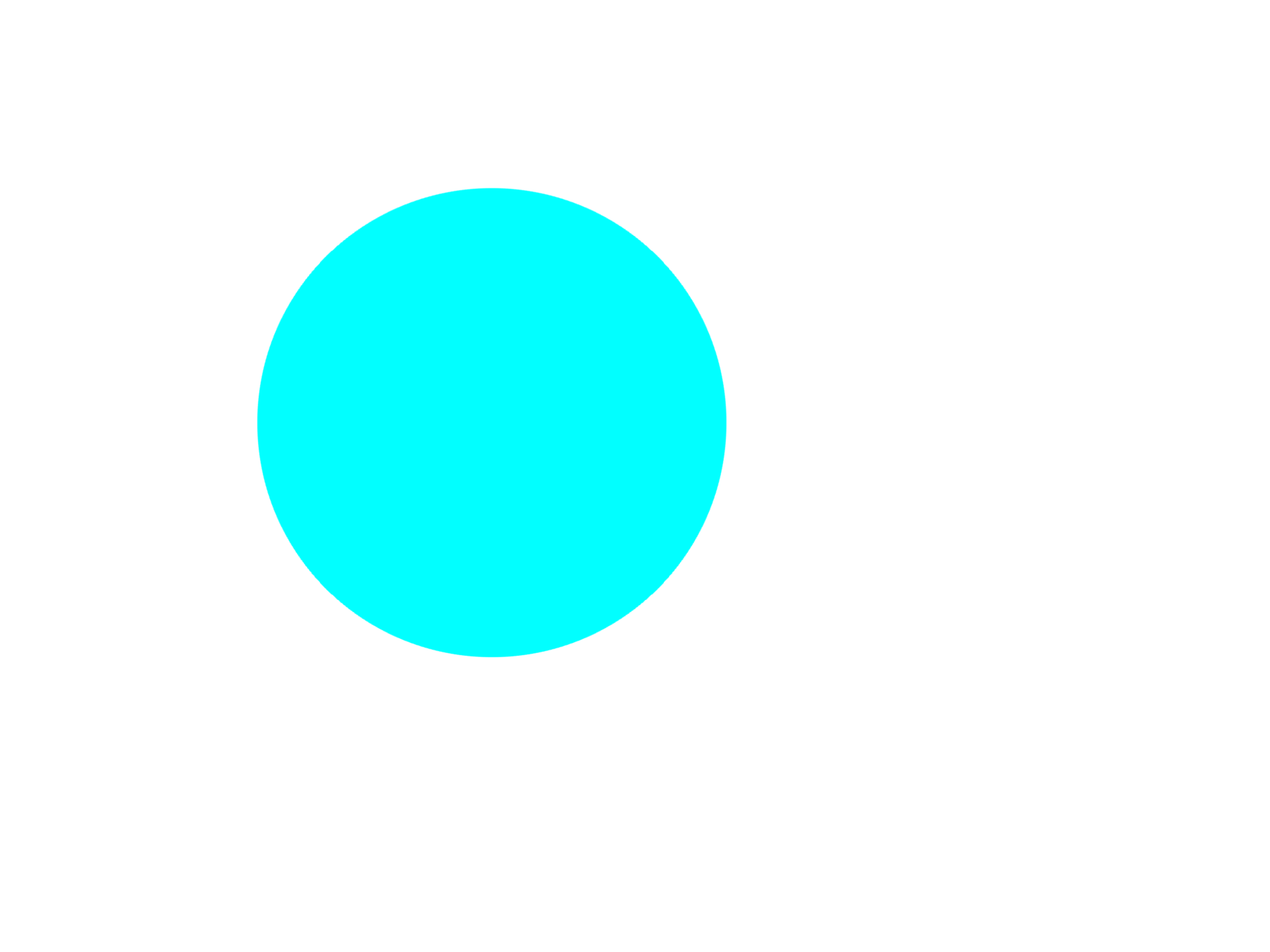
# La modélisation de l’IHM

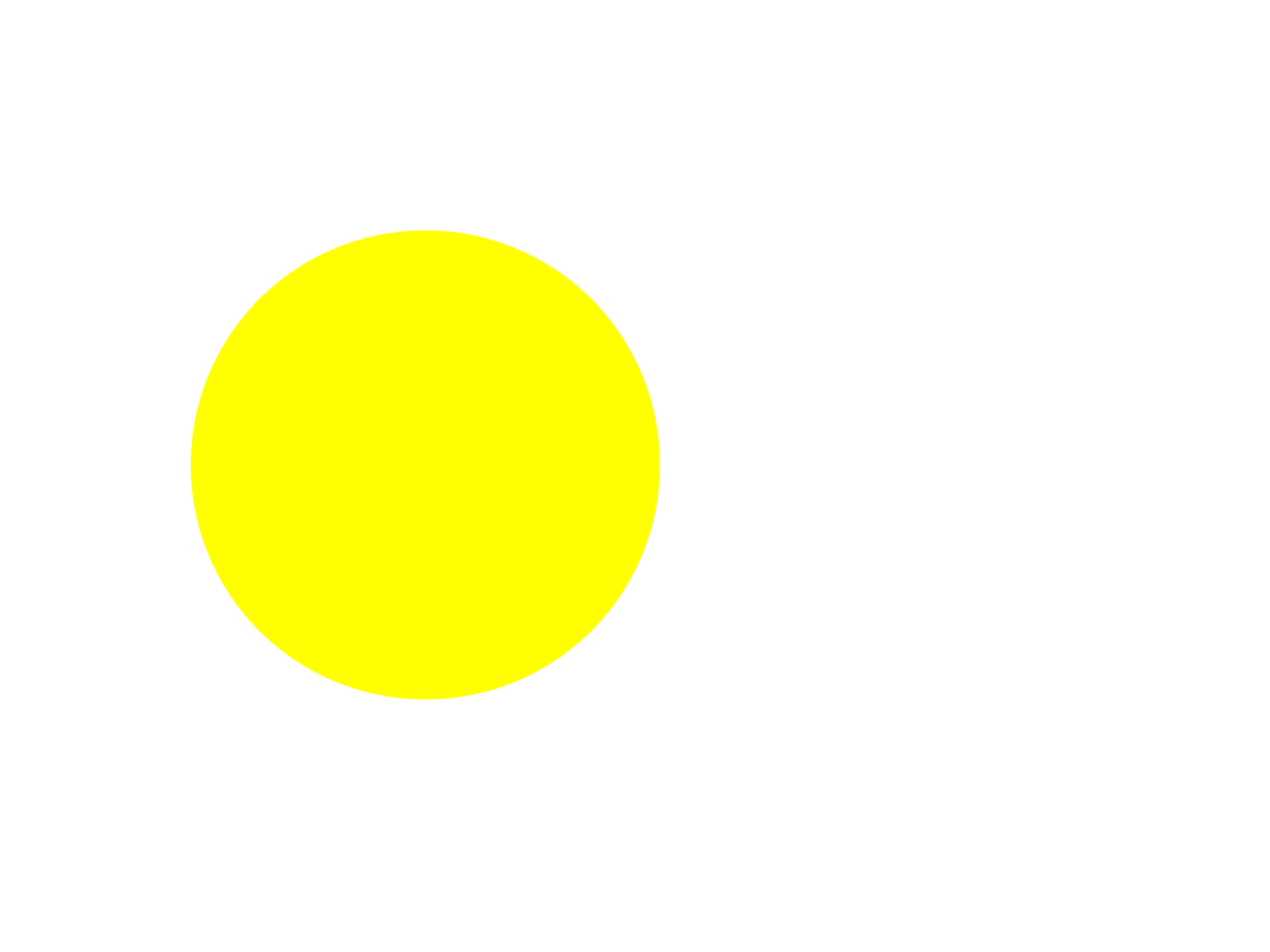
En réfléchissant à l’organisation orientée métier de votre application, dans le but d’être utilisée par une personne lambda, nous avons alors découpé notre IHM en quelques formulaires qui nous semblaient être les plus pertinents :

Informations non modifiables

Informations pouvant être modifiées

Informations obligatoires

Informations optionnelles

Informations calculées

En ouvrant son application, l'utilisateur peut voir les informations de base de tous les vins de la cave au bas de la page. Il a pu sélectionner le vin dans le formulaire pour afficher les informations spécifiques. Il peut également trouver des informations sur le vin en entrant le nom du vin et en limitant la couleur.

De même, il peut modifier les informations spécifiques du vin en accédant au bouton « modifier ». Ce qui nous donne la maquette suivante :



Figure ‑ Affichage principal - Liste des vins

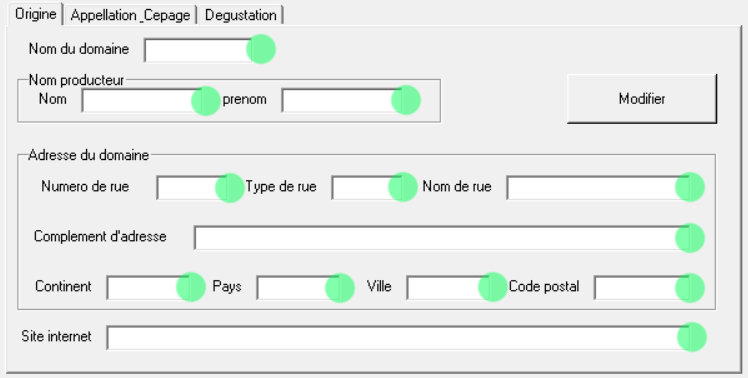


Figure ‑2 Affichage principal - L’onglet de l’origine du vin

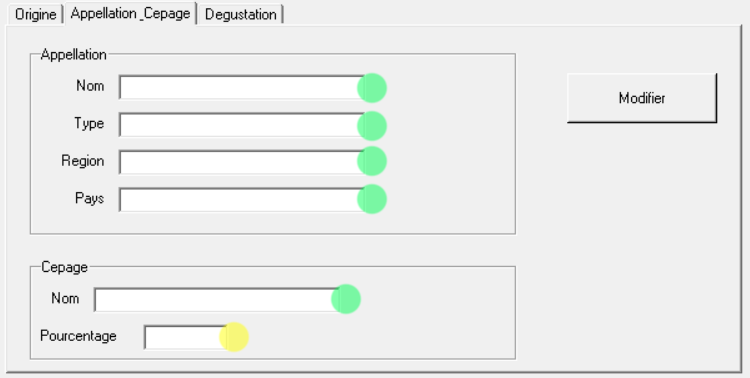


Figure ‑3 Affichage principal - L’onglet de l’appellation et la cépage du vin

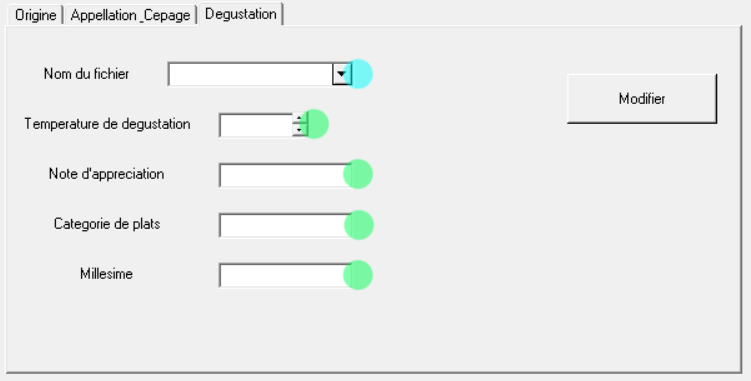


Figure ‑4 Affichage principal - L’onglet de la degustation du vin

Considérons l'enlèvement du vin. L'utilisateur peut renseigner les noms des vins qu'il souhaite retirer et toutes les bouteilles du même nom seront affichées au bas de la page. L'utilisateur retire le vin en sélectionnant la bouteille.

Le même utilisateur peut également ajouter une nouvelle bouteille en sélectionnant le nom du vin puis en renseignant les informations de base du vin.

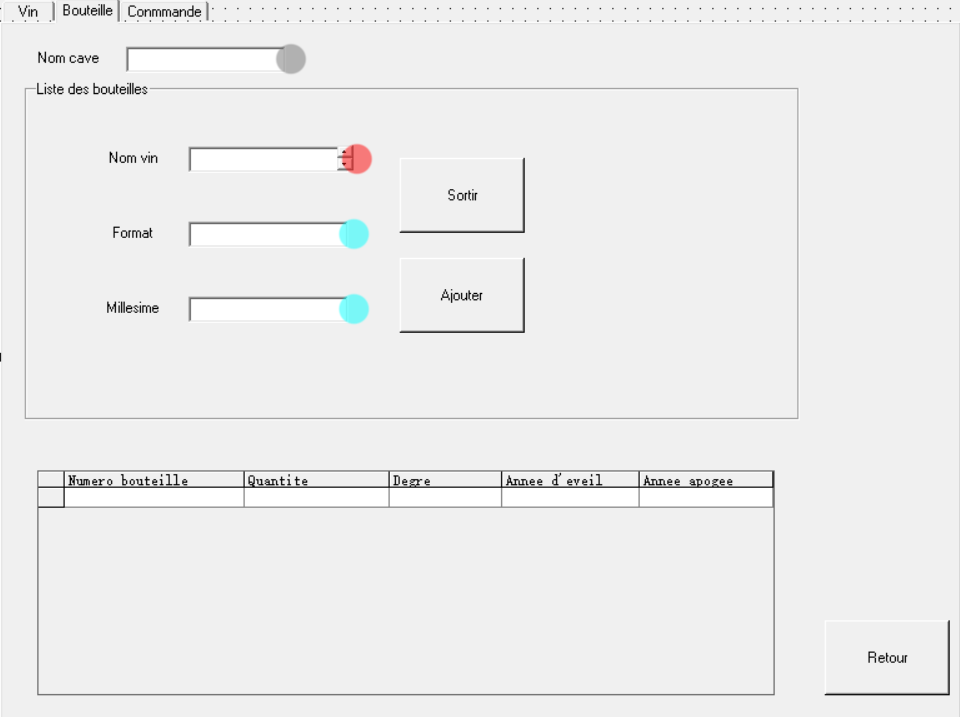


Figure ‑5 Sortir et ajouter bouteille

De même, nous avons une page pour voir toutes les informations de commande.

En entrant l'année, le mois et le jour, le tableau ci-dessous affiche les informations de commande. En sélectionnant les informations de commande, toutes les informations de bouteille contenues dans la commande seront affichées en bas de la page.

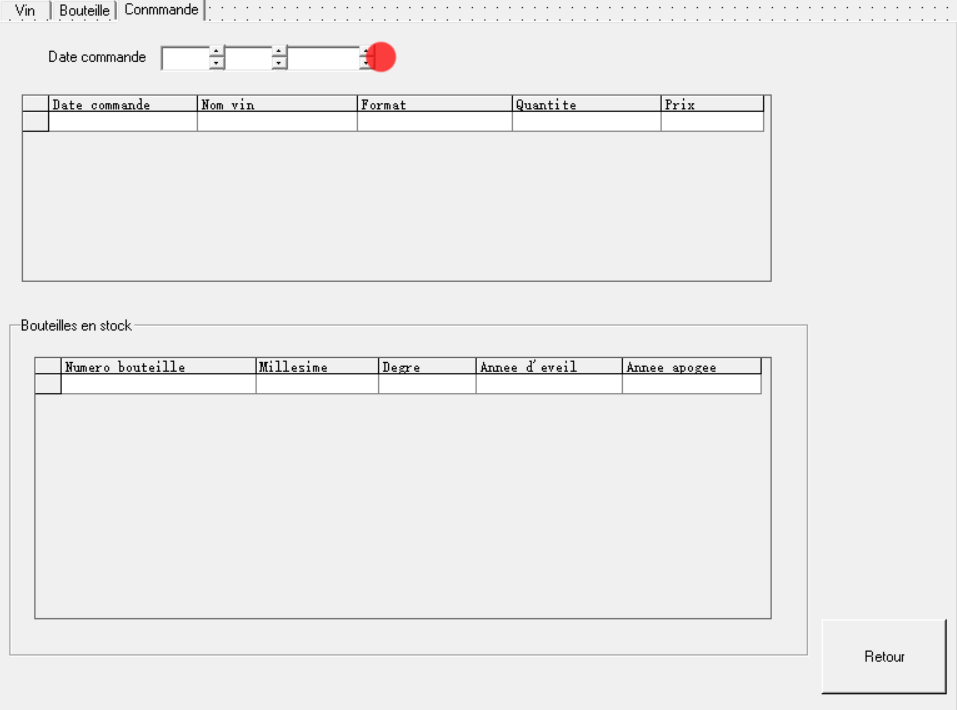


Figure ‑6 Voir les informations de commande

Il peut commander du vin via notre application. Les informations sur le vin rouge pouvant être commandées sont affichées en bas de la page. L'utilisateur peut choisir le vin ou entrer le nom du vin pour déterminer le vin à acheter. L'utilisateur entre ensuite les informations de base et l'application calculera automatiquement le prix de l'achat. L'utilisateur peut commander du vin en appuyant sur le bouton « Commander ».

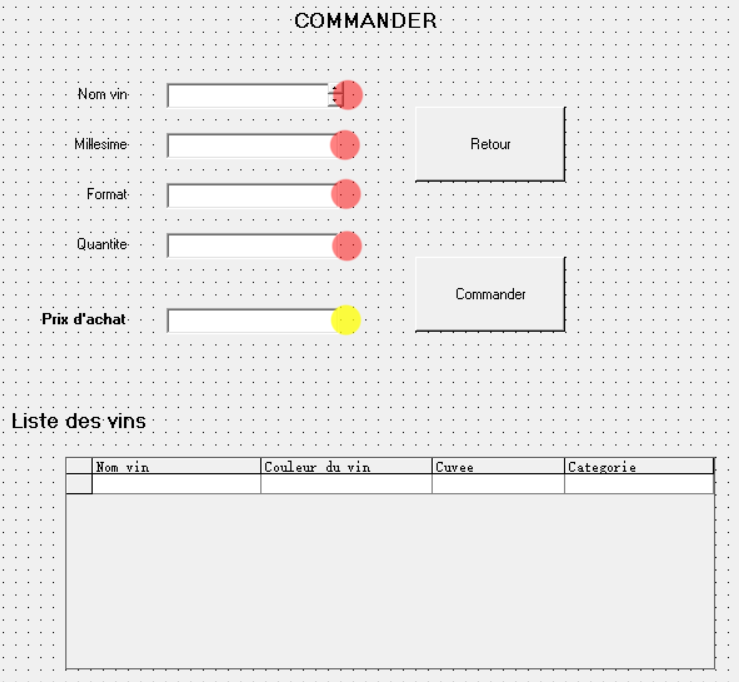


Figure ‑ Visuel de commande de bouteilles

# Dossier de développement du projet

## Présentation de l’interface développée avec MS/ACCESS

## Présentation des requêtes développées

## Présentation du code VBA développé dans l’application

## Présentation des options de sécurité

Nous avons sécurisé l’application de diverses manières.

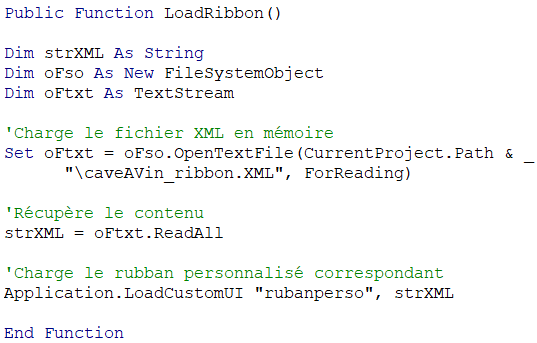
Tout d’abord, il nous a semblé important que l’utilisateur n’ait pas accès à toutes les options proposées par Access. Autrement, il aurait pu créer de nouvelles tables, requêtes… Pour cela, nous avons mis en place un ruban personnalisé.

Celui-ci est implémenté grâce au fichier suivant :

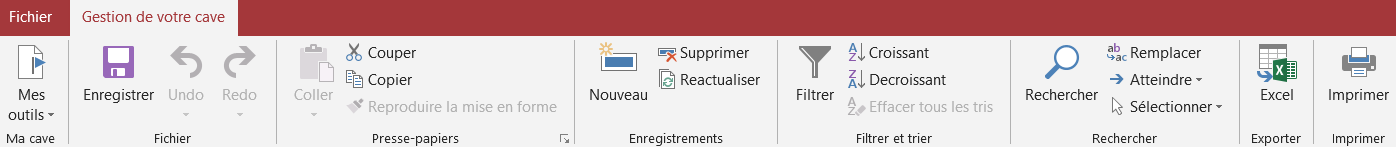
caveAVin\_ribbon.XML



Il est ensuite chargé grâce à la macro/fonction VBA suivante :



Celle-ci est à son tour appelée par la macro AutoExec qui est exécuté au démarrage de l’application. Ce qui nous donne le ruban suivant :



De plus, la macro AutoExec

Nous avons également mis en place une Application permettant De verrouiller l’accès aux données. En effet, La sécurité d’origine n’était pas suffisante. Il suffisait d’appuyer sur la touche MAJ au démarrage de l’application afin d’accéder aux données, formulaires, etc. Par le biais de cet applicatif, nous avons donc la possibilité de désactiver la touche « MAJ », et ainsi de verrouiller l’accès aux tables.



# Sources

***Sites internet***

<https://www.linternaute.com/homme/art-de-vivre/1011553-appellation-cepage-domaine-quelles-differences/>

<https://www.le-vin-pas-a-pas.com/4-niveaux-a-connaitre-pour-bien-classer-les-vins/>

***Consultation de diverses personnes ayant des connaissances en vitiviniculture et même en œnologie***

***Toutes les images présentes dans notre rapport, sont citées dans nos sources, libres de droits ou alors réalisées par nos propres soins.***