



Ecole Supérieure Privée
d'Ingénierie et de Technologies



Rapport réalisé par la team iEco

Classe : 2A18

Les membres de l'équipe :

-  Axel Essimi
-  Mintoua Toupkandi
-  Bilel Taktak
-  Kenza Khalfallah
-  Yacine feki
-  Yassine Essghaier

TUNIS le 12/06/2020

Rapport réalisé pour la société LA FERME



Table des matières

.....	2
1 Introduction générale	6
2 Présentation du projet	7
2.1 Introduction	7
2.2 Présentation de la thématique.....	7
2.3 Description du projet	7
2.4 Public et marché visés.....	8
2.5 Applications concurrentes	8
2.6 Originalité	8
2.7 Action Marketing	9
2.8 Conclusion	9
3 Analyse et Spécifications des besoins.....	10
3.1 Introduction	10
3.2 Identification des acteurs.....	10
3.3 Spécification des besoins fonctionnels.....	10
3.4 Spécifications des besoins non fonctionnels	10
3.5 Diagrammes des cas d'utilisation générale	11
3.6 Diagrammes des cas d'utilisation détaillés.....	11
3.7 Diagrammes de classe d'analyse	16
3.8 Conclusion	17
4 Conception.....	18
4.1 Introduction	18
4.2 Diagramme de classe détaillé.....	18
4.3 Conclusion	19
5 Base de données	19
5.1 Introduction	19
5.2 Le modèle Entités-associations	19
5.3 Le modèle relationnel	20

5.4	Conclusion	21
6	Réalisation	22
6.1	Introduction	22
6.2	Environnement de travail.....	22
6.3	Charte graphique	22
6.4	Modules (imprimés écrans)	23
6.5	Conclusion	27
7	Conclusion générale et perspectives	27

Table des figures

Figure 1 DCU general.....	11
Figure 2 DCU ouvrier	12
Figure 3 DCU conges	13
Figure 4 DCU animaux.....	14
Figure 5 DCU achat.....	15
Figure 6 DCU vente	16
Figure 7 Diagramme de classe.....	18
Figure 8 Modele entite association	20
Figure 9 modele relationel	20
Figure 10 page d'authentification.....	23
Figure 11 code source 1	24
Figure 12 page d'accueil	24
Figure 13 code source 2	25
Figure 14 liste des animaux.....	25
Figure 15 sous menu commande	26
Figure 16 interface gestion de la nourriture	27
Figure 17 charte graphique	23

1 Introduction générale

Le 21^e siècle encore appelé aujourd'hui ère du numérique est une période du monde au cours de laquelle tout en général tend à être numérisé et c'est dans ce sens que l'on voit apparaître de plus en plus d'applications à usage variées.

En effet à la vue du succès inouï que rencontre ces différentes applications sur le marché l'on rencontre de plus en plus de personnes qui se lancent dans les nouvelles technologies enfin d'en faire leur gagne-pain et c'est ainsi que l'on verra une multiplicité de langages de programmation faire leur apparition au cours des années mais également leurs évolutions tout en ayant des particularités qui leur permettent de résoudre des problèmes bien précis comme c'est le cas du langage assembleur pour la programmation des objets embarqués, du html pour la réalisation des pages web mais plus particulièrement les langages orientés objets pour pouvoir créer des objets de façon abstraite bien évidemment mais avec des traits d'usage qui sont les leurs dans la vie réelle. En ce sens dans cette grande et large famille l'un des premiers langages est bien évidemment le C++ successeur du célèbre C. Désirant faire partie de cette course que mène le monde à sa numérisation, nous l'équipe iEco avons décidé de mettre sur le marché notre propre application qui permettra on l'espère d'améliorer et de faciliter la vie de nombreuses personnes avec pour but principal de devenir l'un des acteurs majeur dans notre domaine d'activité.

2 Présentation du projet

2.1 Introduction

Dans les lignes qui suivront, nous débiterons par la présentation, entière et détaillée et progressive de notre projet pour que vous puissiez dans un premier temps vous en faire une première idée avant la phase d'analyse et de spécifications des besoins qui vous permettrons de mieux cerner notre idée.

2.2 Présentation de la thématique

Une ferme intelligente, vous l'aurez sans doute compris c'est l'idée de base de projet d'envergure. En effet nous avons pour objectif la réalisation d'une application complète de gestion d'une ferme intelligente n'ayant pas d'interface client donc il s'agit bien ici là d'une application desktop permettant d'automatiser certaines tâches qu'effectuent généralement les ouvriers de façon manuelle et parmi celles-ci nous pouvons parler de la gestion des profils des différents ouvriers mais également de la nourriture, des animaux, la mise à jour des différents statistiques de la ferme, la surveillance des différents espaces verts mais également de l'automatisation précède d'une identification des différentes ouvertures de la ferme.

2.3 Description du projet

De façon simple et concise notre projet consiste en la réalisation d'une application desktop n'ayant pas d'interface cliente avec le langage de programmation **C++** via le **Framework Qt** pour la réalisation des interfaces utilisateurs. Quant à la partie données, nous mettrons sur pied une base de données avec le logiciel mondialement connu **Oracle** mais ayant effectué toutes les analyses nécessaires au préalable c'est-à-dire les différents diagrammes de cas d'utilisation, de séquence système, diagramme de classe, les différents modèles entités associations grâce au logiciel **starUML**.

2.4 Public et marché visés

Dans ce projet le principal public cible sont tous les acteurs des domaines agricoles, tels que les fermiers, les petites ou moyennes entreprises du domaine, voir même les grands groupes internationaux tels que la banque agricole nationale.

2.5 Applications concurrentes

Après analyse du marché nous avons pu constater qu'il existait déjà de nombreuses solutions similaires à la nôtre comme c'est le cas des logiciels suivant, isaCompta qui a mis l'accent sur gestion complète du personnel agricole et de la comptabilité générale de l'exploitation ce logiciel est développé par la société Isagri, comme autre pionnier du domaine nous avons **EKYLIBRE** qui fait dans la gestion parcellaire et de la production végétale, gestion de la production, des ventes et des achats et développe par le groupe **EKYLIBRE** lui-même. Il en existe de nombreux autres tels que **AGRI4D**, **AgExpert**, AgreoSolution et bien d'autres encore.

2.6 Originalité

A la vue des nombreuses solutions que l'on trouve sur le marché mondial pour aider les acteurs du milieu agricole dans la gestion de leurs fermes, nous nous devons de faire preuve d'innovation dans notre produit pour pouvoir nous démarquer des autres et c'est dans cette optique que nous misons sur l'accessibilité à tous les supports c'est-à-dire sur **toutes les plateformes possibles, la sécurité, la prise en main intuitive de notre produit et ajouté 2 heures de formation offertes pour une meilleure prise en main, cartographie GPS de l'exploitation, assistance par mail, traçabilité des différents produits, un conseiller à portée de main et bien d'autres encore.**

2.7 Action Marketing

A la vue de la multitudes d'applications que comporte les plateformes telles que Play Store ou App store l'on s'aperçoit qu'il est important de miser sur une très bonne action marketing voilà pourquoi chez iEco nous avons décidé, pour la phase de préplacement de créer une liste d'attente ou les personnes intéressées pourront s'inscrire ainsi ils pourront nous donner une idée précise sur ce qu'ils attendent de notre produit, tester notre application en version beta avant de l'ouvrir au grand public, tout en les assurant qu'ils seront prévenus a la sortie de l'appli par mail pour ainsi profiter d'un effet amplificateur viral et après la sortie de notre produit lancer notre site web et se faire connaître sur les réseaux sociaux ou encore à travers des posts sponsorises, des pubs avec des célébrités ou encore proposer des jeux pour attirer des personnes de tout âge.

2.8 Conclusion

Le projet ainsi présente vous permettra d'avoir une vague idée sur le produit final qui sera obtenu et dans quelles circonstances et mesurent, il sera réalisé pour répondre aux attentes en matière d'application actuel sur le marché mondial.

3 Analyse et Spécifications des besoins

3.1 Introduction

Avant toute réalisation d'un projet bien bâti et solide avec des fonctionnalités structurées, l'étape de l'analyse et la spécification des besoins est très importante dans la Conception UML, car elle permet de façon théorique bien évidemment de desceller les failles possibles du système, les améliorations possibles ou encore de montrer ses limites et voilà pourquoi dans les lignes qui suivront nous débuterons notre phase d'analyse et spécifications.

3.2 Identification des acteurs

Nos acteurs dans ce projet seront l'administrateur de l'application et les différents ouvriers car l'application ne possèdera pas d'interface utilisateur pour les clients de la ferme.

3.3 Spécification des besoins fonctionnels

Notre système doit permettre de s'identifier, ajouter, consulter, modifier, supprimer, rechercher des ouvriers, des animaux, de la nourriture, des abris d'animaux et des Commandes.

3.4 Spécifications des besoins non fonctionnels

Notre produit en dehors de ses besoins fonctionnels possède également des fonctionnalités avancées l'envoi de mail dès l'ajout d'un produit, l'impression d'un rapport statistiques, la recherche dynamique ou encore le tri.

3.5 Diagrammes des cas d'utilisation générale

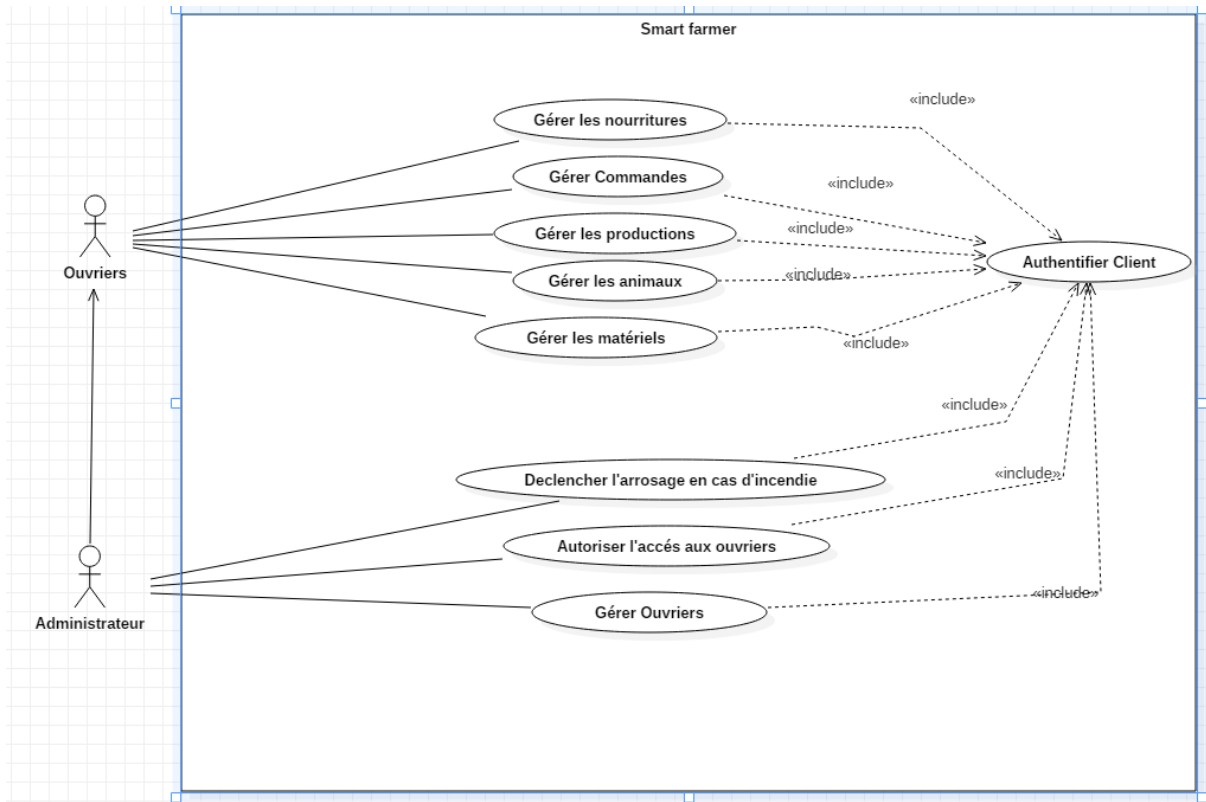


Figure 1 DCU general

3.6 Diagrammes des cas d'utilisation détaillés

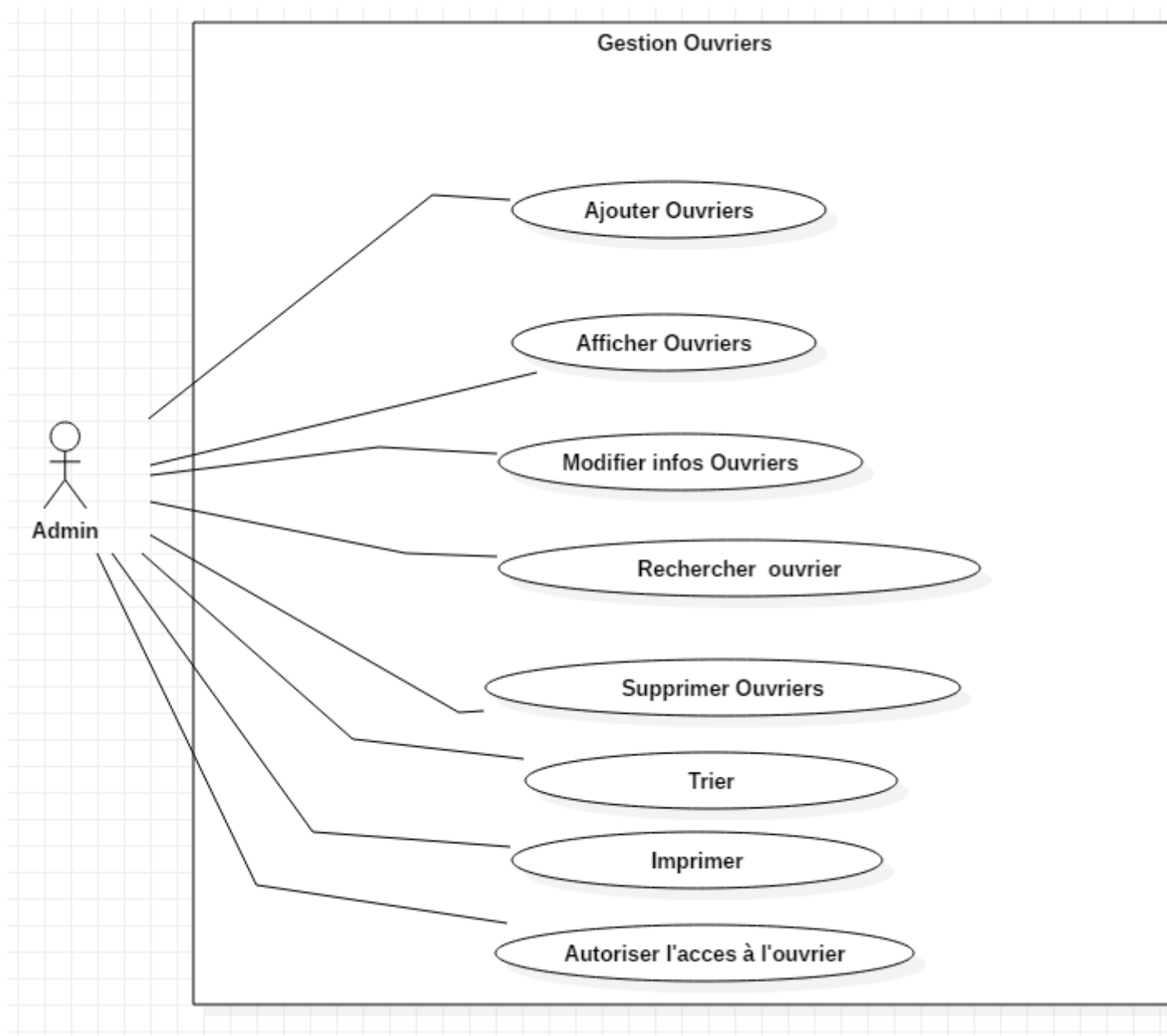


Figure 2 DCU ouvrier

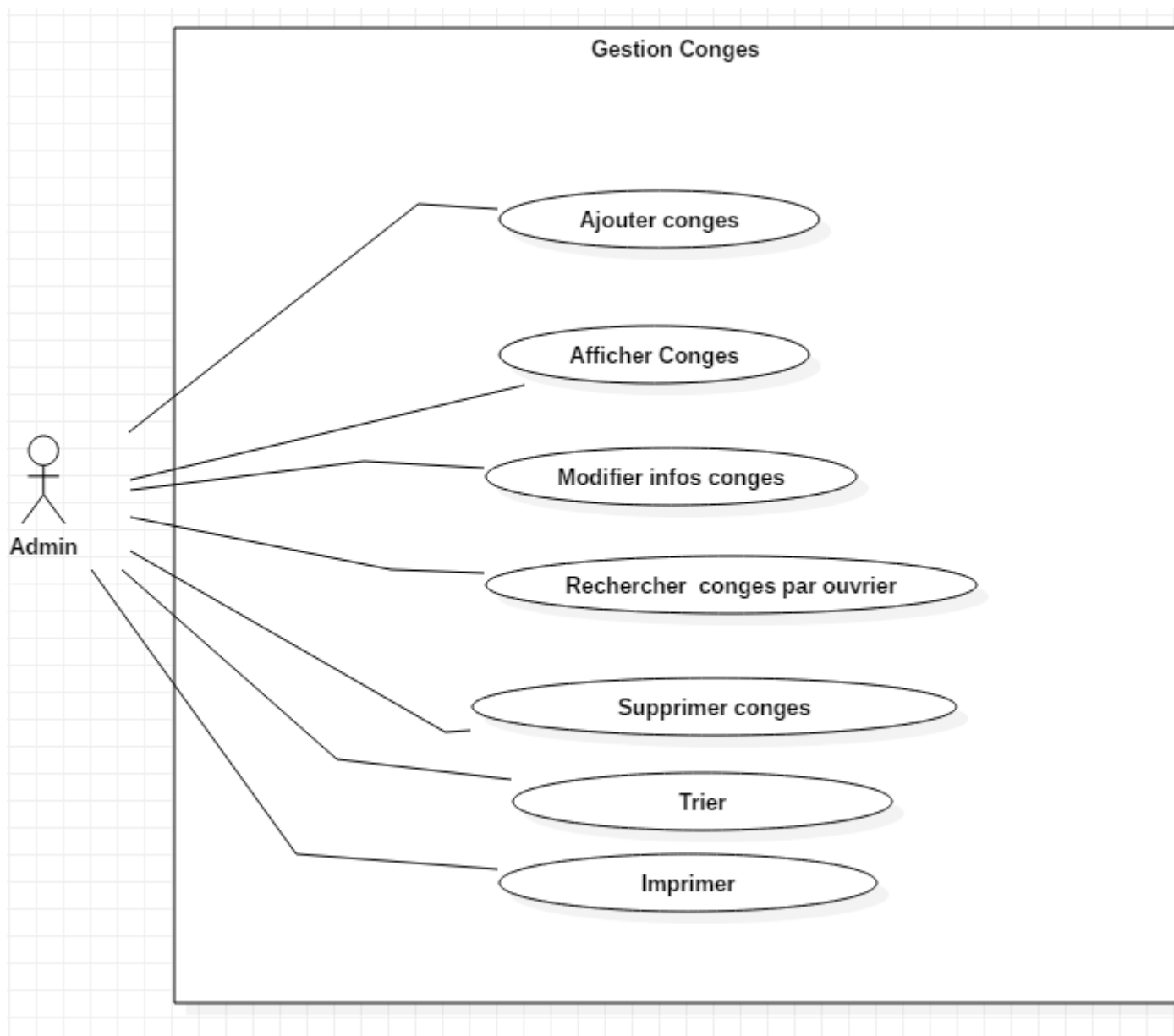


Figure 3 DCU conges

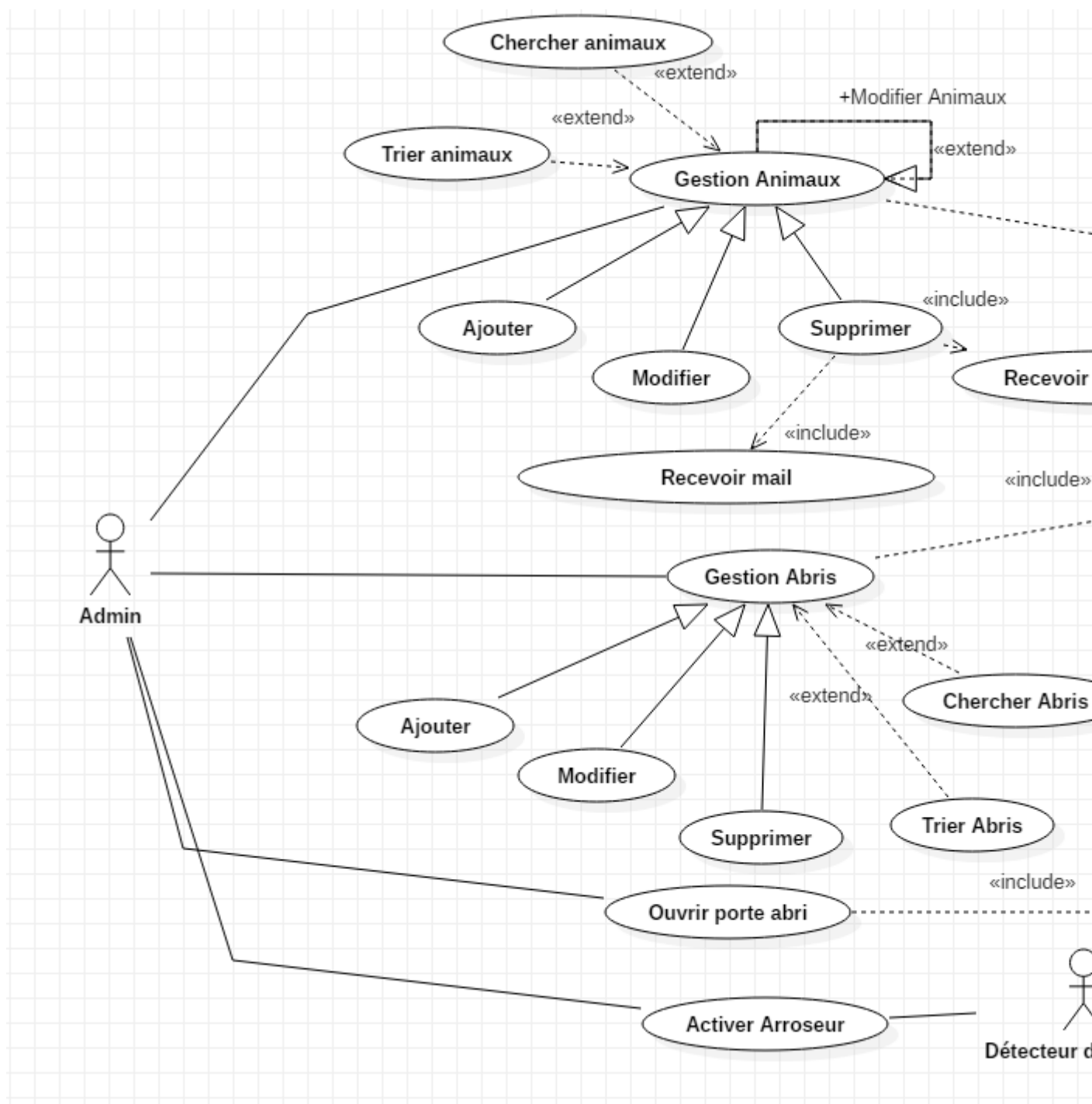


Figure 4 DCU animaux

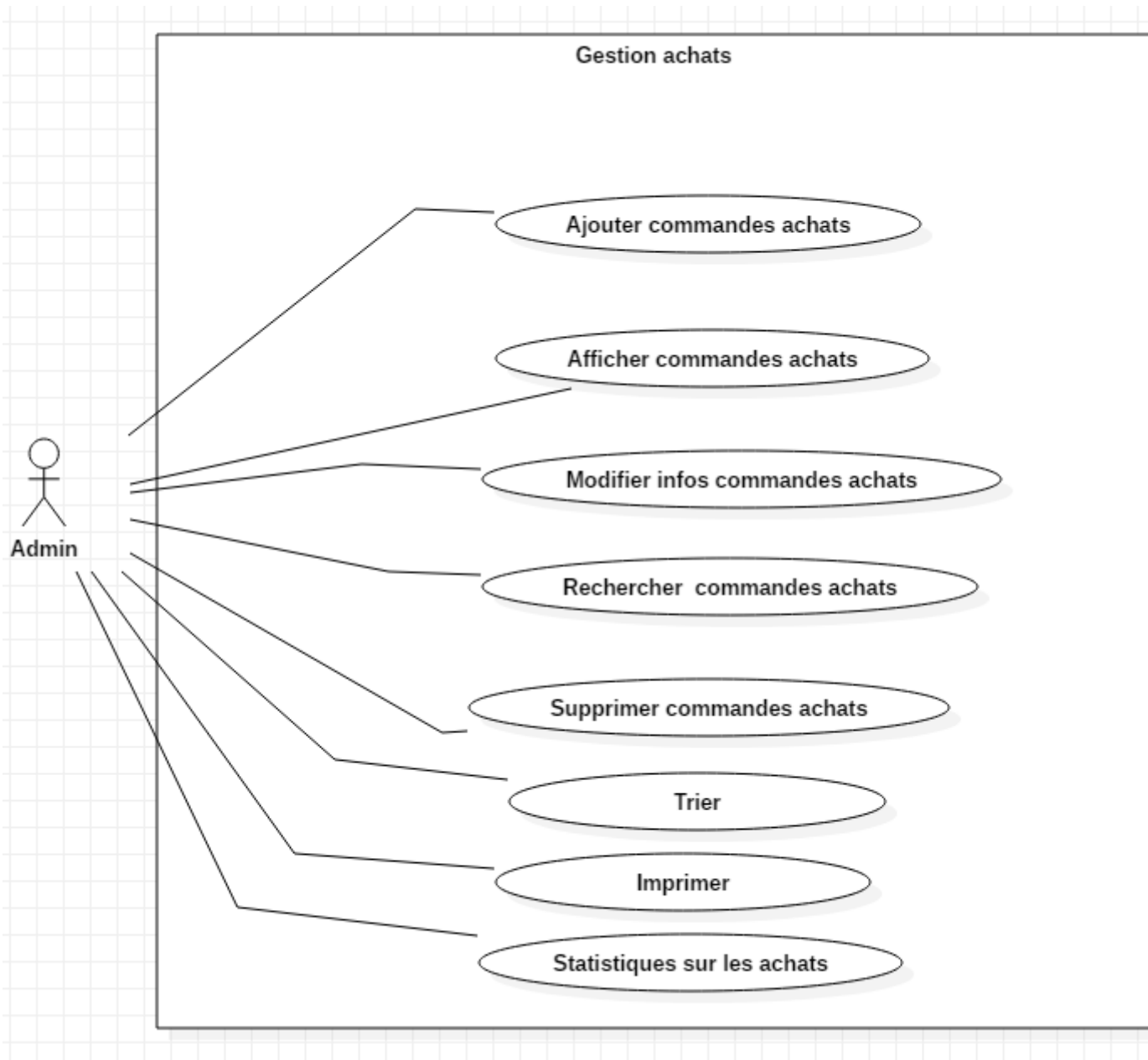


Figure 5 DCU achat

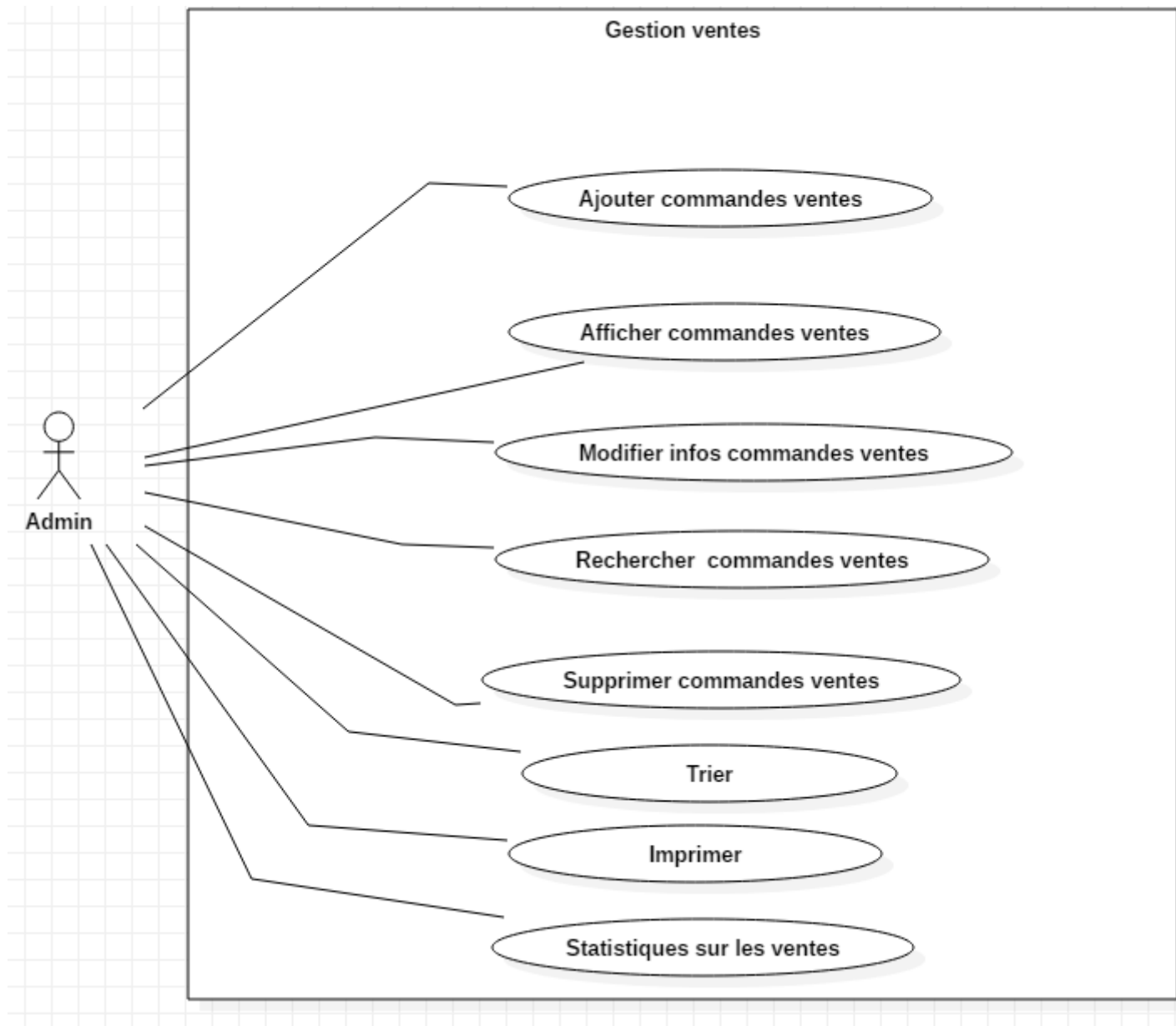
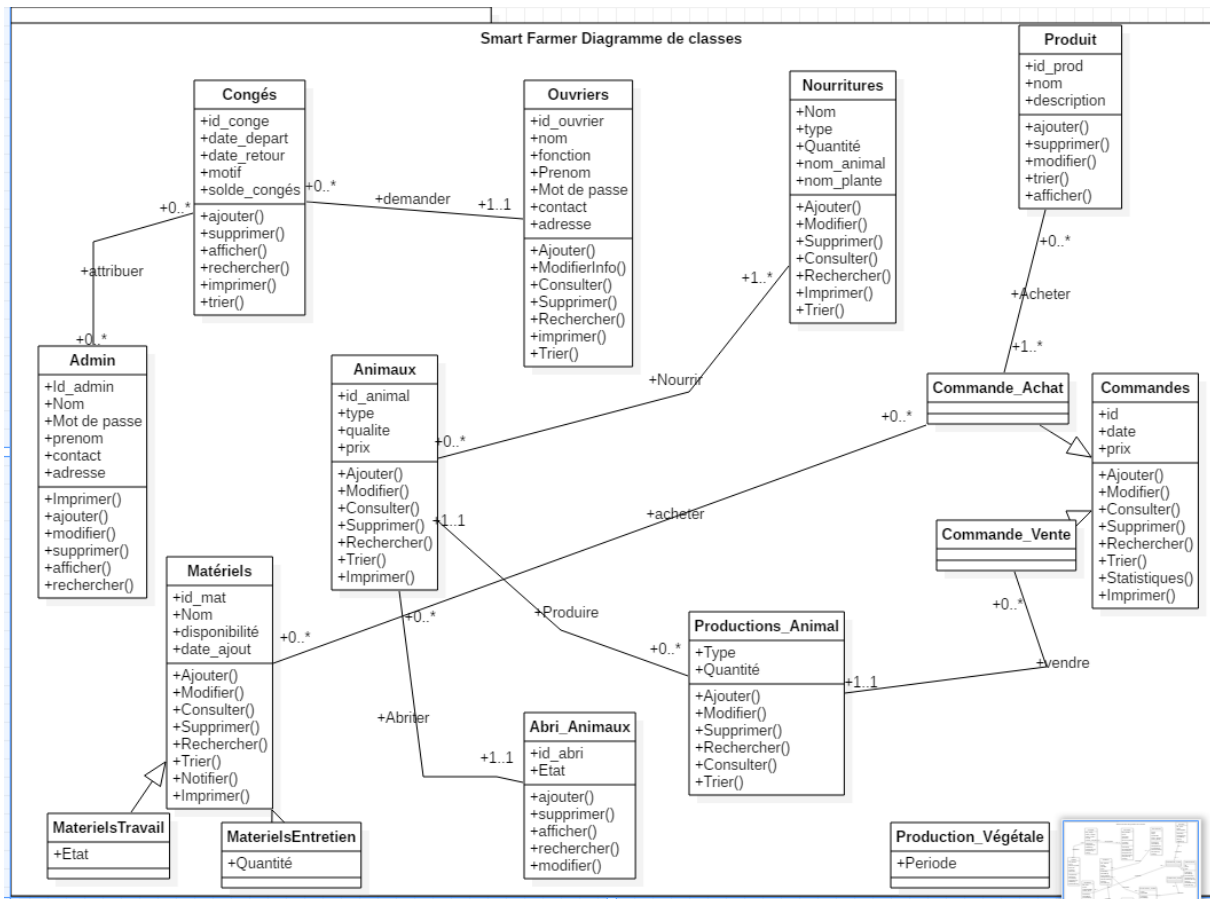


Figure 6 DCU vente

3.7 Diagrammes de classe d'analyse



3.8 Conclusion

Parvenus au terme de la phase d'analyse nous avons pu déterminer les différents intervenants les besoins du site, et la réalisation des différents diagrammes qui nous permettrons de mieux nous organiser avant la phase de conception.

4 Conception

4.1 Introduction

Après la réalisation de la phase d'analyse nous pouvons nous lancer dans la phase de conception qui consiste à structurer, organiser, planifier le projet. Située entre la définition initiale et la conduite à proprement parler des opérations.

4.2 Diagramme de classe détaillé

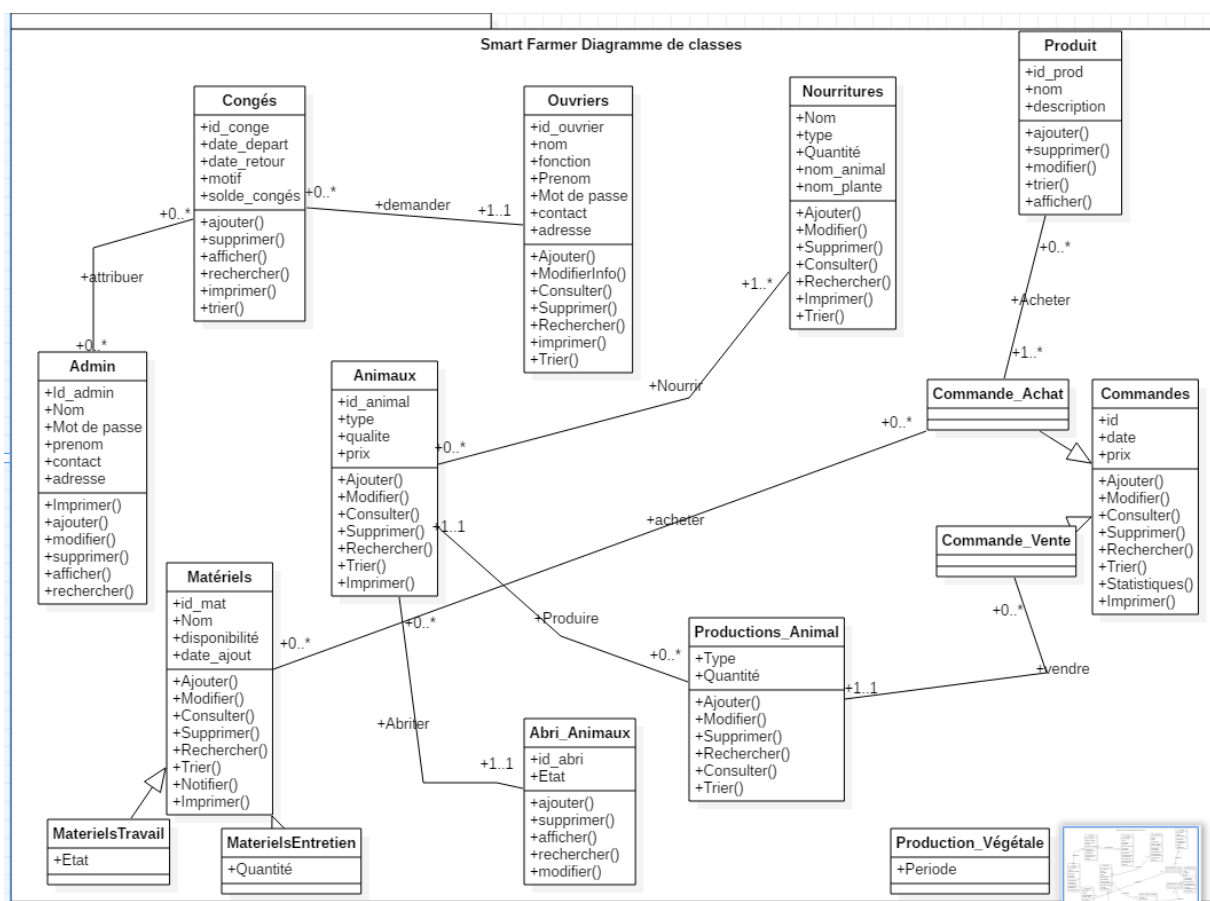


Figure 7 Diagramme de classe

4.3 Conclusion

La réalisation du diagramme de classe ainsi que l'architecture de l'application nous permet d'avoir une vision globale sur tout le programme, Il ne nous reste plus désormais qu'à implémenter le code pour la réalisation du site web.

5 Base de données

5.1 Introduction

Dans ce paragraphe nous vous présenterons notre démarche pour la réalisation de notre base de données bien évidemment qui nécessite des étapes à suivre au préalable tels que la réalisation du modèle entité association suivie de notre modèle relationnel de la base de données elle-même.

5.2 Le modèle Entités-associations

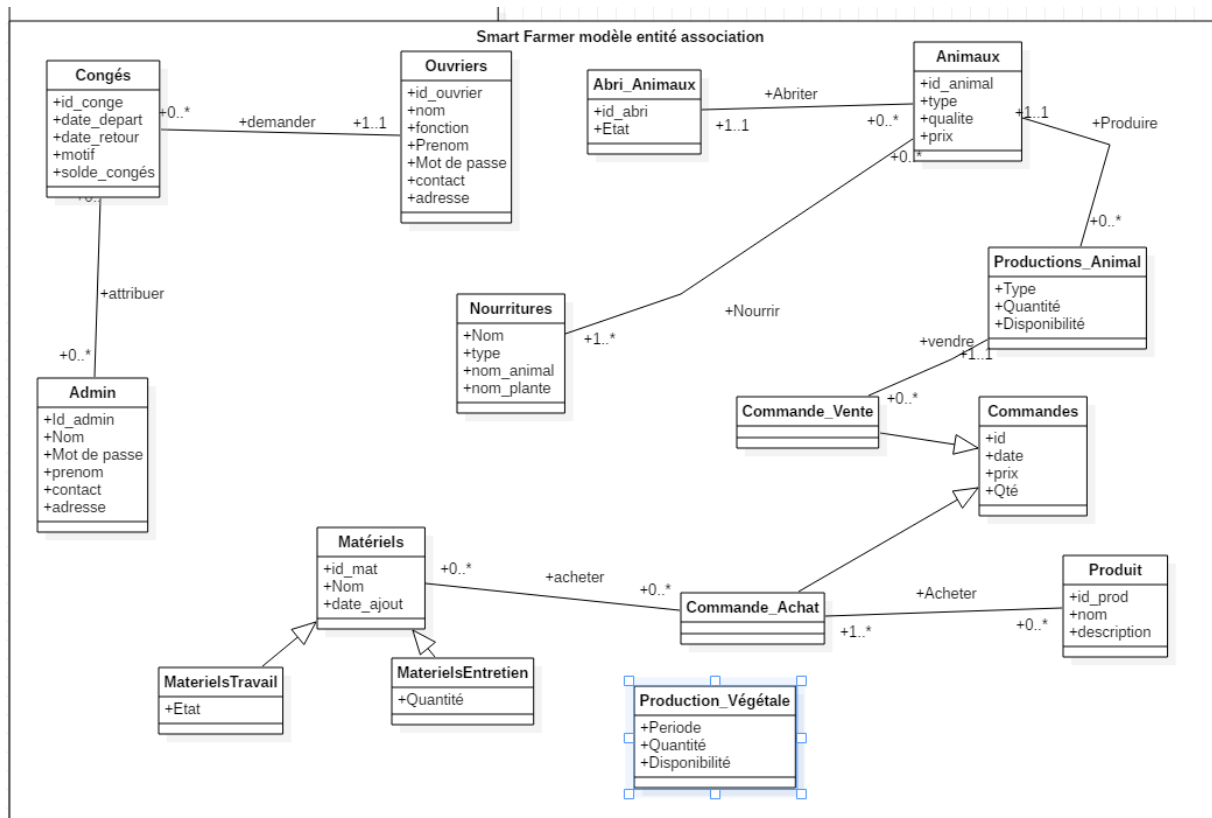


Figure 8 Modèle entité association

5.3 Le modèle relationnel

Modèle Relationnel
Ouvriers (id_ouvrier, nom_prenom, mot de passe, contact, ville, rue, codepostal)
Congés (id_congé, date_depart, date_retour, solde_congé, #id_ouvrier, #id_admin)
Admin (id_admin, nom_prenom, mot de pass, contact, ville, rue, codepostal)
Animaux (id_ani, surnom, race, qualité, prix, objectif, foyer, age)
Abris (id_abri, etat, type, capacité)
Commande_achat(id_achat, date, prix, qté)
Materiel_Entretien (#id_mat, nom, qté, date_ajout)
Materiel_Travail (#id_mat, nom, etat, date_ajout)
Nouriture_Animal (id_nouri, nom_aliment, animal, type)
Nouriture_Vegetal (id_nouri, nom_aliment, plant, type)
Commande_vente (id_vente, date, prix, qté)
Produit (id_produit, nom, description)

Figure 9 modele relationel

5.4 Conclusion

Ainsi nous achevons la phase de création de notre base de données qui est totalement conforme aux standards actuels sur le marché mondiale et donc nous entamons ainsi la dernière phase qui est celle de la réalisation de l'application.

6 Réalisation

6.1 Introduction

Dans les lignes qui suivront vous pourrez observer quelques images issues de l'application finale en particulier celles du code et du rendu de l'appli qui illustre la partie visible par les utilisateurs de l'application ou encore celle utilisée par l'administrateur pour effectuer ses différentes tâches, mais aussi vous pourrez voir les différentes palettes graphiques sélectionnées pour le projet.

6.2 Environnement de travail

Comme il l'a été mentionné précédemment, nous avons utilisé le **langage c++** pour la programmation de notre appli mais agissant de la création des interfaces utilisateurs nous avons fait usage du **Framework Qt** et ses différentes bibliothèques, combiné à ceci l'usage du logiciel **oracle** pour la gestion des bases de données, de **starUML** pour la réalisation des différents diagrammes, du logiciel **arduino** pour la programmation de tout ce qui est électronique dans notre travail, le partage des ressources entre les différents collaborateurs sur la plateforme **Git Hub**.

6.3 Charte graphique

Ci-joint une pièce qui présente les différentes palettes de couleurs présente dans notre application avec une prédominance de la couleur verte et ses différentes dérivées pour un petit clin d'œil à la nature.

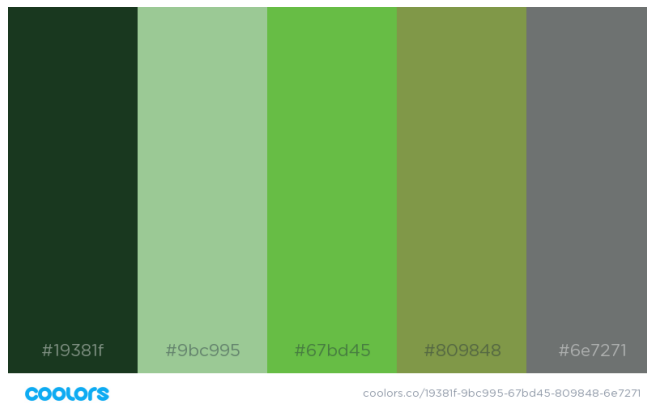


Figure 10 charte graphique

6.4 Modules (imprimés écrans)

Ci-dessous quelques images de notre réalisation finale

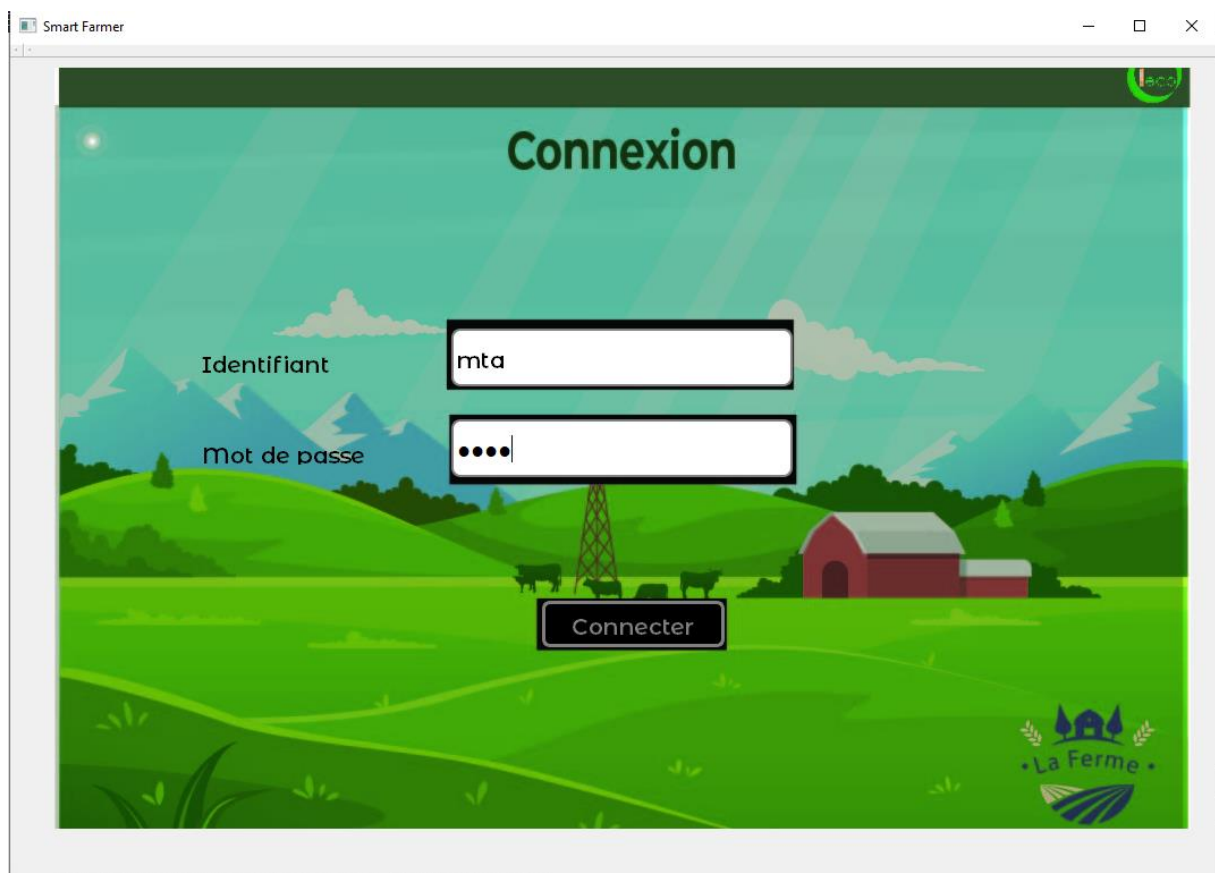


Figure 11 page d'authentification

```

64 int G_Ouvriers::get_contact()
65 {
66     return contact;
67 }
68
69 int G_Ouvriers::get_code()
70 {
71     return code_postal;
72 }
73
74 //fin getters *****
75
76 //debut des crud *****
77
78 bool G_Ouvriers::ajouter()
79 {
80     QSqlQuery query;
81     QString res=QString::number(id_ouvrier);
82     QString res2=QString::number(contact);
83     QString res3=QString::number(rue);
84     QString code=QString::number(code_postal);
85     query.prepare("INSERT INTO ouvrier (id_ouvrier, nom, prenom, mot_de_passe, contact, rue, code_postal, ville) VALUES (:id, :nom, :prenom, :pwd, :contact, :rue, :code, :ville)");
86     query.bindValue(":id", res);
87     query.bindValue(":nom", nom);
88     query.bindValue(":prenom", prenom);
89     query.bindValue(":contact", res2);
90     query.bindValue(":rue", res3);
91     query.bindValue(":code", code);
92     query.bindValue(":ville", ville);
93     query.bindValue(":pwd", password);
94     return query.exec();
95 }
96
97 QSqlQueryModel * G_Ouvriers::afficher()
98 {
99     QSqlQueryModel *model= new QSqlQueryModel();
100     model->setQuery("select * from ouvrier");
101     model->setHeaderData(0,Qt::Horizontal, QObject::tr("ID"));
102     model->setHeaderData(1,Qt::Horizontal, QObject::tr("NOM"));
103     model->setHeaderData(2,Qt::Horizontal, QObject::tr("PRENOM"));
104     model->setHeaderData(3,Qt::Horizontal, QObject::tr("PASSWORD"));
105     model->setHeaderData(4,Qt::Horizontal, QObject::tr("CONTACT"));
106     model->setHeaderData(5,Qt::Horizontal, QObject::tr("RUE"));
107     model->setHeaderData(6,Qt::Horizontal, QObject::tr("CODE POSTAL"));
108     model->setHeaderData(7,Qt::Horizontal, QObject::tr("VILLE"));
109     return model;
110 }

```

Sortie de l'application

LaFerme

QObject::connectSlotsByName: No matching signal for on_comboBox_tri_activated(QString)

virtual void QCPGraph::draw(QCPainter*) invalid key or value axis

QSoundEffect(audio): Error decoding source file: sons/button.mp3

28111221 E1(Etudes)ESPRET(Esprit)DAIR(Semestre 2)Projet 2A_C++(Qt_Formation)build-LaFerme-Desktop_Qt_5.14.1_MinGW_32_bit-Debug\smtp.exe exited with code 0

Figure 12 code source 1

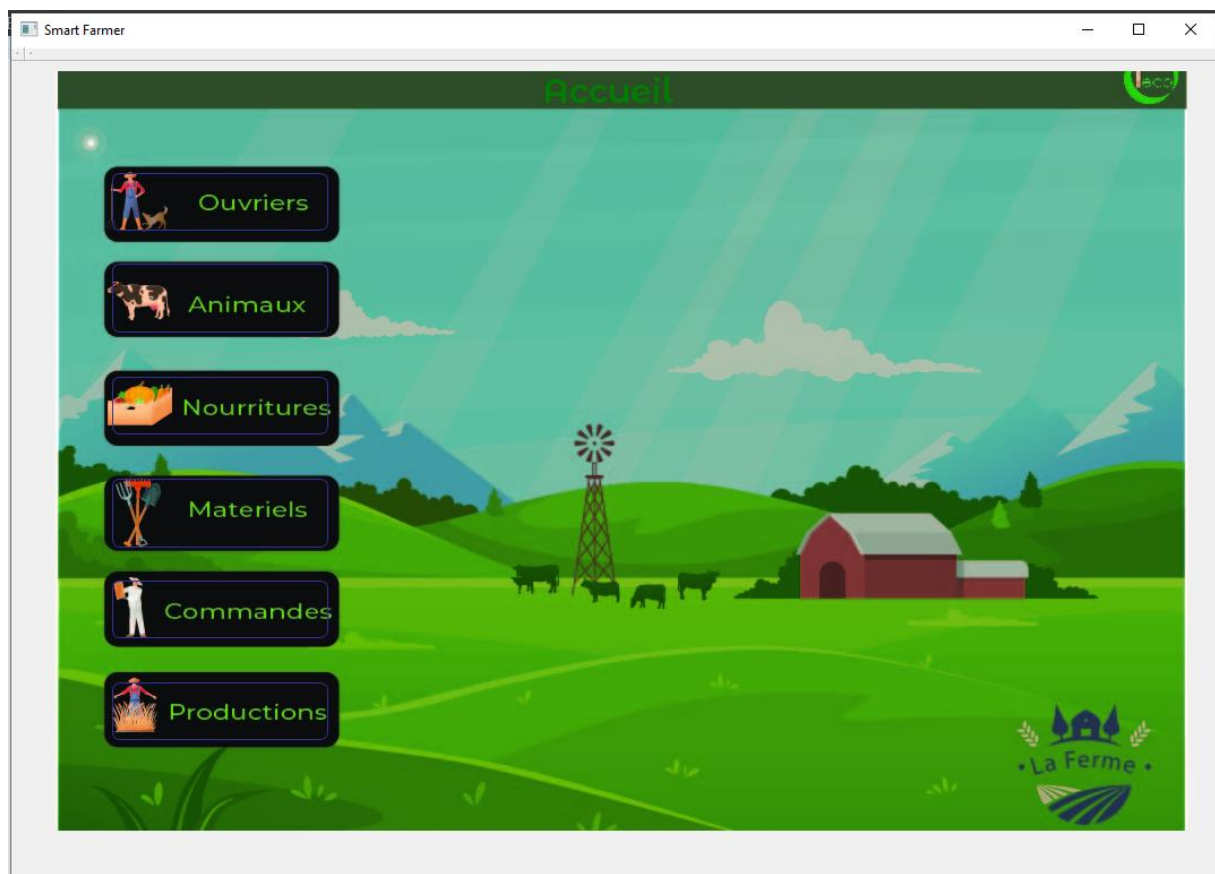


Figure 13 page d'accueil


```

85 void Interface::on_pushButton_modif_con_clicked()
86 {
87     int id = ui->lineEdit_ajout_con_id->text().toInt();
88     int id_admin = ui->lineEdit_ajout_con_id_admin->text().toInt();
89     int id_ouv = ui->lineEdit_ajout_con_id_ouv->text().toInt();
90     int soldes = ui->lineEdit_ajout_con_se->text().toInt();
91     QString date_dep = ui->lineEdit_ajout_con_date_d->text();
92     QString date_retour = ui->lineEdit_ajout_con_date_r->text();
93
94     Conges cong(id, id_admin, id_ouv, soldes, date_dep, date_retour);
95     bool test = cong.modifier(id);
96     if (test)
97     {
98         ui->tab_conges->setModel(tmp_cong.afficher()); //refresh
99
100     QMessageBox::information(nullptr, QObject::tr("Modification Réussi"),
101                             QObject::tr("Ajouté.\n"
102                                     "Click Cancel to exit."), QMessageBox::Cancel);
103
104     //ui->statusBar->showMessage(" ajouté");
105     }
106     else
107     {
108         QMessageBox::critical(nullptr, QObject::tr("Echec Modification"),
109                               QObject::tr("Erreur !.\n"
110                                       "Click Cancel to exit."), QMessageBox::Cancel);
111         //ui->statusBar->showMessage("Erreur");
112     }
113
114 void Interface::on_pushButton_supp_con_clicked()
115 {
116     int id = ui->lineEdit_ajout_con_id->text().toInt();
117     bool test = tmp_cong.supprimer(id);
118     if (test)
119     {
120
121     }
122 }

```

Figure 14 code source 2

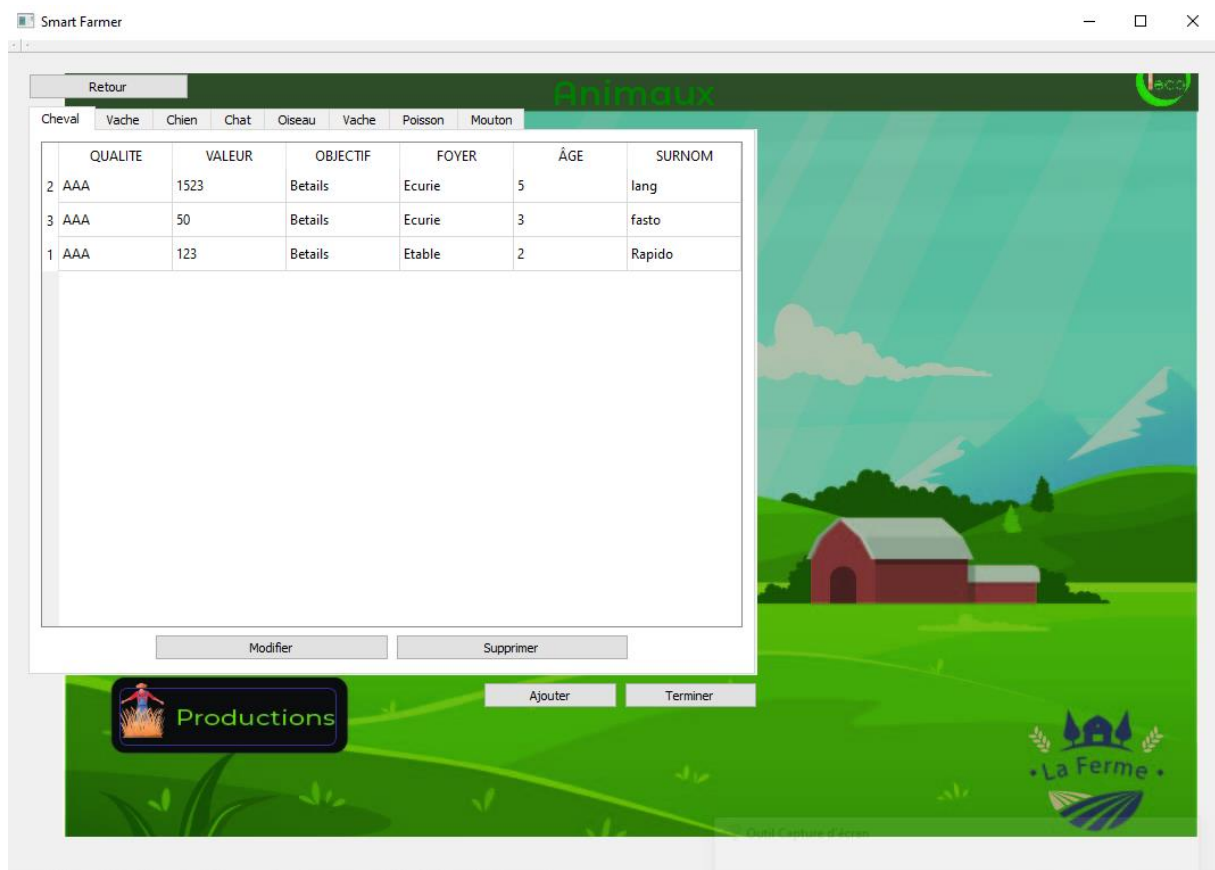


Figure 15 liste des animaux

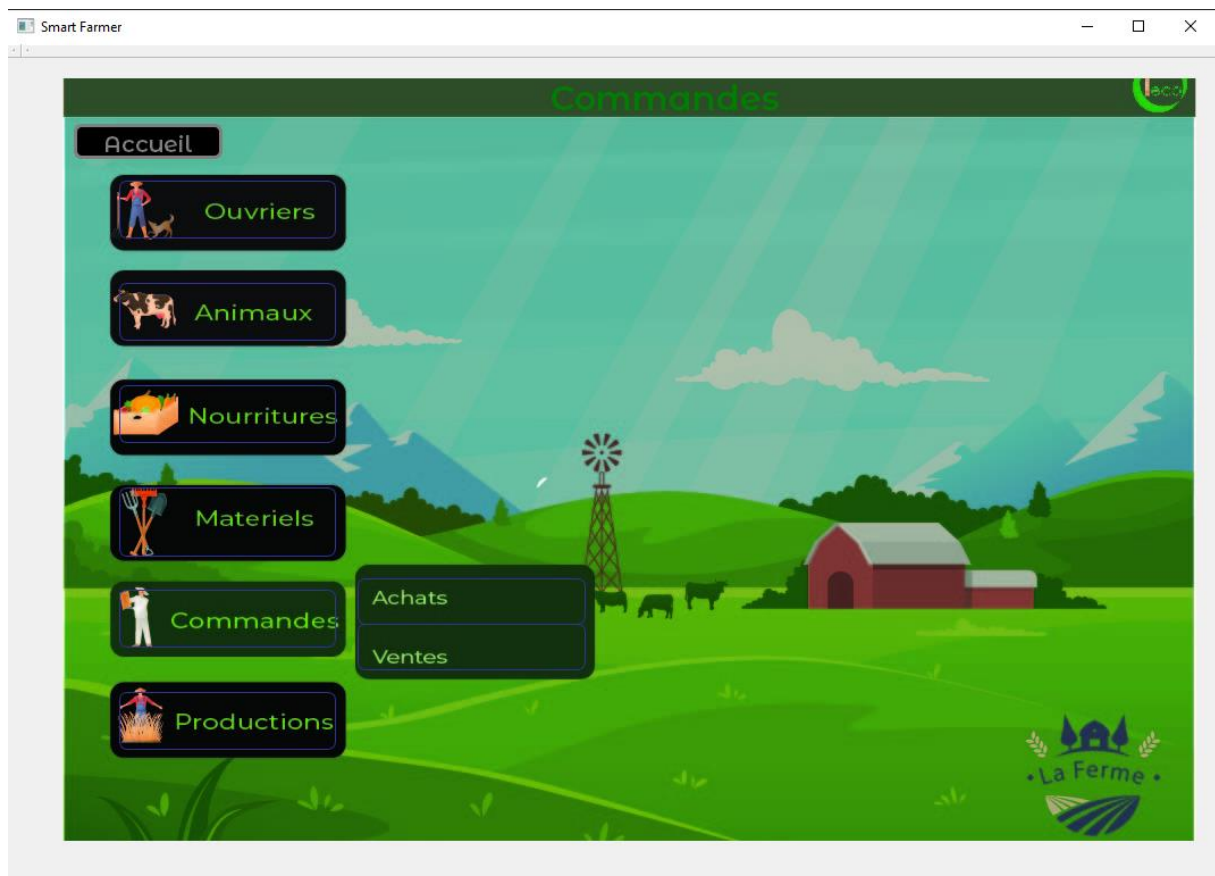


Figure 16 sous menu commande

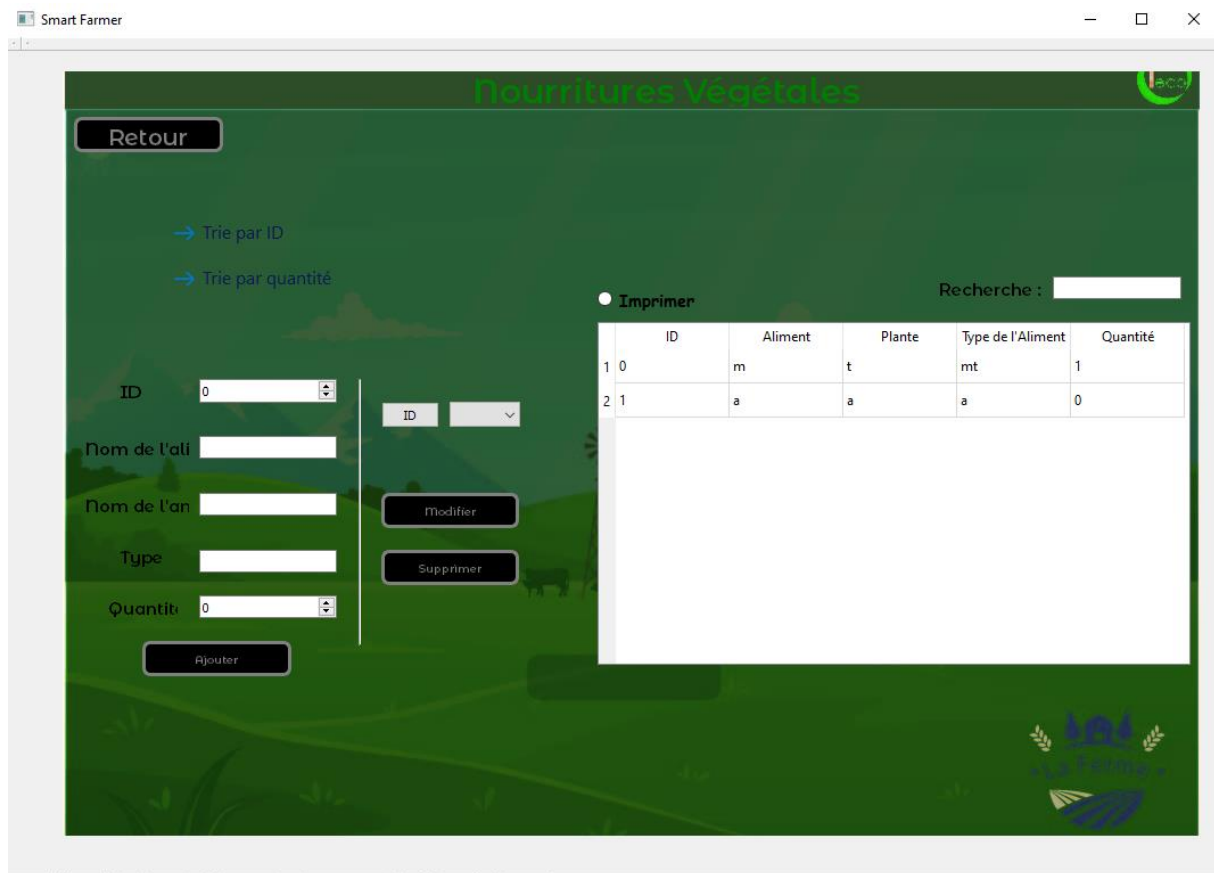


Figure 17 interface gestion de la nourriture

6.5 Conclusion

Vous avez pu observer le rendu de notre travail après la phase de codage voilà pourquoi nous vous invitons à télécharger notre application pour pouvoir tester les différentes fonctionnalités de vous-même.

7 Conclusion générale et perspectives

Parvenus au terme de notre projet, nous pouvons dire que nous avons acheminer notre mission avec succès à savoir la réalisation d'une application

desktop de gestion de ferme intelligente en adéquation avec le cahier de cahier de charges dont il nous a été fait part pour que nous puissions cerner les souhaits et lui présenter les limites des outils à notre disposition à l'heure actuelle, et pour une réalisation optimale nous avons dû suivre les différentes étapes présentées ci-dessus jusqu'à l'obtention du produit final que vous avez pu observer. Dans un souci d'originalité, de qualité, et d'optimisation nous avons œuvré d'arrache pieds pour pouvoir répondre non seulement aux attentes de notre clientèle mais aussi à celui des appli leadeuses et concurrentielles actuellement sur le marché sans toutes fois faire dans le conformisme voilà pourquoi nous terminerons avec cette citation de **STEVE JOBS** qui dit << **pourquoi se joindre à la marine si vous pouvez être un pirate** >>